

UNIVERSITÉ DE NANTES
FACULTÉ DES SCIENCES ET DES TECHNIQUES

ECOLE DOCTORALE 3MPL
MATIERE, MOLECULE, MATERIAUX EN PAYS DE LA LOIRE

Année 2013

Conception, réalisation et exploitation du traitement de données de l'expérience ALICE pour la simulation, la reconstruction et l'analyse

THÈSE DE DOCTORAT
Discipline : Physique
Spécialité : Constituants Élémentaires et Physique Théorique

*Présentée
et soutenue publiquement par*

Federico CARMINATI

Le 22 février 2013, devant le jury ci-dessous

Président : Jaques MARTINO
Rapporteurs : Patrick AURENCHE, Directeur de Recherche Emeritus, LAPTH - Laboratoire d'Annecy-le-Vieux de Physique Théorique ;
Yacine KADI, Professeur, CERN – Organisation Européenne pour la Recherche Nucléaire, Genève, Suisse, & Department of Energy Sciences, Sungkyunkwan University, Republic of Korea.
Examineurs : Sonia KABANA, Professeur II, Université de Nantes, Département de Physique et Laboratoire SUBATECH
Jacques MARTINO, Directeur de l'Institut National de Physique Nucléaire et de Physique des Particules (IN2P3) du CNRS, Paris,
Yves SCHUTZ, CNRS et CERN

Directeur de thèse : Yves SCHUTZ

*- Ou d'aller encore une fois, ô forêt pleine
de mystère, - jusqu'à ce lieu que je connais,
où, dans une eau morte et brunie, trempent
et s'amollissent encore les feuilles des ans
passés, les feuilles des printemps adorables.*

*C'est là que se reposent le mieux mes
résolutions inutiles, et que se réduit à la fin,
à peu de chose, ma pensée.*

André Gide, Paludes

*À Giuliana
sans qui ce travail,
comme beaucoup d'autres choses,
n'aurait pas eu lieu.*

À mes parents.

Remerciements

Je voudrais remercier mon directeur de thèse, le Dr. Y. Schutz, pour son soutien dans cette aventure tardive et inhabituelle. J'ai du mal à imaginer quelqu'un d'autre avec qui j'aurais pu être doctorant et en même temps travailler au jour le jour avec autant de plaisir. Son sens de l'humour, sa compétence et son amitié m'ont été précieux.

Je tiens à remercier l'ensemble des membres du jury pour l'honneur qu'ils m'ont fait en acceptant de juger ce travail de doctorat : Patrick Aurenche, Directeur de Recherche au LAPTH, Annecy ; Sonia Kabana, Professeur à l'Université de Nantes ; Yacine Kadi, directeur du projet HI-ISOLDE au CERN et Professeur à l'Université de Sungkyunkwan (Corée) et Jacques Martino, directeur de l'IN2P3.

Ce travail doit son existence à Giuliana, mon épouse. C'est elle qui m'a encouragé, soutenu et inspiré tout au long, parfois avec discrétion et parfois de façon très explicite. Il m'en a fallu autant pour sortir de mes doutes, et je la remercie pour son énergie et sa détermination. Avec l'amour en prime.

Pour mes collègues, qui m'ont accompagné pendant toutes ces années et avec lesquels nous avons imaginé, bâti et déployé le système de calcul de ALICE, j'éprouve un sentiment de gratitude que les mots ne peuvent exprimer. Ceux qui je voudrais remercier, au CERN mais aussi ceux dispersés aux quatre coins du monde, sont trop nombreux pour les nommer sans craindre d'en oublier. Parmi eux, une personne a tenu une place spéciale, Latchezar Betev, collaborateur fidèle et ami, pierre angulaire sur laquelle tout l'édifice a été bâti. *Parce que c'était lui, parce que c'était moi.*

Un grand merci à Yiota Foka pour son amitié et pour ses conseils tout au long de ce travail.

Merci et merci encore à Maud Struchen, notre chère correctrice, pour sa lecture attentive et intelligente de mon manuscrit.

En dernier je ne peux m'empêcher d'avoir une petite pensée pour ma psychanalyste, Madame Françoise Palmaro, car l'achèvement de ce travail sera pour elle l'occasion d'un changement de sujet dans nos conversations, ce que j'imagine être bienvenu.

Résumé

L'expérience ALICE (A Large Ion Collider Experiment) auprès de l'accélérateur LHC (Large Hadron Collider) du CERN (Organisation Européenne pour la Recherche Nucléaire) s'appuie sur un environnement logiciel intégré pour la conception du dispositif expérimental, l'évaluation de ses performances et le traitement des données. Cet environnement a été conçu par l'auteur. Il inclut des générateurs d'événements, les algorithmes de transport des particules décrivant de façon détaillée les interactions particule-matière (conçus et réalisés par l'auteur), les reconstructions des traces des particules et leur identification, et les analyses finales de physique.

Mots-clés : Ion lourds, logiciel, grille informatique, nuage informatique, génie logiciel, reconstruction, simulation, analyse

Summary

The ALICE (A Large Ion Collider Experiment) at the CERN (Conseil Européen pour la Recherche Nucléaire) LHC (Large Hadron Collider) facility uses an integrated software framework for the design of the experimental apparatus, the evaluation of its performance and the processing of the experimental data. The author has designed this framework. It includes the event generators and the algorithms for particle transport describing the details of the interaction particles-matter (designed and implemented by the author), the reconstruction of the particle trajectories and the final physics analysis.

Keywords: Heavy Ions, software, Grid, Cloud, software engineering, reconstruction, simulation, analysis

Synopsis

Depuis son démarrage en septembre 2010, l'expérience ALICE (A Large Ion Collider Experiment) auprès de l'accélérateur LHC (Large Hadron Collider) du CERN (Organisation Européenne pour la Recherche Nucléaire) a récolté $154 \mu\text{b}^{-1}$ de collisions Plomb-Plomb, $18,2 \text{ pb}^{-1}$ de collisions proton-proton et $60 \mu\text{b}^{-1}$ de collisions proton-Plomb pour un total de $2,9 \times 10^9$ collisions proton-proton, 5×10^8 collisions Plomb-Plomb et 10^7 collisions proton-Plomb. Le détecteur a enregistré 7 Po de données brutes et a produit 10 Po de données pour un total de 550 millions de fichiers, nécessitant pour cela l'équivalent de 55.000 années de temps de calcul. Cette énorme entreprise de traitement de données a été réalisée au cours des trois dernières années sur plus de quatre-vingt centres de calcul, dans trente pays travaillant ensemble grâce à une structure de calcul distribuée appelée Grille informatique, principalement développée par et pour la communauté mondiale de la Physique des Haute Énergies. C'est seulement au prix d'un tel effort que les résultats expérimentaux peuvent être analysés et interprétés pour en extraire les informations « physiques » qui font progresser notre connaissance de la structure intime de la matière.

Pour mettre en œuvre une telle entreprise, la collaboration ALICE a dû développer pendant plus de dix années une suite de programmes, qui compte aujourd'hui plus de 10 millions de lignes de code, et participer activement au développement et au déploiement de la première et de la plus grande structure de calcul intégrée et distribuée au monde. Cette thèse décrit l'architecture de l'infrastructure de calcul de l'expérience ALICE au LHC et son développement, de mon point de vue très personnel qui est celui du coordinateur du calcul qui a guidé ce projet de la phase initiale de conception jusqu'à son déploiement et son application au traitement des données des collisions de protons et d'ions lourds générées par le LHC et détectées par le détecteur construit par la collaboration ALICE.

Si l'effort informatique déployé pour le LHC est considérable, et, dans un sens, d'une ampleur sans précédent, il est tout aussi vrai que depuis ses débuts la Physique des Hautes Énergies s'est fortement appuyée sur les ordinateurs pour extraire les résultats de physique à partir des observations expérimentales.

Toutefois, la complexité de l'informatique pour la Physique des Hautes Énergies n'est pas seulement due à la quantité des ressources nécessaires, mais aussi aux conditions dans lesquelles ces ressources sont agencées et dans lesquelles le logiciel est développé et maintenu. Des nos jours, une expérience de Physique des Hautes Énergies est constituée par une vaste collaboration internationale comptant des milliers de physiciens, ingénieurs et techniciens provenant de centaines d'instituts dans des dizaines de pays différents. Ces chercheurs, ne se rencontrant que rarement, et avec peu de structure hiérarchique entre eux, travaillent indépendamment les uns des autres avec et sur le même code et grâce à une structure informatique unique. Les exigences changent très fréquemment et les problèmes à résoudre repoussent souvent les limites de la connaissance scientifique, à la fois en physique, mais aussi en informatique. On peut dire que la recherche scientifique en physique fondamentale est limitée non pas par la

créativité de ses chercheurs, mais par l'état de l'art actuel de l'ingénierie et de l'informatique.

Il est ainsi fréquent que les physiciens soient à l'origine de développements importants dans le domaine de l'informatique. Parmi les exemples notables, citons l'invention du Toile au CERN et, plus récemment, le développement et le déploiement, pilotés par le CERN, de la Grille de calcul pour le stockage et le traitement des données collectées par les expériences du LHC. Bien qu'actuellement la Grille soit encore en plein essor, l'attention de l'informatique pour la Physique des Hautes Énergies se déplace déjà vers le calcul « de nuage » (en anglais *Cloud Computing*), le calcul massivement parallèle et la virtualisation, pour anticiper les nouvelles technologies à venir.

Bien que ces développements se popularisent à l'extérieur du domaine de la Physique des Hautes Énergies, et qu'ils aient bénéficié d'une certaine exposition médiatique à l'intérieur des expériences de physique, la reconnaissance du métier de « physicien programmeur » comme une des « professions de la physique » est loin d'être acquise. Les physiciens programmeurs eux-mêmes ne se considèrent la plupart du temps que comme des physiciens qui « par nécessité » et « temporairement » se consacrent à l'informatique, même si ce temporaire peut parfois durer le temps d'une carrière.

Une des conséquences de cet état d'esprit a été que plusieurs techniques et concepts de calcul, développés et appliqués dans le cadre de la Physique des Hautes Énergies, ont eu du succès seulement après avoir été « réinventés » et développés indépendamment, souvent des années plus tard. Pour n'en citer que deux exemples, la distribution libre du logiciel source ou les techniques de développement « agiles » ont été utilisées pendant plusieurs années dans la Physique des Hautes Énergies, mais n'ont été reconnues comme des entités conceptuelles importantes qu'après avoir été « conceptualisées » dans d'autres environnements.

Tout comme les détecteurs, les systèmes informatiques pour la Physique des Hautes Énergies sont des réalisations uniques, souvent à la pointe, voire au-delà, de l'état de l'art de la technologie. L'expérience nous a montré que pour réaliser ces systèmes, il est nécessaire de créer un métier associant une formation de physicien à des connaissances approfondies d'ingénieur informatique, sans oublier des compétences de gestionnaire de projet. Effectivement, seule une compréhension détaillée des objectifs et des méthodes utilisées par les physiciens dans leur travail permet de comprendre leurs besoins, à la fois en termes de fonctionnalité mais aussi d'évolution de l'outil informatique et de communication.

Cela ne signifie pas que les ingénieurs informatiques n'aient pas leur place dans les développements des systèmes de calcul pour la Physique des Hautes Énergies, mais la participation des physiciens au développement de leur outil informatique, notamment dans le rôle de planification et de direction du projet de calcul, reste fondamentale.

La principale raison qui a motivé l'écriture de thèse relève de la volonté de présenter de façon détaillée le système informatique d'une expérience de Physique des Hautes Énergies actuelle, non seulement par ses résultats, mais en

décrivant ses défis et la façon dont ils ont été abordés. Nous espérons ainsi pouvoir fournir une contribution «théorique» pour expliquer les techniques et les «traditions» de l'informatique de la Physique des Hautes Énergies, avec l'intention de les faire mieux connaître et de fournir les arguments nécessaires à la promotion du dialogue et de la collaboration interdisciplinaire dans d'autres domaines où l'informatique est soit l'objet de recherche en soi, soit un outil essentiel. Le moment me semble très opportun, parce qu'une nouvelle génération d'expériences vient de démarrer après près de 20 années de développement, ce qui nous permet de décrire une situation dans le domaine du calcul mature et accomplie. La prochaine génération d'expériences presque certainement introduira des changements radicaux dans ce domaine, ou du moins je l'espère.

Synopsys

Since it started in September 2010, the ALICE experiment (A Large Ion Collider Experiment) at the LHC (Large Hadron Collider) at CERN (European Organization for Nuclear Research) has collected $154 \mu\text{b}^{-1}$ Lead-Lead collisions, 18.2 pb^{-1} of proton-proton collisions and $60 \mu\text{b}^{-1}$ of proton-Lead collisions for a total of 2.9×10^9 proton-proton, 5×10^8 Lead-Lead and 10^7 proton-Lead collisions. The detector has recorded 7 PB of raw data and has produced 10 PB of data for a total of 550 million files, requiring the equivalent of 55,000 years of computing time. This huge data processing effort was carried out during the last three years on more than eighty computer centres in 30 countries working together via a distributed computing infrastructure called Computing Grid, primarily developed by and for the global high energy physics community. Only at the price of such an effort, can the experimental results be analysed and interpreted to extract the “physical” information for the advancement of our knowledge of the intimate structure of matter.

To reach this goal, the ALICE collaboration has been developing for more than ten years a set of programs, which now has over 10 million lines of code, and has actively participated in the development and deployment of the first and largest integrated worldwide computing infrastructure. This thesis describes the architecture of the computing of the ALICE experiment at LHC and its development infrastructure, from my very personal point of view as Computing Coordinator, having guided this project from the initial design phase to its deployment and its use for the treatment of data of protons and heavy ions collisions generated by the LHC and detected by the detector built by the ALICE collaboration.

If the computer effort for the LHC is considerable, and, in a sense, of unprecedented scale, it is also true that since its inception high-energy physics has heavily relied on computers to extract the physics results from experimental observations.

The complexity of computing for high-energy physics is not only due to the amount of resources deployed, but also to the conditions in which these resources are arranged and the software is developed and maintained. Nowadays, a high-energy physics experiment is formed by a large international collaboration of thousands of physicists, engineers and technicians from hundreds of institutes in dozens of different countries. Thanks to a single computing infrastructure, these researchers, who rarely meet, work in a structure with little hierarchy independently of each other on the same code. Requirements change very frequently and problems often push the limits of scientific knowledge, both in physics, but also in computer technology. We can say that scientific research in fundamental physics is not limited by the creativity of its researchers, but by the current state of the art of engineering and computer science.

Thus, it is common that physicists are at the origin of important developments in the field of information technology. Notable examples include the invention of the Web at CERN and, more recently, the development and deployment, lead by CERN, of the LHC Computing Grid for the storage and treatment of data collected

by the LHC experiments. Although currently the Grid is still in full swing, the attention of computer science for high energy physics already moves to "Cloud" computing, massively parallel computing and virtualization, to anticipate the new technologies to come.

Although these developments are becoming popular outside the field of high-energy physics, having received some media exposure within physics, the recognition of the profession of "programming physicist" as one of the "professions of Physics" is far from acquired. The programming physicists do consider themselves most of the time as physicists who "by necessity" and "temporarily" are doing computing, even if this temporary can sometimes be for the duration of a career.

One of the consequences of this attitude has been that several computing techniques and concepts, developed and deployed by high energy physics, found their success only after being "reinvented" and developed independently, often years later. Two examples of this are: the free distribution of the program source and the usage of "agile" development techniques. Both have been used for many years in high energy physics, but have been recognized as important concepts only after being "conceptualized" in other environments.

Detectors and computer systems for high-energy physics are unique achievements, often at the edge, or beyond, state-of-the-art technology. Experience has shown that to realise these systems, it is necessary to create a professional hybrid, by combining physicist training with in-depth knowledge of computer engineering, and project management skills. It is a fact that only a detailed understanding of the objectives and the methods used by physicists in their work allows understanding of their needs, both in terms of functionality but also to relative to the developments in computer and communication.

This does not mean that computer engineers do not have their place in the development of the high-energy physics computing systems, but the participation of physicists in the development of their computing tools, particularly in the planning and management of the computing project, remains fundamental.

The main reason for writing this thesis is the desire to present in detail the computing infrastructure of a modern high-energy physics experiment, not only in terms of its results, but also in terms of the challenges that have been faced and how they have been tackled. It is intended to make a "theoretical" contribution to illustrate the techniques and "traditions" of computing in high-energy physics, with the intention of raising awareness and providing the necessary arguments for the promotion of dialogue and interdisciplinary collaboration with other areas where Information Technology is either the subject of research itself, or an essential tool. The timing seems very appropriate: a new generation of experiments has just started up after nearly 20 years of development allowing the description of a mature and accomplished situation in the field of computing. It is hoped that the next generation of experiments will almost certainly introduce radical changes in this field.

Table des matières

Table des matières.....	15
Index des figures.....	17
Index des tables.....	25
Introduction	27
1 Le Programme de Physique du détecteur ALICE	37
1.1 Aperçu théorique.....	37
1.2 Observables du détecteur ALICE dans les collisions des ions lourds.....	44
2 Le détecteur ALICE et son modèle de calcul	55
2.1 Les conditions expérimentales.....	55
2.2 L'organisation hiérarchique de l'expérience ALICE.....	58
2.3 Le détecteur ALICE	60
2.4 Le traitement des données.....	90
2.5 Le traitement des données en-ligne.....	91
2.6 Le traitement des données hors-ligne	100
2.6.1 La simulation.....	100
2.6.2 La reconstruction.....	101
2.6.3 Alignement et calibrage.....	102
2.6.4 L'Analyse.....	104
2.7 Le processus de traitement des données.....	105
3 Le modèle de développement de AliRoot	107
3.1 Le logiciel dans la Physique des Hautes Énergies	107
3.1.1 Le génie logiciel classique	108
3.1.2 Génie logiciel et Physique des Hautes Énergies	110
3.1.3 Le cycle de développement.....	112
3.1.4 Vérification et documentation	114
3.1.5 La question du portage du code	116
3.1.6 Le facteur humain.....	118
3.1.7 Génie logiciel moderne	120
3.2 Le développement de AliRoot.....	124
3.2.1 L'organisation du code	125
3.2.2 Le cycle de développement.....	127
4 La structure du code AliRoot.....	133
4.1 Histoire de AliRoot	133
4.2 Le logiciel Root.....	138
4.3 Structure de AliRoot.....	141
4.4 Principes de conception de AliRoot	142
4.5 La simulation.....	145
4.6 Système d'alignement et de calibrage.....	149
4.7 La reconstruction	154
4.7.1 Reconstruction dans les détecteurs centraux.....	158
4.7.2 Reconstruction dans le spectromètre MUON.....	162
4.7.3 Identification des particules chargées.....	163
4.7.4 La reconstruction dans les autres détecteurs.....	164

4.8	L'analyse.....	166
4.9	La visualisation	172
5	La Grille informatique de ALICE.....	179
5.1	Définition et historique de la Grille du point de vue de la Physique des Hautes Énergies	179
5.1.1	<i>La naissance de la Grille et de la boîte à outils Globus</i>	<i>179</i>
5.1.2	<i>Le modèle de calcul MONARC.....</i>	<i>183</i>
5.1.3	<i>La Grille dans la Physique des Hautes Énergies</i>	<i>185</i>
5.1.4	<i>Le développement de l'intergiciel de Grille de LHC.....</i>	<i>186</i>
5.1.5	<i>Le développement de AliEn.....</i>	<i>190</i>
5.2	Structure de la Grille de ALICE	195
5.2.1	<i>Le catalogue de fichiers de AliEn</i>	<i>196</i>
5.2.2	<i>Le gestionnaire de charge de travail de AliEn.....</i>	<i>205</i>
5.2.3	<i>Le système de contrôle MonALISA.....</i>	<i>212</i>
5.2.4	<i>L'évolution future de la Grille</i>	<i>218</i>
5.3	Utilisation de la Grille de ALICE	222
5.4	Conclusions et perspectives	223
6	AliRoot en action	227
6.1	Propriétés globales de la collision	229
6.1.1	<i>Multiplicité.....</i>	<i>229</i>
6.1.2	<i>Abondance des particules identifiées.....</i>	<i>232</i>
6.1.3	<i>Spectre des particules</i>	<i>233</i>
6.1.4	<i>L'état initial de la collision.....</i>	<i>234</i>
6.2	Propriétés du QGP.....	234
6.2.1	<i>Écoulement collectif des particules.....</i>	<i>235</i>
6.2.2	<i>Suppression des jets</i>	<i>237</i>
6.3	La saga du J/ψ	240
7	Conclusions	243
8	Glossaire.....	247
9	Références.....	263

Index des figures

Figure 1 : Évolution du nombre de composants fondamentaux au cours de l'histoire de l'humanité.....	27
Figure 2 : Diagramme de phase de la matière QCD. Les lignes continues indiquent les transitions du premier ordre. La ligne pointillée indique une région où la transition est soudaine mais continue. Le point séparant les deux régions indique le point critique du second ordre.....	37
Figure 3: Le domaine des variables Bjorken- x et M^2 pour la production de particules atteignable dans les collisions noyau-noyau aux énergies du SPS, RHIC et LHC. M est la masse invariante de l'état final de la collision. Avec l'augmentation de l'énergie des faisceaux il est possible de créer des particules de masse plus élevée grâce à des collisions plus « dures » (petites valeurs de x).	40
Figure 4 : Densité d'énergie quanta-chromodynamique Les trois lignes sont les prévisions pour deux saveurs de quarks légers (up et down, courbe rouge), trois saveurs légères (up, down et étrange, courbe bleue) et pour le cas plus réaliste de deux saveurs légères (up et down) et une plus massive (étrange, courbe verte). Les calculs ont été effectués dans un treillis $N_\tau = 4$ avec gauge améliorée et action calculée pour des fermions superposés. Les températures maximales initiales atteintes avec les différents accélérateurs sont indiquées. Les flèches de couleur sur l'ordonnée montrent la valeur de la limite de Stefan-Boltzmann pour un gaz idéal de quarks et gluons (Nucl. Phys. B 605, 579–599 (2001)).....	43
Figure 5 : Luminosité moyenne avec $\mathcal{L}^0 = 10^{27} \text{ cm}^{-2} \text{ s}^{-1}$ et $I_0 = 6,8 \times 10^7$ ions par paquet pour les collisions Plomb-Plomb avec un (trait plein), deux (ligne à tirets) et trois régions d'interaction (ligne pointillée).....	56
Figure 6 : Organisation de l'expérience ALICE.....	59
Figure 7 : Représentation du détecteur ALICE dans sa configuration de 2010. La signification des acronymes est donnée dans le texte.	61
Figure 8 : Plan du système trajectographique interne.	62
Figure 9: Structure du détecteur SPD.	63
Figure 10 : Résolution de la mesure de la position du vertex déterminée à partir des données de l'ITS. La résolution est estimée en divisant aléatoirement les particules de l'événement en deux groupes et en calculant la différence entre les positions du vertex obtenues à partir des deux « sous-événements ». Les événements utilisés sont les collisions Plomb-Plomb à 2,76 TeV.....	64
Figure 11 : Résolution du paramètre d'impact (distance du vertex de la ligne du faisceau) dans le plan perpendiculaire aux faisceaux pour les trajectoires	

reconstruites à partir du seul ITS pour des données réelles et des données simulées.....	65
Figure 12 : Résolution relative de la perte d'énergie spécifique pour les traces reconstruites à partir des données du seul ITS en fonction de l'impulsion transversale. Les données utilisées proviennent des collisions proton-proton de 2010 et 2011.....	66
Figure 13 : Plan de la chambre à projection temporelle (TPC) de ALICE.....	67
Figure 14 : Distribution de la perte d'énergie spécifique, mesurée par la TPC, des traces de tous les événements incluant au moins une particule identifiée avec une charge électrique supérieure à 1 en valeur absolue. Les collisions Plomb-Plomb à 2,76 TeV de l'année 2011 ont été utilisées. Seules les particules chargées négativement sont représentées. Les dix candidats de noyaux anti-alpha sont indiqués en rouge.	68
Figure 15 : Résolution relative de la mesure de l'impulsion transversale obtenue avec la reconstruction des traces à partir des données combinées de la TPC et de l'ITS. Les événements utilisés proviennent de la deuxième reconstruction des collisions Plomb-Plomb à 2,76 TeV de 2010 (période de prise de données LHC10h).....	69
Figure 16 : Plan du détecteur à transition de radiation.	70
Figure 17 : Charge déposée dans le TRD par les électrons et les pions d'impulsion de 2 GeV/c. Pour les électrons, le dépôt de charge moyen est plus élevé en raison de la contribution supplémentaire de l'absorption du rayonnement de freinage (Bremsstrahlung) et du rayonnement de transition dans le gaz des chambres à dérive.	71
Figure 18 : Signal du méson J/ψ dans la distribution de masse invariante des paires d'électrons identifiés dans les collisions Plomb-Plomb pour une classe de centralité entre 0 et 40 %. L'apport du TRD à la réduction du bruit de fond obtenue en réduisant la contamination des hadrons mal identifiés comme électrons y est démontrée.....	72
Figure 19 : Plan du détecteur de temps de vol avec détail de l'installation dans l'armature spatiale.....	73
Figure 20 : Distribution des particules détectées en fonction de leur vitesse relative (β) mesurée par le TOF et de leur impulsion. Les données utilisées sont les événements Plomb-Plomb de 2011. Le bruit de fond est dû à des erreurs d'appariage entre le TOF et la TPC.....	74
Figure 21 : Résolution de la mesure du temps de vol par le TOF. Des pions sont sélectionnés dans un intervalle étroit d'impulsions (0,95 GeV/c < p 1,05 GeV/c).....	75
Figure 22 : Plan du calorimètre électromagnétique.	76

Figure 23 : Comparaison des résolutions en énergie de EMCAL obtenues à partir de données simulées et à partir des données de faisceau test d'électrons mono-énergétiques.. La fonction ajustée aux données est	77
Figure 24 : Distribution de la masse invariante des paires rayon gamma rayon gamma dans EMCAL dans le cas des collisions proton-proton à 7 TeV pour un intervalle d'impulsion transversale entre 16 et 18 GeV/c.....	78
Figure 25 : Plan de l'identificateur de particules à haute impulsion.....	79
Figure 26 : Distribution de l'angle Tcherenkov en fonction de l'impulsion dans l'HMPID pour les particules primaires chargées émises dans les collisions proton-proton à 7 TeV. Les courbes indiquent les distributions théoriques attendues pour les pions, les kaons et les protons.	80
Figure 27 : Pouvoir de séparation de l'HMPID en unités d'écart type moyen en fonction de l'impulsion transversale, obtenu pour des particules produites dans les collisions proton-proton à 2,76 TeV. Les lignes pointillées indiquent les niveaux de séparation à deux et trois écarts types moyens. Les valeurs obtenues sont compatibles avec les spécifications de conception du détecteur.....	81
Figure 28 : Spectre de masse invariante mesuré par PHOS des paires de photons entre 1 et 2 GeV/c produits dans les collisions proton-proton à 2,76 TeV avec (distribution supérieure) et sans (distribution inférieure) bruit de fond autour du pic du π^0 . Les données utilisées sont le résultat de la troisième reconstruction de la période de prise de données LHC11a en 2011.....	82
Figure 29 : Spectre de masse invariante des paires de muons détectées par le spectromètre MUON indiquant clairement les pics de masse du J/ψ et du ψ' . Les données utilisées sont celles des collisions proton-proton à 2,76 GeV..	83
Figure 30 : Distribution de l'énergie détectée par ZDC dans le calorimètre à neutrons en fonction de l'énergie détectée (unité arbitraire) dans le calorimètre électromagnétique. Les régions de centralité surimposées sont déterminées par la simulation.....	84
Figure 31 : Dépendance de l'énergie moyenne des photons détectés par le PMD à l'énergie du faisceau dans les collisions proton-proton.....	85
Figure 32 : Moyenne du cosinus de l'harmonique d'ordre k de la distribution du plan de réaction calculé à partir des données de FMD et deux sous-événements sélectionnés à partir des données de la TPC en fonction de la centralité.....	86
Figure 33 : Distribution de la densité des particules chargées par unité de rapidité mesurée avec le SPD, le FMD et le V0 pour les collisions Plomb-Plomb à 2,76 TeV. Plusieurs classes de centralité sont représentées.....	87
Figure 34 : Différence entre les temps de détection du signal de l'interaction mesurés par chacun des deux détecteurs T0, et corrigés par la position du	

vertex d'interaction. La distribution supérieure montre l'évolution de la résolution sur un certain nombre de périodes de prise de données.....	88
Figure 35 : Distribution de la multiplicité des muons détectés dans ACORDE comparée aux distributions simulées pour des protons et des noyaux de Fer comme rayons primaires.....	89
Figure 36 : Plan du système d'acquisition de données de l'expérience ALICE.....	94
Figure 37 : Plan du système de contrôle de l'expérience ALICE.....	98
Figure 38 : Synopsis de la programmation XP.	122
Figure 39: Évolution du dépôt de AliRoot.....	125
Figure 40: Aperçu du système de compilation et vérification de AliRoot.....	128
Figure 41 : Aperçu des librairies de Root.....	139
Figure 42 : Représentation fonctionnelle de AliRoot.....	142
Figure 43 : La structure de AliRoot.	143
Figure 44 : Conception et schéma du MonteCarlo Virtuel.	145
Figure 45 : Description de la structure des classes du MonteCarlo Virtuel.....	146
Figure 46: Vue schématique des relations entre les procédures d'alignement et de calibrage et les sources de données dans les systèmes AliRoot et AliEn.	150
Figure 47: Aperçu du flux général des données pendant la reconstruction.	155
Figure 48: Vue schématique des trois passages pour la reconstruction des trajectoires dans le « baril ».....	159
Figure 49: Efficacité combinée des reconstruction (symboles pleins) et de la probabilité d'obtenir une piste fausse (symboles vides) en fonction de l'impulsion transversale pour des différentes densités de particules chargées.	161
Figure 50 : Efficacité et contamination d'identification d'un Kaon pour ITS, TPC et TOF seuls (trait plein) et pour leur combinaison (points avec barres d'erreur).....	163
Figure 51 : Train d'analyse construit à partir des tâches.....	169
Figure 52 : Vue d'ensemble du flux de données dans l'analyse.....	170
Figure 53 : Aperçu du flux de travail dans l'analyse montrant que le même code peut être utilisé pour des modes d'exécution différents de façon transparente.....	171

Figure 54 : Géométrie du détecteur ATLAS dans EVE. L'Explorateur d'objets affiche la hiérarchie de nœuds. Une boîte de découpage est utilisée pour montrer la région centrale du détecteur.....	172
Figure 55 : En haut : collision proton-proton à 14 TeV. En bas : collision Plomb-Plomb à 5,5 TeV par nucléon. À gauche les trajectoires simulées et à droite les trajectoires reconstruites en partant des données simulées.	173
Figure 56 : Représentation des données calorimétriques dans EVE avec TEveCalo3D (à gauche) et TEveCaloLego (à droite).....	174
Figure 57 : En haut : exemples de visualisation de données brutes. En bas visualisation de données TPC de ALICE pour des événements Plomb-Plomb. Des sous-classes spécifiques ont été développées pour l'affichage des données en 2D (gauche et droite) et 3D (à droite).....	175
Figure 58 : Visualisation en ligne de la première collision proton-proton à 900 GeV enregistrée par le détecteur ALICE en 2010.....	177
Figure 59 : Schéma simplifié d'une Grille basée sur la boîte à outils Globus.....	180
Figure 60 : Modèle de calcul pour LHC basé sur MONARC. Cette figure a été tirée du rapport final du projet MONARC. Les chiffres de la capacité du réseau en 2005 sont les estimations faites en 2000.	184
Figure 61 : Aperçu des services de AliEn.....	192
Figure 62 : AliEn vu comme Grille superposée.....	193
Figure 63 : Aperçu de l'évolution de AliEn depuis le début du projet jusqu'à aujourd'hui.....	195
Figure 64 : Exemple de l'interface de ligne de commande au catalogue de fichiers AliEn.....	196
Figure 65 : Aperçu de l'interface graphique au catalogue de fichiers de AliEn. .	197
Figure 66 : Schéma du système de gestion des fichiers xRootd. La première fois que le client demande d'accéder au fichier X (a), le redirecteur demande à tous les serveurs de données qui a le fichier X (b). Quand un serveur répond (c), il communique l'information au client (d) qui ouvre une connexion directe au serveur pour accéder au fichier. L'information est sauvegardée dans la mémoire du redirecteur, et la deuxième fois qu'un client demande le fichier X les étapes b) et c) ne sont pas exécutées.	198
Figure 67 : Évolution du nombre d'entrées dans le catalogue de fichiers de AliEn. Le nombre de GUIDs est supérieur au nombre de noms logiques car les fichiers générés par des tâches en erreur ou les fichiers en cours d'effacement sont encore présents dans le catalogue, mais ils ne sont pas « exposés » aux utilisateurs avec un nom logique.	202

Figure 68 : Trafic intégré des éléments de stockage de la Grille de ALICE à partir du démarrage de l'accélérateur.	203
Figure 69 : Résultat de la recherche du terme « big data » sur Google trends montrant l'intérêt croissant pour le traitement de grandes quantités de données.	204
Figure 70 : Flux d'exécution d'une tâche du gestionnaire de charge de travail de AliEn. La tâche pilote envoie régulièrement des signaux au gestionnaire des ressources de calcul pour l'informer qu'elle est active. Ces signaux s'appellent « battements du cœur » (en anglais <i>heartbeat</i>). Si ces signaux cessent, la tâche est déclarée « ZOMBIE » (mort vivant) et après un certain temps elle est déclarée expirée et considérée comme perdue.....	205
Figure 71 : Schéma de fonctionnement d'une tâche maître.....	206
Figure 72 : Vue d'ensemble des tâches impliquées dans la reconstruction d'une période de prise de données. La dépendance entre ces tâches peut être exprimée comme un graphe acyclique orienté. L'ensemble est visualisé avec le système de surveillance MonALISA.....	207
Figure 73 : Évolution du nombre de tâches sur la Grille de ALICE depuis 2006.	210
Figure 74 : Les quatre couches, les services de base et les composants de l'infrastructure de MonALISA. Le terme Proxies indique la couche de services de procuration.....	212
Figure 75 : La page principale du système de surveillance MonALISA pour l'expérience ALICE.....	213
Figure 76 : Données brutes récoltées par ALICE.....	214
Figure 77 : Évolution du nombre moyen de tâches en exécution sur la Grille de ALICE.	217
Figure 78 : Contribution des différents centres de calcul à la Grille de ALICE en termes de temps de calcul fourni.	218
Figure 79 : Pages de MonALISA avec l'état de tous les éléments de stockage sur disque de ALICE.....	219
Figure 80 : Éléments de stockage sur bande dans la Grille de ALICE.	220
Figure 81 : Aperçu des essais de performance exécutés par MonALISA entre chaque paire de VOBoxes avec l'historique.....	221
Figure 82 : Évolution des références à Grille (en bleu) et Nuage (en rouge) sur la toile depuis 2005.....	222
Figure 83 : Densité de particules chargées par unité de pseudorapidité et par paire de nucléons ($0,5 N_{part}$) mesurée dans les collisions Plomb-Plomb et	

proton-proton en fonction de l'énergie du centre de masse (à gauche) et en fonction du nombre de nucléons participants (à droite), comparée avec les prédictions théoriques pour les collisions Plomb-Plomb à 2,67 TeV. Les données de RHIC ont été normalisées à celles du LHC à $N_{part} = 350$, en utilisant un facteur multiplicatif de 2,14. Les valeurs à 2,76 TeV sont une combinaison statistique des résultats obtenus par trois détecteurs du LHC.229

Figure 84 : Densité de particules chargées en fonction de la pseudorapidité mesurée dans les collisions proton-Plomb à $\sqrt{s_{NN}} = 5,02$ TeV, comparée aux prédictions théoriques [173, 174, 175, 176, 177].230

Figure 85 : Production de particules dans les collisions Plomb-Plomb à $\sqrt{s_{NN}} = 2,76$ TeV comparée aux prédictions d'un modèle statistique pour deux températures, l'une ($T=152$ MeV, $\mu_B=1$ MeV) résultant d'un ajustement aux données (ligne pointillée) et l'autre d'un ajustement ($T=164$ MeV, $\mu_B=1$ MeV-fixé) excluant les baryons (proton et Λ) (ligne pleine).231

Figure 86 : Spectres en impulsion transverse mesurée pour diverses espèces de particules mesurés à rapidité centrale dans les collisions Plomb-Plomb à $\sqrt{s_{NN}} = 2,76$ TeV (ALICE) et Or-Or à $\sqrt{s_{NN}} = 200$ GeV (RHIC). Les lignes pleines sont le résultat d'un ajustement aux données ALICE d'une fonction « Blastwave » avec $\langle\beta_t\rangle=0,65c$ et $T=95$ MeV.232

Figure 87 : Spectre des photons directs mesuré dans les collisions centrales (0-40 %) Plomb-Plomb à $\sqrt{s_{NN}} = 2,76$ TeV. Les lignes continues représentent le résultat d'un ajustement d'une loi de puissance à la partie haute énergie et d'une fonction exponentielle à la partie basse énergie.234

Figure 88 : Spectres de l'écoulement elliptique en fonction de l'impulsion transversale pour des espèces de particules de masse différente produites dans les collisions Plomb-Plomb à $\sqrt{s_{NN}} = 2,76$ TeV et de centralité intermédiaire.235

Figure 89 : Spectres de l'écoulement elliptique pour différentes espèces de particules produites dans les collisions Plomb-Plomb à $\sqrt{s_{NN}} = 2,76$ TeV et de centralité intermédiaire. À gauche, écoulement elliptique en fonction de l'impulsion transverse. À droite, écoulement elliptique en fonction de l'énergie cinétique transverse, normalisé au contenu en quarks de valence des particules. Les valeurs de v_2 sont calculées avec la méthode du plan de l'événement (EP dans la figure) et des cumulants à deux particules avec intervalle de rapidité (SP dans la figure) [195].236

Figure 90 : (à gauche) Coefficient de l'écoulement elliptique v_2 des mésons D^0 , D^+ et D^{*+} mesuré dans les collisions Plomb-Plomb à $\sqrt{s_{NN}} = 2,76$ TeV dans la classe de centralité 30-50 % comparé au v_2 des hadrons chargés. (à droite) v_2 de J/ψ en fonction de l'impulsion transverse.237

Figure 91 : Facteur de modification nucléaire R_{AA} pour les particules chargées mesurées par ALICE dans les collisions Plomb-Plomb centrales (0-5 %) en

comparaison avec les résultats de CMS et différents modèles théoriques.
 Voir la référence [202] pour les détails.....238

Figure 92 : (gauche) R_{AA} en fonction de p_t pour $\pi^\pm$, $K^\pm$, p et p pour les collisions
 Plomb-Plomb à centralité 0-5 %. (droite) R_{AA} en fonction de p_t pour D^0 , D^+ ,
 D^{*+} et D_s^+ pour la classe de centralité 0-7,5 %.....239

Figure 93 : Facteur de modification nucléaire des particules chargées en fonction
 de l'impulsion transverse pour les collisions proton-Plomb à $\sqrt{s_{NN}}=5,02$ TeV. Les données sont comparées à des mesures pour les collisions
 Plomb-Plomb à $\sqrt{s_{NN}}=2,76$ TeV dans deux classes de centralité différentes :
 0-5 % and 70-80 %.....240

Figure 94 : R_{AA} en fonction du nombre des participants pour deux intervalles
 d'impulsion transversale ($0 < p_t < 2$ et $5 < p_t < 8$ GeV/c), superposée aux
 résultats d'un modèle de transport partonique. Les contributions des
 composantes primordiales (en pointillés) et de régénération (ligne
 continue) sont représentées, de même que leur somme. La zone comprise
 entre les courbes représente une estimation de l'incertitude théorique....241

Index des tables

Table 1 : Paramètres pour les faisceaux proton-proton et Plomb-Plomb pour ALICE au LHC. La grande valeur de σ_t pour les collisions proton-proton dépend de la valeur de β^* qui est modifiée pour ALICE pour réduire la luminosité. Pour les deux premières années d'opération, la valeur de l'énergie est la moitié de celle indiquée. Dans la période de mise au point du collisionneur le nombre de paquets est inférieur à celui indiqué.....	57
Table 2 : Valeurs des sections efficaces mesurées dans les collisions proton-proton. Les erreurs marquées <i>model</i> sont dues à l'utilisation de différentes simulations pour déterminer l'efficacité. L'incertitude marquée <i>lumi</i> , correspond à l'incertitude dans la détermination de la luminosité lors des balayages de van der Meer.....	58
Table 3 : Exemple d'un mélange de classes de déclenchement. Ce mélange a été utilisé pendant la prise de données des collisions Plomb-Plomb en 2011. Dans la colonne de gauche nous avons le type d'événement enregistré et dans celle de droite la fréquence attendue.	91
Table 4 : Dimensions moyennes des événements mesurées pendant les deux premières années de fonctionnement. La taille effective de chaque événement change de façon substantielle selon les modalités de la collision.	93
Table 5 : Tableau récapitulatif du code de AliRoot	126
Table 6 : Composition de la source de Root	140

Introduction

La passion que l'homme met à distinguer le fondamental de l'éphémère et à rechercher l'origine de la réalité telle qu'il la perçoit est aussi ancienne que sa prise de conscience et, donc, que l'histoire de l'humanité. En intégrant la religion et, par bien des aspects, l'art dans cette quête de l'essentiel, il est juste d'affirmer que telle est la principale (pré)occupation philosophique de l'homme depuis que l'homme pensant est.

L'origine historique de la recherche scientifique dans laquelle s'intègre mon sujet de thèse remonte à la révolution scientifique et intellectuelle du XVII^{ème} siècle initiée par Galilée lorsqu'il introduisit la mathématique comme langage universel de la connaissance scientifique et l'expérience raisonnée comme méthode d'acquisition de cette connaissance. Le mouvement des corps était expliqué grâce à l'élaboration ontologique des lois *universelles* de la gravitation, reliant terre et cieux et établissant l'unité scientifique de l'univers : la première étape vers la synthèse newtonienne. En même temps, l'identification des composants élémentaires de la matière sortait peu à peu des laboratoires d'alchimistes pour devenir objet d'études des chimistes.

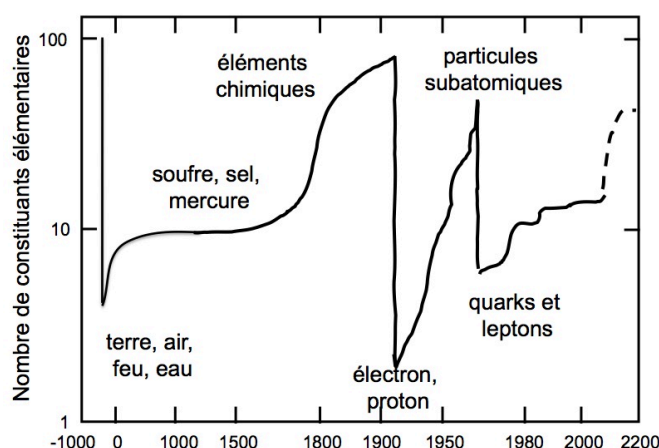


Figure 1 : Évolution du nombre de composants fondamentaux au cours de l'histoire de l'humanité.

Il est intéressant de noter que C. G. Jung considère que les alchimistes sont aussi à l'origine de l'étude de la psyché et pas seulement celle des transformations de la matière [1]. Selon la conception de Jung de *l'Unus Mundus* [2], il y a une interprétation de la réalité qui dépasse la dichotomie corps-âme. Donc toute étude de la réalité physique est aussi une étude de l'humain et vice-versa.

La composition de la matière, initialement considérée par les philosophes grecs comme le résultat de la réunion de quatre éléments (terre, eau, air et feu) sous l'effet de l'Amour qui unit et de la Haine qui désunit, avait été considérablement enrichie par les alchimistes avec l'introduction d'éléments de contrôle (sel, soufre, mercure) et les métaux¹. Le nombre de composants de la matière ne cesse

¹ Notons, en passant, que si la théorie des quatre composants fondamentaux, élaborée par Empédocle (« Connais premièrement la quadruple racine de toutes choses : Zeus aux feux lumineux, Héra mère de vie, et puis Aidônéus, Nestis enfin, aux pleurs dont les

de s'accroître suite aux travaux des chimistes (Boyle, Cavendish, Lavoisier) et à la renaissance de l'hypothèse atomiste de Démocrite par J. Dalton, jusqu'à la synthèse de D. Mendeleïev. La richesse expérimentale accumulée par les chimistes et les physiciens du 17^{ème} au 19^{ème} siècle sur la matière et ses transformations, organisée par la classification de Mendeleïev, ouvre la porte à une nouvelle synthèse encore plus puissante et plus unificatrice. Elle résulte de la découverte de la structure de l'atome par Rutherford et Bohr et du développement de la mécanique quantique au début du 20^{ème} siècle. Les composants fondamentaux de la matière sont réduits à trois « briques », le proton, le neutron et l'électron, à partir desquelles toute matière connue peut être construite.

Loin de marquer la « fin de l'histoire » pour la physique, ces découvertes entraînent d'autres découvertes : de nouvelles particules « élémentaires » qui, tout en étant « inutiles » à la constitution de la matière ordinaire, participent, ainsi que leurs antiparticules, aux processus élémentaires, c'est-à-dire ceux qui impliquent les composants élémentaires de la matière. La découverte du muon en 1936 jeta ainsi les physiciens dans un océan de perplexité : « Who ordered that ? » (« Mais qui a commandé ça ? »), s'exclama Isaac Rabi.

Après la deuxième guerre mondiale, le rythme de découverte de nouvelles particules « élémentaires » s'accroît, au point qu'en 1963 leur nombre atteint était de l'ordre de la centaine. Une nouvelle synthèse fut proposée par Murray Gell-Mann [3], qui identifia une symétrie, basée sur le groupe de symétrie $U(3)$ de la théorie des groupes, sous-jacente à toutes ces particules. Cette nouvelle synthèse postule l'existence de trois générateurs abstraits du groupe de symétrie, que Murray Gell-Mann appela quarks. Que cette interprétation ait été retenue plutôt qu'une autre relève d'un « miracle » de la physique moderne. Très rapidement, dans les années 1970, les quarks ont été découverts en tant qu'entités réelles avec les expériences de diffusion d'électrons sur des protons réalisées auprès de l'accélérateur linéaire de Standford (*Standford Linear Accelerator Centre*, SLAC [4]). La nature était redevenue « simple » : les particules « élémentaires » de matière étaient réduites à six quarks et six leptons auxquels il faut ajouter leurs antiparticules, toutes découvertes expérimentalement.

Le modèle des quarks, qui est aujourd'hui le fondement de notre compréhension de la matière, malgré son énorme succès expérimental et théorique introduit l'un des plus sérieux problèmes de la physique contemporaine : tous les efforts tentant d'observer des quarks « libres » ont été jusqu'à présent voués à l'échec. L'interaction entre quarks les confine dans les hadrons (particules constituées de quarks) : le quark isolé n'existe pas.

Le nœud du problème s'est ainsi déplacé des entités aux relations entre elles. Depuis la découverte de la nature composite de l'atome, la découverte de nouvelles particules a été accompagnée par celle des forces qui gouvernent leurs

mortels s'abreuvent. ») et adoptée par les philosophes grecs de la pensée dominante, imposée par les écoles de Platon et Socrate au détriment de la pensée matérialiste de l'école de Démocrite, a été depuis abandonnée, elle a néanmoins le mérite d'avoir identifié les quatre états thermodynamiques de la matière aux conditions ordinaires de pression et de température.

interactions. Trois d'entre elles (les interactions forte, électromagnétique et faible) sont désormais décrites par une théorie unique, baptisée « Modèle Standard de la Physique des Particules ». Cette théorie constitue la description la plus aboutie des phénomènes microscopiques. Même si la résolution des équations du Modèle Standard ne peut se faire analytiquement mais doit fait appel à des techniques d'approximation, les prédictions du modèle sont d'une précision redoutable et aucun phénomène observable n'est aujourd'hui en mesure de contredire ni le formalisme ni les fondements du modèle.

Malgré cela, la physique est entrée dans une crise grave, que certains considèrent même définitive pour la discipline [5]. En effet, le Modèle Standard laisse un certain nombre de questions sans réponse.

En premier lieu les deux édifices majeurs de la physique moderne que sont le Modèle Standard et la Relativité Générale, restent séparés, chacun cantonné dans son domaine d'application, la description de la nature microscopique pour l'un et celle de la nature macroscopique pour l'autre. Unifier ces deux descriptions en une théorie unique, la Théorie du Tout, est le rêve des physiciens, même s'il n'existe aucune preuve matérielle qui justifierait cette unification ou, autrement dit, aucune preuve de l'insuffisance d'un côté du Modèle Standard pour expliquer l'ensemble des phénomènes subatomiques, et d'un autre côté aucun « indice » réel pour nous inciter à étendre la relativité générale au delà de son domaine d'application. Tout cela peut sembler en contradiction avec le besoin d'unification. L'opinion de l'auteur est que l'aspiration à unifier les lois de la nature est identique à celle qui a conduit Ockham à inventer son fameux « rasoir » : « Les multiples ne doivent pas être utilisés sans nécessité » (« pluralitas non est ponenda sine necessitate ») [6]. Cette aspiration découle d'un souci d'économie intellectuelle qui confine à l'esthétisme. Néanmoins, toutes les synthèses scientifiques réussies ont été fécondes de nouveaux résultats et ont marqué une avancée importante de la connaissance.

Une différence majeure distingue les deux théories. La Relativité Générale d'une part est une théorie « indépendante de l'environnement » (en anglais *Background Independent*) [7], qui crée l'espace et le temps, alors que le Modèle Standard, et la mécanique quantique en générale, supposent l'existence a priori de l'espace et du temps. Cela est un défaut « formel » du modèle standard, qui sans diminuer son efficacité « pratique », le rend inapte comme candidat de la Théorie du Tout.

Un autre défaut majeur du Modèle Standard est la nécessité d'introduire un grand nombre de constantes numériques (19, sans compter 7 ou 8 constantes pour expliquer la masse des neutrinos) et de leur attribuer des valeurs arbitraires et non corrélées, dont en particulier les masses des particules élémentaires (fermions et bosons) et les constantes de couplage des forces. Idéalement, une théorie ne devrait faire appel à aucune constante « externe » mais les calculer à partir des principes premiers de la théorie.

De plus, la valeur de ces constantes s'étend sur une échelle énorme, ainsi 5 ordres de grandeur séparent la masse de l'électron de celle du quark top. Nous sommes dès lors confrontés à un vrai défi pour imaginer un mécanisme capable de briser la symétrie originelle de notre Univers quand toutes les particules avaient la même masse et toutes les interactions la même force, et de générer

une telle diversité de valeurs pour la masse des particules élémentaires ainsi que pour les constantes de couplage des interactions².

L'interrogation précédente peut être reformulée en se demandant si une théorie peut expliquer pourquoi la nature a attribué certaines valeurs plutôt que d'autres aux constantes fondamentales.

La réponse « anthropique » à cette question est que les valeurs des constantes sont celles qui ont généré un univers adapté pour la naissance de l'humanité qui, au cours de son évolution, les a mesurées. Une telle réponse est considérée comme peu satisfaisante par une grande partie des chercheurs, qui estiment qu'en dehors du banal truisme, elle ne peut constituer un principe « générateur » pour la compréhension de l'univers.

La cosmologie soulève quant à elle d'autres interrogations. Il existe de fortes indications que nous ne connaissons qu'une partie infime (de l'ordre de 4 %) de la matière qui nous entoure. Le mystère de la matière et de l'énergie noire reste pour l'instant entier et ne trouve pas d'explication scientifique.

Les fondements épistémologiques mêmes de la mécanique quantique posent problème. Une théorie qui nécessite la présence de l'observateur pour décrire la nature, par ailleurs sans définir précisément cet observateur, ne peut être totalement satisfaisante.

La situation est ainsi mûre pour l'émergence d'une nouvelle « synthèse ». La théorie des cordes proposée dans sa forme initiale par G. Veneziano en 1969 et développée dans sa forme actuelle par E. Witten dans les années '90 [8] est venue à point. Cette théorie avait débuté sous les auspices et avec les caractéristiques d'une synthèse réussie, tout comme d'autres dans le passé (Newton, Maxwell, Einstein, Gell-Mann, Weinberg, Salam et Glashow). La théorie décrit toutes les particules du Modèle Standard comme des états de vibration de « cordes » monodimensionnelles, ouvertes ou fermées. L'idée est d'une simplicité captivante, mais malgré le travail acharné de dizaines de théoriciens pendant vingt années, cette théorie n'a pas encore trouvé une formulation satisfaisante. De plus, la théorie des cordes est incapable de faire des prédictions, ce qui en fait une théorie irréfutable dans le sens poppérien du terme, contrairement aux grandes synthèses du passé, qui ont « réexpliqué » toutes les observations précédentes et prédit de nouvelles observations, offrant l'opportunité de les « réfuter » et ouvrant de nouveaux champs d'investigation.

Malgré ses atouts « esthétiques » et intellectuels, la théorie des cordes n'a pas encore tenu ses promesses, ce qui suscite des doutes légitimes sur sa capacité à les tenir.

Il faut peut-être rappeler ici que la théorie des cordes, bien que largement « majoritaire » auprès des théoriciens, n'est pas la seule théorie proposée pour réunifier le Modèle Standard avec la Relativité Générale. Pour ne citer que l'une d'entre elles, le modèle dit des géométries non commutatives élaboré par A. Connes [9] offre un espoir d'achever la grande unification des lois de la physique.

² Une constante de couplage est une quantité qui régit l'interaction entre deux entités physiques et elle exprime la notion intuitive de « force » de l'interaction.

Pour revenir au Modèle Standard, il existe une dernière particule dont l'existence a été postulée en 1964 et qui semble avoir été détectée en 2012 par les expériences ATLAS et CMS au LHC : le boson de Higgs [10]. Cette particule est d'une importance toute particulière parce que elle est censée être l'expression d'un champ qui, en interagissant avec les champs associés aux particules du Modèle Standard, brise la symétrie originelle et génère la masse des particules. Cette brisure de symétrie est décrite par le mécanisme de Brout-Englert-Higgs-Hagen-Guralnik-Kibble [11] introduit en 1964. Selon ce mécanisme, une symétrie locale peut être brisée spontanément, en introduisant un champ scalaire de valeur moyenne non nulle dans le vide. Ce mécanisme a été imaginé pour expliquer la masse non nulle de certains bosons de jauge ($W^\pm$, Z), observation a priori incompatible avec le Modèle Standard. Le concept de masse y est interprété comme l'effet d'une interaction entre ces bosons et le champ de Higgs. Dans le Modèle Standard, le mécanisme de Higgs briserait l'interaction électrofaible basée sur le groupe de jauge $SU(2) \times U(1)$ en conférant une masse non nulle à plusieurs de ces bosons de jauge et plus précisément les bosons W et Z , alors que le photon reste, lui, de masse nulle. Cela expliquerait donc la différence de portée entre l'interaction faible, transportée par les bosons W et Z (groupe de jauge $SU(2)$), qui est de l'ordre de 10^{-18} m, alors que l'interaction électromagnétique qui est transportée par le photon (groupe de jauge $U(1)$) est de portée infinie. Le mécanisme de Higgs serait également, de façon indirecte, la source de la masse non nulle des fermions.

Tous ces développements consacrent une attention de plus en plus importante à la structure du vide. Loin d'être simplement l'état d'absence de matière, le vide semble avoir une structure propre, qui préfigure et façonne les propriétés de la matière qu'il contient en puissance : « Le vide *sans matière* est instable. Il existe un état d'énergie minime qui est composé de cellules, chacune contenant une paire de gluons dans un état de singlet de couleur et de spin. [...] On peut parler d'un *vide liquide* », selon Gottfried et Weiskopf [12].

Finalement, un constat de la physique contemporaine fait consensus : le témoin est passé dans les mains des physiciens expérimentateurs. D'un côté la découverte du boson de Higgs, si elle est confirmée, scellerait le sort du Modèle Standard. D'un autre côté la découverte éventuelle de nouveaux phénomènes tracera les pistes à suivre pour les développements théoriques.

Depuis les années soixante, l'instrument de prédilection pour la physique des particules est l'accélérateur de particules. L'exploration de régions spatiales de plus en plus petites nécessite des longueurs d'onde de dimension comparable. Pour explorer le monde subatomique, voire subpartonique, seules les particules subatomiques de haute énergie permettent d'accéder aux longueurs d'onde adéquates selon la relation de de Broglie [13] associant l'impulsion de la particule et sa longueur d'onde.

Le plus puissant des accélérateurs jamais construits, le Grand Collisionneur Hadronique (Large Hadron Collider ou LHC [14]), construit au CERN (Genève, Suisse) permet de réaliser des collisions depuis la fin 2009.

L'exploration d'une nouvelle région d'énergie est rendue possible grâce aux collisions de protons accélérés 7 TeV ($1 \text{ TeV} = 10^{12} \text{ eV}$) réalisant une énergie de 14 TeV dans le centre de masse proton-proton. Cette énergie ne sera atteinte

qu'en 2014, après d'importants travaux d'amélioration de la machine prévus en 2012 et 2013. En attendant, l'énergie maximale sera limitée à 4 TeV (8 TeV dans le centre de masse proton-proton), ce qui représente toutefois une énergie 4 fois supérieure à celle du Tevatron, l'accélérateur du laboratoire Fermi de Chicago, qui, jusqu'en 2010 détenait, avec 1,96 TeV dans le centre de masse proton-antiproton, le record de la plus haute énergie.

Il n'est pas rare d'entendre dire qu'un accélérateur de particules est une « machine à remonter le temps », puisqu'il permet de créer des densités d'énergie proches de l'énergie d'excitation thermique moyenne de l'Univers au moment du Big Bang. Cela n'est pas entièrement correct, dans la mesure où les conditions d'une collision ne sont pas celles d'un « bain thermique » avec un grand nombre de composants.

T. D. Lee [15] a posé les fondements de la physique des ions lourds en 1975 : « Nous sommes aujourd'hui confrontés à deux mystères majeurs : celui des symétries manquantes et celui des quarks invisibles. La solution de ces deux mystères est probablement liée à la structure du vide. [...] Dans la plupart des expériences de Physique des Hautes Énergies, les dimensions observables sont d'autant plus petites que l'énergie est plus grande. Pour étudier la question du vide, une direction différente doit être suivie : étudier les phénomènes collectifs en distribuant une énergie élevée dans un grand volume. ».

Si l'on considère cette citation à la lumière de celle de Gottfried et Weiskopf, on voit que si l'augmentation de l'énergie nous permet d'appréhender les « détails » de la matière de plus en plus fins, elle concentre notre vision sur une seule des cellules de ce vide « liquide » dont parlent Gottfried et Weiskopf. Comme le dit Lee, pour pouvoir progresser dans notre compréhension du vide, il faut bien sûr une énergie élevée, mais il faut aussi pouvoir la distribuer dans un grand volume.

Réaliser des collisions d'ions lourds à haute énergie est le seul moyen qui permette d'étudier les phénomènes collectifs dans un domaine de densité et de température suffisamment élevées pour créer un état particulier de la matière dans lequel les quarks et les gluons sont « libérés » de leur confinement au sein des hadrons (le plasma de quarks et gluons, en anglais *Quark Gluon Plasma* ou *QGP*). Les collisions de noyaux lourds réalisées à RHIC (Relativistic Heavy Ion Collider, à Brookhaven, E.U.) ont permis de recueillir de solides indices de l'existence de cet état et d'établir ses propriétés. Cependant, les résultats indiquent que les propriétés du QGP de RHIC sont très différentes de celles prédites par la théorie. Le QGP de RHIC serait un liquide parfait plutôt que le gaz parfait attendu. Explorer le diagramme de phase de la matière nucléaire bien au-delà de la température critique et bien au-delà de la température atteinte au RHIC est l'objectif majeur du programme ions lourds au LHC.

D'un point de vue plus général, l'étude des collisions d'ions lourds revêt un intérêt qui va au-delà de l'existence et de l'étude du QGP. Si l'on se réfère au concept d'émergence [16], où les phénomènes complexes dans un ensemble de composants élémentaires apparaissent justement sous l'effet des interactions entre les composants, nous savons depuis longtemps qu'il est très difficile, voire impossible, de prévoir et caractériser les phénomènes complexes simplement en étudiant les propriétés des composants. Pour comprendre les propriétés de la matière, considérée comme un ensemble d'éléments, il est nécessaire d'étudier le

comportement de plusieurs éléments en interaction. De façon plus figurative, il est très difficile de prévoir la forme d'une fourmilière à partir de l'étude d'une fourmi isolée.

On peut également relever que les collisions entre ions lourds sont plus « efficaces » dans l'utilisation de l'énergie des faisceaux que ne le sont les collisions de protons. En effet, une collision entre protons utilise seulement au mieux un tiers de l'énergie du centre de masse, car ce ne sont pas les protons qui entrent en collision, mais deux des six quarks en présence. En revanche, dans une collision centrale d'ions lourds, plus de la moitié de l'énergie est « utilisée ». Pour comprendre cela, on peut se représenter une collision entre protons à haute énergie comme la rencontre de deux sacs de trois quarks où seul un quark de chaque proton participe à la collision. Les deux quarks « spectateurs » ou « blessés » continuent leur course sans participer à la collision. Intuitivement seulement un tiers de l'énergie est utilisé dans le choc. Ceci est différent dans les collisions d'ions lourds, parce que les quarks blessés se trouvent au milieu des autres particules qui participent à la collision et peuvent interagir à nouveau avec elles, en déposant davantage d'énergie dans la « boule de feu »³.

Si l'énergie des composants des projectiles est assez élevée, ils peuvent s'éloigner rapidement de la zone de collision. Dans le cas de RHIC, et encore plus dans le cas du LHC, cela contribue à la formation d'une zone où les particules produites par la collision ont « oublié » les nombres quantiques des faisceaux. Dans cette zone, les quarks sont excités à partir du vide et la création de quarks se fait par paires quark-antiquark : le contenu « net » de quarks est donc égal à zéro. C'est dans cette zone « sans baryons » (en anglais *baryon free zone*) que se forme, après thermalisation des constituants, la matière à densité d'énergie élevée proche des conditions du Big Bang. C'est une espèce de « vide matériel » qui peut être étudié à partir des mesures de l'état final de la collision.

C'est avec cet objectif que, depuis les années quatre-vingt, des accélérateurs toujours plus puissants ont été construits pour réaliser des chocs entre un grand nombre de composants élémentaires et des noyaux atomiques. Au début, les accélérateurs fournissaient des faisceaux d'ions bombardant une cible fixe. L'énergie dans le centre de masse, égale à la racine carrée de l'énergie du faisceau, ne dépassait pas une dizaine de GeV par nucléon. C'est seulement avec l'avènement de RHIC, le premier collisionneur d'ions lourds, que l'énergie dans le centre de masse pouvait atteindre jusqu'à 200 GeV par nucléon.

Le collisionneur LHC accélère également des ions lourds, notamment du Plomb, et permettra d'atteindre en 2014 une énergie maximale dans le centre de masse de 5,5 TeV par nucléon. Des quatre principales expériences au LHC, l'une est optimisée pour l'étude des collisions d'ions lourds. L'expérience ALICE a été

³ Tout cela n'est que très schématique. Les quarks ont une vitesse relative à l'intérieur du noyau et, comme la probabilité d'interaction varie avec l'énergie du centre de masse de deux quarks qui interagissent, l'énergie des quarks blessés est en général supérieure à deux tiers. En outre, il faut considérer aussi les collisions des gluons de la mer. Dans le cas des collisions d'ions lourds, le couple de quarks « blessés » a une probabilité réduite de ré-interagir (voir par exemple Zeit. *Für Phys C*, 52(4), 643-653, DOI: 10.1007/BF01562339).

conçue pour observer de façon optimale les collisions de noyaux de Plomb, tout en gardant un fort potentiel pour l'étude des collisions de protons.

Comme pour toutes les expériences de physique des hautes énergies, le traitement des données de l'expérience ALICE se fait sur une base statistique. Plusieurs millions, voire milliards de collisions entre particules ou noyaux sont enregistrés chaque année, générant plusieurs pétaoctets (10^{15} octets) de données. Le stockage, le traitement, la distribution et l'analyse de cette énorme quantité de données nécessitent une infrastructure informatique de grande envergure. Elle doit être adaptée, non seulement aux nécessités techniques spécifiques qu'impose le traitement de ces données, mais aussi à l'environnement matériel, social et culturel d'une collaboration en Physique des Hautes Énergies.

La complexité du problème est telle que chacune des expériences LHC a dédié un projet à la conception, la réalisation et l'exploitation de l'environnement informatique pour le traitement des données « hors-ligne » (en anglais *Offline*). Le traitement hors-ligne entre en jeu dès que les données ont été enregistrées⁴. Dans le cas de l'expérience ALICE, le projet Offline a impliqué pendant presque dix ans plusieurs dizaines de physiciens programmeurs et d'experts informatiques.

La description de l'évolution, de la structure et de la mise en exploitation de ce projet constitue le sujet du présent mémoire. L'intérêt de cette description réside dans le fait qu'il existe très peu de travaux dédiés à ce sujet, et cela pour des raisons essentiellement historiques.

La recherche scientifique après la deuxième guerre mondiale a toujours été dépendante de l'ordinateur, à tel point que ce sont les physiciens eux-mêmes qui ont développé les premiers ordinateurs. Néanmoins, les physiciens utilisent l'ordinateur comme un outil, alors que le développement théorique sur les ordinateurs est devenu la thématique d'une nouvelle discipline, l'informatique.

Lors de la conception d'une expérience de physique, il n'est pas rare que les besoins du cahier des charges dépassent l'offre technologique de l'industrie. Pour faire face à une telle situation, les laboratoires de recherche collaborent avec l'industrie pour un programme de recherche et développement commun. Ainsi, d'un côté, l'industrie sera amenée à satisfaire les exigences technologiques de l'expérience, et de l'autre l'industrie va acquérir un savoir-faire nouveau. Cette démarche est l'un des principaux mécanismes du « transfert de technologie » entre la recherche de pointe et l'industrie.

La situation de l'informatique pour la physique est différente. Seulement très rarement les physiciens peuvent compter sur l'industrie du logiciel pour des projets communs et déléguer à l'industrie la réalisation du produit final. Même la science de l'information et le génie logiciel ont du mal à fournir aux physiciens des outils « théoriques » adaptés à leurs besoins.

⁴ On verra plus tard que cette définition « classique » du logiciel hors-ligne en rapport avec le logiciel et le matériel « en-ligne » (en anglais *Online computing software and hardware*) est très schématique, la réalité est beaucoup plus nuancée.

Les raisons de cette situation sont à chercher dans l'environnement très particulier de la Physique des Hautes Énergies qui développe et utilise ses propres outils. Cette situation a conduit à l'émergence d'une nouvelle catégorie de chercheurs : l'informaticien – physicien. Ces chercheurs ont développé un savoir-faire et des outils de travail efficaces mais qui n'ont jamais été formalisés et sont donc inconnus de l'informatique « classique ».

Le manque de communication, voire la méfiance réciproque, est monnaie courante. Le physicien programmeur se considère comme un physicien qui doit, temporairement et par nécessité, même si le temporaire peut couvrir toute une carrière, s'occuper des outils informatiques pour les besoins de l'expérience. Très rarement, il se considère comme un expert informatique travaillant à la pointe du développement de cette discipline, même si cela est le cas. Les informaticiens, de leur côté, considèrent que les physiciens sont des utilisateurs de l'informatique qui pourraient avantageusement bénéficier des méthodes et outils mis au point par la recherche informatique, s'ils en étaient conscients.

Bien que la situation quelque peu caricaturale qui vient d'être décrite doive être nuancée pour tenir compte de l'existence de passerelles entre les deux disciplines et de la variété des attitudes personnelles, elle est assez représentative de la réalité.

Au cours des dernières années, cette situation a pris un tournant assez intéressant. Les conditions du marché des produits informatiques se sont rapprochées par certains aspects de celles des expériences de physique, et on assiste à un processus de formalisation de techniques et concepts très proches de ceux utilisés depuis longtemps par les physiciens, mais qui n'avaient jamais été décrits ni même imaginés en termes d'entités conceptuelles de la discipline informatique.

Cet exposé a pour ambition de fournir une réflexion sur ces sujets ainsi qu'une contribution à ce processus de la part de l'auteur qui a été pendant vingt-huit ans, au CERN, au cœur des développements informatiques pour la Physique des Hautes Énergies.

Le présent document s'articule autour de sept chapitres. Dans le premier chapitre, nous présentons les objectifs du programme de physique de l'expérience ALICE. Dans le deuxième chapitre, nous décrivons le détecteur de l'expérience ALICE ainsi que son modèle de calcul. Le troisième chapitre est dédié au développement de l'environnement informatique de l'expérience ALICE, et en particulier à l'infrastructure du logiciel AliRoot pour le traitement des données. Le quatrième chapitre décrit dans le détail la structure du code AliRoot et de ses différents composants. Le cinquième chapitre expose la structure de la Grille informatique développée par l'expérience ALICE et utilisée pour le traitement distribué des données. Le sixième chapitre présente les premiers résultats obtenus grâce au logiciel AliRoot et à la Grille à partir des données récoltées entre la fin 2009 et l'hiver 2012. Nous concluons dans le septième chapitre et discutons des perspectives pour l'infrastructure informatique de l'expérience et en général pour l'informatique pour la Physique des Hautes Énergies.

1 Le Programme de Physique du détecteur ALICE

1.1 Aperçu théorique

Le but de la physique des ions lourds est d'étendre le domaine d'application du Modèle Standard pour comprendre l'émergence de phénomènes collectifs et les propriétés macroscopiques, impliquant plusieurs degrés de liberté, à partir des lois microscopiques de la physique des particules. A cette fin, la physique des ions lourds étudie la matière nucléaire dans des conditions extrêmes de densité et de température.

Parmi les phénomènes collectifs prévus par le Modèle Standard, l'exemple le plus frappant est la transition de phase du champ quantique à une densité d'énergie ou température critique (voir Figure 2). Selon la théorie cosmologique du Big Bang, l'univers a évolué à partir d'un état initial de densité d'énergie extrêmement élevée vers l'état actuel lors de son expansion. En refroidissant, l'Univers a traversé une série de phases thermodynamiques suivant le Modèle Standard, qui sont brièvement décrites dans le paragraphe suivant. Les caractéristiques globales de notre Univers actuel, telles que l'asymétrie baryonique ou la structure à grande échelle de la distribution des galaxies, sont le résultat de ces transitions de phase.

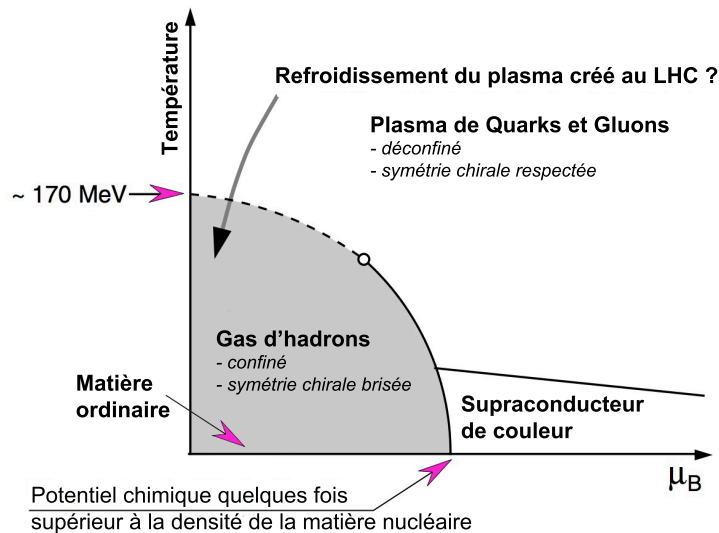


Figure 2 : Diagramme de phase de la matière QCD. Les lignes continues indiquent les transitions du premier ordre. La ligne pointillée indique une région où la transition est soudaine mais continue. Le point séparant les deux régions indique le point critique du second ordre.

Dans le contexte du Modèle Standard, la théorie physique qui rend compte de l'interaction entre quarks et gluons est la Chromodynamique Quantique (en anglais *Quantum Chromo Dynamics*, QCD). Proposée en 1973 par H.D. Politzer, F. Wilczek et D. Gross [17], la QCD est une théorie quantique des champs qui décrit l'interaction forte, une des interactions fondamentales de la matière.

La théorie de la QCD possède deux caractéristiques qui sont nécessaires à la description des phénomènes relatifs à la force forte, mais qui rendent très difficiles les calculs mathématiques des grandeurs observables. Les équations de base de la théorie sont fortement non linéaires et la constante d'interaction entre composants sensibles à la force forte est inversement proportionnelle au logarithme de l'impulsion relative des particules qui interagissent. Cette deuxième caractéristique est dénommée « liberté asymptotique » et rend compte du « confinement », c'est-à-dire l'impossibilité expérimentale de détecter des quarks « isolés » (déconfinés) et le comportement de particules « libres » des quarks dans les diffusions très dures (en anglais *Deep Inelastic Scattering, DIS*).

Face à un système d'équations dont la solution analytique nous échappe, une possibilité souvent utilisée en mécanique quantique consiste à approximer le système qui nous intéresse avec un système que nous savons résoudre de façon analytique « perturbé » par une interaction que nous savons décrire mais pour laquelle nous ne disposons pas d'une solution analytique. La solution finale est donc une superposition d'états du système de base dictée par l'interaction « perturbatrice » en forme de somme d'une série. Cette méthode suppose que le système de départ ne soit pas trop différent du système final, et donc que la perturbation soit « petite ». Dans le cas contraire, la convergence de la série n'est plus assurée. De plus, cette approche est intrinsèquement linéaire, et donc, a fortiori applicable à une théorie non linéaire seulement si la constante de couplage à l'origine de la non-linéarité est petite et donc la non-linéarité est négligeable.

Dans le cas de QCD, cela est le cas seulement à hautes énergies, où la constante de couplage devient petite grâce au phénomène de liberté asymptotique. Aux basses énergies les séries ne convergent pas et des divergences apparaissent dans les calculs. Nous nous trouvons donc dans la situation inédite en physique d'une théorie qui est capable de décrire les situations extrêmes des collisions à très haute énergie, par le biais d'une approximation, mais qui n'est pas capable de fournir une description des états ordinaires de la matière, comme par exemple la structure du noyau d'un atome. Une autre façon de faire face à ces difficultés consiste à calculer les solutions de QCD seulement sur des nœuds discrets d'un réseau espace-temps au lieu du continuum de l'espace-temps. Cette technique s'appelle QCD sur réseau (en anglais *Lattice QCD*) [18]. Ainsi, au prix de la discrétisation d'une solution qui en réalité est continue, les divergences mathématiques de la théorie sont éliminées et l'étude quantitative des prédictions de la théorie est possible pour les phénomènes « non-perturbatifs » tels que le confinement et la formation d'états de la matière comme le plasma des quarks et gluons (QGP) dont nous parlerons plus loin.

Dans les conditions de très haute température et densité, la matière sujette à l'interaction forte est supposée être dans un état appelé plasma de quarks et gluons (QGP, en anglais *Quark Gluon Plasma*), dans lequel les degrés de liberté quarks et gluons sont libérés, avec l'apparition de propriétés très différentes de celles de la matière hadronique que nous trouvons habituellement autour de nous. De telles conditions ont prévalu dans l'Univers primordial pendant quelques microsecondes après sa formation.

L'époque « cosmologique » du QGP est effectivement cachée aux observations astronomiques à cause de l'évolution ultérieure de l'Univers. Le seul moyen d'étudier cet état fondamental de la matière est alors via la collision de noyaux lourds dans le laboratoire. Dans les collisions d'ions lourds aux énergies ultra-relativistes, la matière nucléaire est chauffée et comprimée jusqu'à des conditions bien au-delà de la transition de phase QCD, et la même matière qui remplissait l'Univers très jeune est générée pour un court instant.

Une transition de phase reflète une brisure d'une des symétries fondamentales dans la théorie. Au-dessous de la température critique de la QCD, la matière peut être décrite en termes de degrés de liberté hadroniques, avec les quarks et les gluons confinés dans des objets de couleur neutre. Lorsque la température augmente les hadrons perdent leur identité (transition de phase de déconfinement) et les quarks et les gluons ne sont plus liés dans des hadrons. La transition de phase de déconfinement résulte de la brisure de la symétrie Z_3 (symétrie exacte dans la limite d'une QCD de jauge pure) à haute température.

Les calculs numériques de QCD sur réseau prédisent que la transition entre le QGP et la matière hadronique est en fait une transition continue et non une transition de phase d'ordre fini. Ces calculs suggèrent également que les degrés effectifs de liberté de la QCD aux températures obtenues avec les collisions d'ions lourds au LHC peuvent ne pas être des quarks et des gluons nus, se propageant librement, mais plutôt des objets plus complexes. La question cruciale des degrés de liberté effectifs dans le QGP peut également être adressée par l'expérience, et à l'heure actuelle c'est un domaine d'intense activité.

Une deuxième transition de phase est reliée à la génération de la masse hadronique, due à la présence d'un condensat de quark-antiquark de masse non nulle dans le vide à basse température. Selon QCD, à hautes températures, la masse du condensat du vide tend vers zéro, et les masses des quarks prennent leurs valeurs nues lors d'une transition de phase chirale. Dans ce cas la valeur moyenne de la densité du condensat quark-antiquark du vide, c'est-à-dire le paramètre d'ordre de la transition chirale, présente une soudaine diminution au voisinage de la température critique. La raison sous-jacente de cette transition de phase est la restauration approximative de la symétrie chirale (symétrie exacte dans la limite d'une masse des quarks nus nulle) du lagrangien QCD.

Les principaux objectifs du programme scientifique de l'expérience ALICE sont justement la confirmation expérimentale de ces transitions de phase, la vérification des prédictions de la QCD sur réseau reflétant les symétries fondamentales de la théorie ainsi qu'une étude détaillée des propriétés de la matière interagissant par le biais de l'interaction forte à haute température. La détermination précise des propriétés du QGP, y compris la température critique, les degrés de liberté, la vitesse du son dans la matière, et, en général, les coefficients de transport et l'équation d'état, est le but ultime de ce domaine scientifique. Ces études nous conduiront vers une meilleure compréhension de la chromodynamique quantique comme véritable théorie d'un problème à N corps. En outre, de telles mesures éclaireront la question complexe du déconfinement et la restauration de la symétrie chirale.

Avant le démarrage du programme d'ions lourds du LHC, la nature du QGP comme un liquide quasi parfait non visqueux avait émergé des résultats des

expériences au SPS du CERN et au RHIC de BNL. Grâce aux deux premières années de fonctionnement du LHC, la collaboration ALICE a confirmé cette image élémentaire. En effet, d'un côté de la matière nucléaire y a été créée à des valeurs de température, de densité et de volume jamais atteintes précédemment, et d'un autre côté, pour toutes les sondes du QGP, les mesures dépassent en précision et en portée cinématique celles effectuées au cours de la dernière décennie. Pendant plusieurs années, la théorie prédisait que le QGP à haute température serait un gaz de quarks et gluons interagissant faiblement et dont les constituants pourraient parcourir entre les interactions des distances longues par rapport à la taille d'un proton, distance qui représente, à peu près, l'échelle du confinement. Les expériences d'ions lourds indiquèrent au contraire un comportement fondamentalement différent et surprenant à savoir le comportement d'un plasma fortement couplé avec un très faible libre parcours moyen, qui présente un degré élevé de collectivité, et qui absorbe une fraction importante de partons de haute énergie le traversant. Les modes de quasi-particule dominants dans ce milieu sont différents des quarks et gluons nus interagissant avec une faible force de couplage. Nous présenterons quelque'un de ces résultats dans le Chapitre 6.

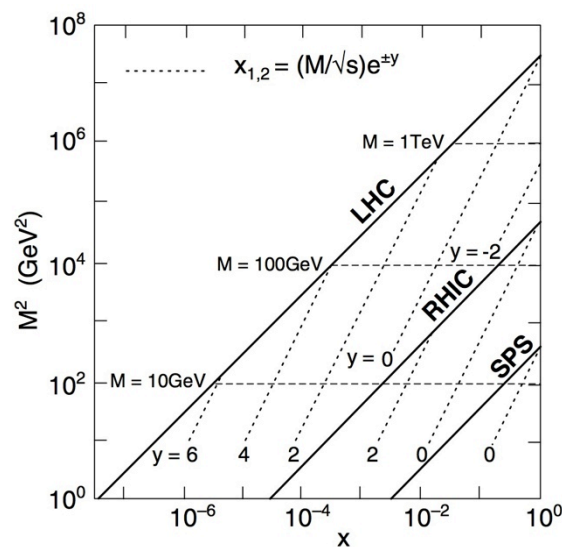


Figure 3: Le domaine des variables Bjorken- x et M^2 pour la production de particules atteignable dans les collisions noyau-noyau aux énergies du SPS, RHIC et LHC. M est la masse invariante de l'état final de la collision. Avec l'augmentation de l'énergie des faisceaux il est possible de créer des particules de masse plus élevée grâce à des collisions plus « dures » (petites valeurs de x).

Les calculs récents de la QCD sur réseau confirment cette image : la transition vers le QGP à potentiel baryo-chimique nul est continue, se produisant pendant un intervalle de température de 150-170 MeV. Une telle transition progressive est similaire à l'ionisation d'un plasma atomique, qui n'a pas de température de transition de phase nette. Cependant, la QCD sur réseau prévoit que même à une température de plusieurs fois la température de transition, la densité d'énergie est d'environ 15 % inférieure à celle prévue par la loi de Steffan-Boltzmann pour un gaz parfait de quarks et gluons sans interactions, avec une convergence vers

cette limite qui s'avère extrêmement lente. Cela indique également que les degrés de liberté effectifs du QGP à température finie ne sont pas les quarks et les gluons nus, mais plutôt des formations plus complexes dont la nature n'a pas encore été comprise. Cette question pose un sérieux défi pour l'expérience et la théorie, et il peut avoir une profonde connexion à d'autres domaines de la physique et de la cosmologie.

Le constat que le QGP est un liquide quasiment parfait est basé sur des mesures de spectres inclusifs (écoulement radial) et de l'anisotropie azimutale de la production de particules (écoulement elliptique) pour des hadrons identifiés à faible impulsion transversale (p_t), jusqu'à environ 3 GeV/c, et aussi sur la comparaison avec les calculs relatifs au modèle de l'hydrodynamique relativiste visqueuse⁵. Le bon accord des calculs hydrodynamiques avec les mesures d'écoulement de ALICE et d'autres expériences fournit une sérieuse évidence de la formation d'un QGP quasi équilibré dans les collisions au LHC, aussi bien que des outils puissants pour déterminer ses propriétés. Le très faible rapport entre la viscosité de cisaillement et la densité d'entropie du QGP, déduit de la comparaison entre les calculs basés sur le modèle et les mesure expérimentales, suggère un libre parcours moyen très court dans ce milieu composé de modes de quasi-particules fortement couplées, dont l'origine est l'objet d'études expérimentales ultérieures.

Le moyen principal pour sonder la nature des composants de la matière est de créer de nouveaux états à travers des collisions fortement inélastiques. On appelle une collision inélastique lorsque la structure interne des participants se trouve altérée à la suite du processus de diffusion. Si cette altération est profonde, c'est-à-dire si la cible se trouve dans un état final hautement excité, on parle de « diffusion dure » ou de « diffusion profondément inélastique » (en anglais *Deep Inelastic Scattering*, DIS).

En étudiant les diffusions dures entre électrons et protons dans les années soixante [19], Bjorken a découvert que leur section efficace pouvait être décrite à l'aide d'une seule variable qu'on appelle x de Bjorken (en anglais *Bjorken-x*) qui combine énergie et impulsion transférée pendant la collision. Cette variable exprime la fraction de l'impulsion du hadron portée par le composant élémentaire (quark ou gluon) qui participe à la diffusion. La découverte de cette invariance d'échelle (en anglais *scaling law*) a été d'une importance fondamentale dans la physique du vingtième siècle parce qu'elle apporte la preuve de l'existence à l'intérieur du hadron de centres de diffusion ponctuels. Cela a donné une existence réelle aux quarks qui jusque-là étaient considérés comme des éléments abstraits d'un système de symétrie élaboré pour classer les hadrons.

Les diffusions dures à haute x (~ 1) impliquent les composants « réels » des participants qu'on appelle normalement quarks de valence. Comme nous avons vu plus haut, ceux-ci se comportent dans ce cas comme des particules « libres » à l'intérieur du hadron (*asymptotic freedom*). En revanche, les diffusions à petit x

⁵ L'hydrodynamique est une approche très générale de la dynamique des fluides, décrivant le comportement à grande longueur d'onde d'un système complexe en quasi-équilibre.

(~ 0) impliquent les composantes « virtuelles », c'est-à-dire les gluons qui sont échangés entre les quarks de valence et les paires virtuelles de quarks qui se forment et se désintègrent en permanence grâce au principe d'indétermination de Heisenberg. On appelle ces derniers « la mer » (en anglais *the sea*) en relation à l'appellation donnée par Fermi.

ALICE est en mesure d'explorer la physique de l'interaction forte dans un nouveau domaine de très faibles valeurs de la variable Bjorken- x , comme on peut le voir dans la Figure 3. Dans cette figure les lignes diagonales en trait continu délimitent l'espace des valeurs de masse invariante et de x qui peut être exploré par les différents accélérateurs. Au LHC, cela permet l'exploration de l'évolution dans le temps de gluons près de la saturation de l'espace des phases. Dans ce domaine inexploré, la théorie prévoit des phénomènes de saturation qui pourraient donner naissance à un nouvel état de la matière, le condensat vitreux de couleurs (en anglais *Colour Glass Condensate*, CGC). Quand un noyau atomique voyage à une vitesse proche de la vitesse de la lumière, les gluons à l'intérieur du noyau apparaissent à un observateur comme un « mur de gluons ». Grâce à la contraction de Lorentz, ce mur de charges « colorées » a une grande densité (d'où le terme condensat). A l'intérieur de ce condensat les gluons ne modifient leur position que lentement en raison de la dilatation du temps. Ceci rappelle le comportement des silicates et du verre, qui sont des solides sur une échelle de temps courte, mais des liquides sur une échelle plus longue. Contrairement au plasma quark-gluon produit dans la collision, le condensat vitreux de couleur, est une propriété intrinsèque des particules qui ne peut être observée que dans des conditions de haute énergie telles que celles au RHIC ou au LHC. Le CGC est important car il est proposé comme une forme universelle de la matière à haute énergie, pour laquelle des méthodes de la chromodynamique classique ont été développées récemment et qui serait alors l'état précurseur du QGP.

Aux énergies du LHC, les processus de diffusion dure contribuent à la production de particules de façon plus importante qu'aux énergies des accélérateurs précédents. Ces processus, dont la section efficace de production peut être calculée exactement par les approximations perturbatives de la chromodynamique quantique, génèrent au tout début de la collision des sondes pénétrantes du milieu formé subséquemment dans la collision. En d'autres termes les collisions d'ions lourds auto-génèrent leurs propres sondes.

En particulier, les sondes dures à interaction faible (photons, mais aussi Z^0 et $W^\pm$) deviennent accessibles et peuvent être exploitées comme témoins directs du processus dans lequel elles ont été créées.

Le temps de vie du QGP à LHC est supérieure d'un ordre de grandeur à celui du milieu déconfiné créé à RHIC. Dans ces conditions, les caractéristiques collectives de la distribution des hadrons dans l'état final sont principalement déterminées par la dynamique des partons au sein du QGP. Il devient ainsi plus aisé de déterminer les propriétés du QGP à partir de l'observation de l'état final de la collision.

L'équation d'état de la matière nucléaire a pu être calculée à partir de la chromodynamique quantique et dans le secteur non perturbatif en utilisant des techniques de calcul sur réseau [20]. Le résultat d'un tel calcul est illustré par

l'équation d'état de la Figure 4 représentant la densité d'énergie normalisée à la puissance quatre de la température (ϵ/T^4) en fonction de la température.

Pour un gaz parfait, la densité d'énergie est égale au nombre de degrés de liberté effectifs multiplié par la quatrième puissance de la température. La forte augmentation près de la température critique (T_c , ligne verticale) indique que le système n'est pas seulement chauffé, mais qu'il se passe quelque chose de dramatique : il subit une transition de phase de la matière hadronique au plasma de quarks et gluons avec une augmentation correspondante du nombre de degrés de liberté. Au-dessus de T_c , le plasma de quarks et gluons est seulement chauffé et ϵ/T^4 reste constant.

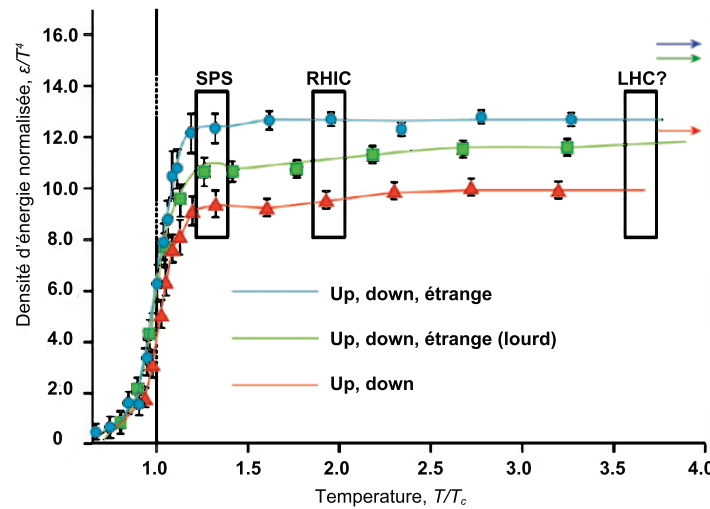


Figure 4 : Densité d'énergie quasto-chromodynamique Les trois lignes sont les prévisions pour deux saveurs de quarks légers (up et down, courbe rouge), trois saveurs légères (up, down et étrange, courbe bleue) et pour le cas plus réaliste de deux saveurs légères (up et down) et une plus massive (étrange, courbe verte). Les calculs ont été effectués dans un treillis $N_\tau = 4$ avec gauge améliorée et action calculée pour des fermions superposés. Les températures maximales initiales atteintes avec les différents accélérateurs sont indiquées. Les flèches de couleur sur l'ordonnée montrent la valeur de la limite de Stefan-Boltzmann pour un gaz idéal de quarks et gluons (Nucl. Phys. B 605, 579–599 (2001)).

On observe autour d'une température critique estimée à 175 ± 15 MeV, une augmentation rapide du nombre de degrés de liberté, la température restant quasi constante. Ceci est la signature d'une transition de phase avec une transition continue entre les phases. L'équation d'état est en général assez peu sensible aux valeurs des masses des quarks. Aux énergies du LHC, la thermodynamique du QGP devrait être dans un régime des trois saveurs. Ce qui est intéressant dans cette prédiction, c'est la rapidité du changement de la densité critique à l'approche de T_c . Si d'une part ceci amène à de grandes incertitudes dans la détermination de ϵ_c , ce changement rapide est aussi à l'origine d'importantes fluctuations qui pourraient être détectées expérimentalement comme signal de la transition de phase dans une analyse événement par événement.

Un autre effet prévu par les calculs QCD sur réseau est la diminution drastique de l'énergie libre dite aussi potentiel des quarks lourds. A une température voisine de la température critique, $T \approx 0,9 T_c$, la différence d'énergie entre deux quarks lourds à une distance égale à un femto mètre (10^{-15}m) et l'infini n'est que de 500 MeV. On peut donc s'attendre à une dissociation des états liés du charmonium $c\bar{c}$ pour des températures proches de T_c . La disparition des états liés pour les quarks légers autour et au-delà de T_c pourrait résulter d'une restauration partielle de la symétrie chirale, au moins pour les quarks légers, ce qui est maintenant confirmé par les études des fonctions spectrales [21].

Pour des températures plus élevées, $\geq 5 T_c$, les calculs de QGP sur réseau deviennent impossibles. En considérant qu'à ces températures le QGP est un gaz de gluons et de quarks et que la symétrie chirale est restaurée, les petites valeurs de la constante de couplage de l'interaction forte ($\alpha_s < 0,3$ à $T > 300$ MeV) permettent d'appliquer les techniques perturbatives aux calculs QCD.

L'image du QGP qui émerge des calculs à faible couplage est celle d'un plasma ordinaire dans le régime ultra-relativiste, si l'on ignore les effets liés à la nature non abélienne des composants. Les différentes impulsions des constituants génèrent une échelle d'interactions et degrés de liberté qui peuvent être étudiés par des théories appropriés, qui peuvent aussi être étendues au régime non perturbatif.

1.2 Observables du détecteur ALICE dans les collisions des ions lourds

Dans cette section sont décrites les principales observables, accessibles à l'expérience ALICE. Par observables nous entendons les quantités physiques mesurables avec le détecteur ALICE et qui permettent d'atteindre les principaux objectifs du programme scientifique de l'expérience tels qu'ils sont présentés dans la section précédente.

La multiplicité des particules chargées. C'est l'observable « du premier jour » la plus importante. Mesurée comme la densité de particules chargées par unité de rapidité ($\rho_{ch}(y=0) = dN_{ch}/dy|_{y=0}$), cette observable est une caractéristique globale du milieu produit par la collision. Comme elle est reliée à la densité d'énergie, elle participe au calcul d'un certain nombre de paramètres collectifs de la collision. Du point de vue expérimental, elle est un indice de la précision avec laquelle les autres observables peuvent être mesurées. La production des particules étant dominée par la production de particules de faibles impulsions transversales, créées dans le cas de collisions entre ions lourds par les processus « mous », elle ne peut pas être calculée précisément par les méthodes perturbatives de la QCD. Dans le cas des collisions proton-proton toute la production de particules (98 %) est due à des processus durs, et sa dépendance à l'énergie peut être approximativement paramétrée comme suit :

$$\rho_{ch}(y=0) \approx 0.6 \ln(\sqrt{s}/1.88) \quad (s \text{ en GeV}). \quad (1)$$

Les résultats de RHIC suggèrent que le nombre de particules générées par participant dans une collision centrale est de 50 % supérieur dans les collisions ion-ion à celui dans les collisions proton-proton à énergie de collisions dans le

centre de masse nucléon-nucléon identique. Les données expérimentales existantes ne permettent qu'une détermination assez ambiguë de la dépendance de la multiplicité à l'énergie de la collision : $f(s) = : \sim \sqrt{s}^\alpha, \sim \ln(s)$ ou $\sim \ln^2(s)$ ce qui donnait pour la densité de particules chargées à mi rapidité au LHC une fourchette de prévision de 1000 – 3500, qui est en accord avec la valeur expérimentale déterminée, égale à environ 1500 [22] pour les collisions Plomb-Plomb à 2,76 TeV/N (voir Chapitre 6).

Distributions spectrales des particules. Les distributions spectrales des particules reflètent toutes les étapes de la collision, du processus de formation jusqu'à l'hadronisation en passant par la phase de refroidissement et expansion. Elles constituent par conséquent l'observable de prédilection pour étudier les propriétés de la « boule de feu » créée par la collision entre ions lourds ainsi que son évolution dynamique. La majorité des hadrons produits ont une énergie cinétique transversale $m_t - m_0 = \sqrt{m^2 + p_t^2} - m_0 \leq 2 \text{ GeV}$. La distribution de départ est la distribution invariante en impulsion :

$$E \frac{d^3 n}{d^3 p} = \int_{\sigma} f(x, p) p^\lambda d\sigma_\lambda. \quad (2)$$

où $f(x, p)$ est la distribution invariante de densité de probabilité des particules et σ une surface spatio-temporelle qui marque la séparation entre la zone d'expansion du QGP et la zone d'hadronisation où quarks et gluons cessent d'être déconfinés. Cette formule est normalement exprimée, après transformation, en termes du spectre d'impulsion transversale de chaque espèce i de particule, comme suit :

$$\begin{aligned} \frac{dN_i}{dy m_t dm_t d\varphi_p} &= \frac{2g_i}{(2\pi)^3} \sum_{n=1}^{\infty} (\mp)^{n+1} \int r dr d\varphi_s \tau_f \exp\left[n(\mu_i + \gamma_\perp \vec{v}_\perp \cdot \vec{p}_\perp)/T\right] \\ &\times \left[m_t K_1(n\beta_\perp) - (\vec{p}_t \cdot \vec{\nabla}_\perp \tau_f) K_0(n\beta_\perp) \right], \end{aligned} \quad (3)$$

avec $\mu_i(\vec{r})$ potentiel chimique en fonction de la distance de l'axe de collision, $\tau_f(\vec{r})$ et $T(\vec{r})$ le temps propre et la température au « gel chimique » (voir plus en bas), communes à toutes les particules et $K_i(z) = \int_0^\infty dy \cosh^i(y) e^{-z \cosh(y)}$ les fonctions de Bessel modifiées. La somme s'étend sur tous les termes de la statistique de Bose-Einstein (-) ou Fermi-Dirac (+). Le premier terme de cette formule correspond à l'approximation de Boltzmann pour la distribution thermique, qui s'avère suffisante dans la plupart des cas sauf pour les pions. Cette formule est rarement utilisée dans sa forme complète, d'autant plus qu'elle peut être approximée de plusieurs façons selon qu'on choisit de retenir une partie des observables qui figurent dans la formule et d'intégrer par rapport aux autres.

Les paramètres des modèles décrivant l'occupation de l'espace de phase peuvent être déduits de la connaissance détaillée de la distribution de l'impulsion des particules et de la composition chimique de l'état final de la collision : la température et le potentiel chimique, les coefficients de la vitesse de l'écoulement radial, des écoulements orienté et elliptique, les paramètres dimensionnels dérivés des corrélations des paires de particules, les fluctuations

événement par événement et les corrélations entre les impulsions et les multiplicités des différents espèces de hadrons.

Pour donner une idée très schématique du rôle de ces observables, on peut dire que l'asymétrie géométrique de la collision dans le cas des collisions non centrales est convertie en asymétrie en impulsions par le QGP, sous l'effet de l'interaction forte entre constituants. Ce phénomène génère l'écoulement elliptique, c'est-à-dire l'anisotropie de la distribution des impulsions des particules par rapport au plan de la réaction dans les collisions non centrales. En autres termes, l'anisotropie géométrique de l'état thermalisé initial (qui peut être un état où le degré de liberté est partonique) génère, au cours de l'évolution hydrodynamique du système, une anisotropie dans l'espace des impulsions, anisotropie qui se traduit par un écoulement elliptique. Certaines fluctuations événement par événement seraient aussi liées aux états initiaux de l'évolution du plasma. Ceci peut fournir des informations sur l'équation d'état du QGP. Par exemple si la transition de phase entre QGP et matière ordinaire est du premier ordre (présence de chaleur latente et coexistence de phases différentes pendant la transition) on peut s'attendre à des grandes fluctuations de la densité d'énergie. Par contre dans une transition du deuxième ordre (dite aussi de phase continue), on peut s'attendre à des fluctuations dynamiques moins importantes. Les rapports entre les abondances des différentes particules et donc leurs potentiels chimiques sont déterminés par la température et le potentiel chimique dans le cadre d'un modèle statistique, canonique ou grand canonique. Le moment où ces rapports sont fixés est appelé « gel chimique » (en anglais *chemical freezeout*). On observe expérimentalement que le gel chimique est établi immédiatement après l'hadronisation du QGP, quand les quarks et les gluons se recombinent en hadrons « incolores » et les couleurs redeviennent confinées [23]. Le fait expérimental que la température du gel chimique est très proche de la température critique de transition de phase n'a pour l'instant pas de solide explication théorique. Les spectres des différentes espèces de particules doivent leur forme aux phénomènes de diffusion quasi-élastique et résonant qui suivent le gel chimique jusqu'au « gel cinétique » quand les particules cessent d'interagir et acquièrent leur spectre final.

Dans le scénario décrit plus haut, les particules d'impulsion transversale élevée sont porteuses de l'information des collisions dures survenues dans les instants initiaux de la collision. Il en résulte un spectre dont la forme à petites impulsions transversales dépend de la masse de la particule. L'extrapolation de la forme du spectre à basse impulsion serait très aléatoire, si la sensibilité expérimentale pour la détection des particules de basse impulsion était imparfaite. Une excellente efficacité de détection à basses impulsions transversales est cruciale pour déterminer l'abondance des différents types de particules à partir de l'intégrale du spectre. Il faut aussi noter que la détermination de la température du QGP requiert un ajustement simultané des paramètres T (température de l'ensemble grand-canonique des particules) et $v_\perp$ (flux transversale) aux spectres des différentes espèces de particules, pour départager les valeurs des deux paramètres qui sont fortement corrélés pour chaque spectre pris singulièrement. La température de l'ensemble grand-canonique des particules peut aussi être déterminée à partir d'un ajustement des abondances relatives calculées dans le

cadre d'un modèle statistique. Dans ce cas, les paramètres d'ajustement T , μ et le volume (grand canonique) sont trouvés en déterminant la température d'un système hadronique en équilibre avec les potentiels chimiques mesurés et avec un certain nombre de conditions au contour, tels que étrangeté nette nulle et conservation du nombre baryonique. La différence entre ces deux déterminations de la température offre des indications sur la contribution des rediffusions quasi-élastiques au spectre ainsi que celle des processus microscopiques à l'équilibre chimique.

Pour les collisions périphériques, c'est-à-dire lorsque la distance entre les trajectoires des deux noyaux n'est pas nulle, la zone de recouvrement des deux noyaux en collision est asymétrique. L'efficacité avec laquelle cette asymétrie spatiale est transférée dans l'espace des impulsions dépend du degré de rediffusion. Si le système est à l'équilibre thermique, l'anisotropie des impulsions atteint la limite « hydrodynamique » de saturation (rediffusion infiniment rapide). La détermination des paramètres de l'écoulement elliptique pour chaque type de particule permet de déterminer la « rigidité » de l'équation d'état du QGP pendant les moments initiaux de son expansion. Pour comprendre comment l'écoulement est relié au QGP il faut prendre en compte que l'expansion hydrodynamique ne génère pas d'écoulement. En effet l'écoulement ne peut s'établir qu'au tout début tant qu'il y a une asymétrie géométrique et dans un système en équilibre thermique. Donc s'il y a écoulement, il provient d'une phase où le système était à une température au-delà de la température critique et où les partons sont le degré de liberté et en plus dans un état à l'équilibre. C'est donc une signature de la matière partonique initiale, ce qui constitue l'a priori de toutes les autres mesures

La décomposition de la composante en φ_s de l'équation (3) en série de Fourier permet d'identifier les paramètres des écoulements direct (v_1) et elliptique (v_2) :

$$\frac{dN(b)}{dy m_t dm_t d\varphi_p} = \frac{1}{2\pi} \frac{dN(b)}{dy m_t dm_t} \times \left(1 + 2v_1(p_t, y; b) \cos(\varphi_p) + 2v_2(p_t, y; b) \cos(2\varphi_p) + \dots \right). \quad (4)$$

Corrélation de particules. La caractéristique principale des collisions d'ions lourds est l'émergence d'un état collectif de la matière à l'équilibre thermique qui influe sur la corrélation entre les particules de l'état final. Les propriétés spatio-temporelles de la source émettrice des particules peuvent être déduites de la fonction de corrélation à l'aide des méthodes d'interférométrie et en tenant compte du caractère symétrique (statistique Bose-Einstein) ou antisymétrique (statistique Fermi-Dirac) de la fonction d'onde de l'état à deux particules, ainsi que des interactions dans l'état final. La fonction de corrélation entre deux particules d'impulsion-énergie p_1 et p_2 est définie comme suit :

$$C(p_1, p_2) = \frac{d^6 N}{dp_1^3 dp_2^3} \bigg/ \frac{d^3 N}{dp_1^3} \frac{d^3 N}{dp_2^3}, \quad (5)$$

où $d^6 N / dp_1^3 dp_2^3$ est la fréquence combinée de trouver la particule 1 dans l'état d'impulsion p_1 et la particule 2 dans l'état d'impulsion p_2 . La fonction de

corrélation est d'habitude représentée en fonction des variables $q = p_1 - p_2$ et $K = (p_1 + p_2)/2$. Notons que q est une quantité invariante étant la différence entre deux quadrivecteurs. Les dimensions spatiales Δx de la source sont déduites de la fonction de corrélation à partir de la relation d'incertitude d'Heisenberg $\Delta x \cdot q \sim \hbar$: l'inverse de la largeur de la fonction de corrélation en fonction de q donne une information sur les dimensions spatiales de la source. En indiquant avec λ l'intensité de la corrélation et avec R^2 le rayon *gaussien*, la fonction de corrélation peut être écrite comme :

$$C(K, q_{inv}) = \left((1 - \lambda) + \lambda A(q_{inv}) \right) \left[1 + \exp(-R_{inv}^2 q_{inv}^2) \right] D(q_{inv}). \quad (6)$$

Le facteur A est le facteur dû à l'interaction coulombienne intégrée sur une source sphérique et le facteur $D(q_{inv})$ exprime les corrélations à grande échelle, dues, par exemple, à la conservation de l'énergie et de l'impulsion. Cette méthode est connue comme interférométrie HBT (Hanbury-Brown et Twiss) et elle permet d'estimer les dimensions de la source. Le paramètre $\lambda(K)$ dépend de plusieurs effets physiques et expérimentaux comme les interactions dans l'état final, la désintégration des résonances, mais aussi la désintégration d'un condensat chiral désorienté⁶. De plus, la mesure combinée de la dépendance $R(K)$ et du spectre des particules permet de déterminer la température et le flux transversale qui sont corrélés différemment pour les deux observables.

Les interactions dans l'état final, correctement identifiées, peuvent également fournir des informations sur le temps de formation des hadrons.

Pour LHC, si la densité des hadrons au moment de l'hadronisation est une quantité universelle, on peut s'attendre à une augmentation des rayons de corrélation proportionnelle à $(dN_{ch}/dy)^{1/3}$, ce qui semble être effectivement le cas [24]. Les corrélations de spin pourraient fournir des renseignements additionnels, et une confirmation indépendante des corrélations déterminées avec d'autres méthodes.

Fluctuations. Un certain nombre de caractéristiques, telles que la susceptibilité du système, les propriétés dynamiques du plasma dans les états initiaux de son évolution, la présence de charges fractionnaires libres ou les propriétés globales telles que la capacité thermique, peuvent être étudiées à partir des fluctuations événement par événement autour de la valeur moyenne d'une variable [25]. La compréhension de l'importance des fluctuations est en cours de développement.

⁶ Le condensé chiral désorienté (en anglais *Disoriented Chiral Condensate*, DCC) est une portion de vide avec une rotation de la valeur de son paramètre d'ordre chiral. Le vide QCD contient un condensé de bosons de façon analogue au boson de Higgs dans la théorie électrofaible. Cette condensation est le résultat de la brisure spontanée de la symétrie chirale de la QCD $SU_L(2) \times SU_R(2) = O(4)$. L'excitation collective de ce condensé, ce sont les pions (bosons de Goldstone), et il se transforme comme un 4-vecteur. Si à l'intérieur de la « boule de feu » son orientation est différente, des régions de vide désorienté finiront par se désintégrer en vrai vide. Le produit de cette désintégration sera un flot cohérent d'un champ de pions avec les nombres quantiques du vide désorienté, notamment l'isospin. Ceci conduit à la signature la plus fondamentale pour la recherche de DCC, à savoir une fluctuation importante dans la fraction de pions neutres produits.

Jets. Lors de collisions entre protons ou ions lourds, les partons de l'état initial peuvent donner lieu à des collisions « dures » qui génèrent des produits à grande impulsion dans le plan transversale à la direction de la collision. Ces partons colorés hors couche de masse génèrent une gerbe partonique et finalement hadronisent en une gerbe de hadrons « incolores », qui garde la mémoire du parton initiale. Cette gerbe de hadrons observée dans un certain angle solide ($\Delta\vartheta$) pointant dans une certaine direction (η, φ) et avec une certaine énergie totale (E_t) est appelée *jet*. De façon très schématique, l'impulsion du jet donne des informations sur celle du parton initiale, et la structure du jet sur le processus de fragmentation et d'hadronisation générés par le parton.

Dans les collisions d'ions lourds, la fragmentation des jets est modifiée par rapport aux collisions proton-proton, suite à la diffusion du jet dans le milieu chaud et dense du plasma avant le processus de fragmentation. De la comparaison de la fragmentation des jets mesurée dans les collisions d'ions lourds avec celle mesurée dans les collisions proton-proton l'on pourra extraire des propriétés du milieu telles que sa densité de charges de couleur.

Aux énergies du LHC, les calculs QCD perturbatifs indiquent que dans les collisions d'ions lourds jusqu'à ~ 30 partons « durs », avec une énergie transversale supérieure à 100 GeV, sont produits par événement. De plus, 30 % de la section efficace inélastique peut être dus aux mini-jets, jets dont l'énergie est grande par rapport à la masse hadronique mais très inférieure à l'énergie de collision (~ 10 GeV pour LHC), suggérant que les processus « durs » contribuent de façon prépondérante à l'état final.

Les mesures calorimétriques offrent le moyen expérimental le plus courant pour identifier les jets, c'est-à-dire l'identification d'une certaine quantité d'énergie dans un angle solide limité. ALICE est dotée d'un calorimètre électromagnétique qui ne couvre que 22 % de l'acceptance du « baril »⁷ central, 25 % si nous ajoutons le spectromètre de photons à haute définition (PHOS). Ceci ne constitue pas pour autant une limitation sérieuse pour la performance du détecteur dans l'étude des jets. Il est en effet possible d'identifier les jets à l'aide des trajectographes de particules chargées, qui peuvent avantageusement remplacer un calorimètre hadronique.

Plusieurs observables peuvent donner des indications sur la dynamique des jets dans les collisions entre ions lourds, en particulier :

- La réduction de la production de particules à haut p_t qui traduit la perte d'énergie des partons traversant le milieu ;
- La différence de la perte d'énergie des hadrons selon que l'origine est un jet de quark ou un jet de gluon qui permet de contraindre davantage les modèles de perte d'énergie puisqu'il est prédit que les gluons perdent deux fois plus d'énergie que les quarks ;
- La dépendance du spectre hadronique avec la géométrie de la collision, qui permet de tester la dépendance en L^2 , où L est la distance parcourue dans le milieu, de la perte d'énergie ;

⁷ On appelle « baril » l'ensemble des détecteurs à l'intérieur de l'aimant solénoïdale avec symétrie cylindrique. Voir Section 2.3.

- La modification de la structure hadronique des jets inclusifs et des jets étiquettes par un photon direct ou par un Z^0 qui offre une contrainte plus sensible aux paramètres des modèles de perte de d'énergie ;
- Les modifications différenciées de la fonction de fragmentation selon que le jet résulte de la fragmentation d'un gluon, d'un quark léger ou d'un quark lourd qui permettront de contraindre les modèles de perte d'énergie.

Photons directs. Les photons de haute impulsion (au-delà d'une dizaine de GeV) sont une sonde privilégiée des collisions d'ions lourds. Produits dans les processus durs lors des premiers instants de la collision, du fait de leur faible couplage au milieu, ils sont vecteurs d'informations uniques du système quantique créé par la collision. Le milieu lui-même est source de photons directs : les photons thermiques avec une impulsion de l'ordre de la température du milieu rayonnés par la source chaude avec un spectre caractéristique du corps noir de la phase partonique et hadronique, les photons produits avec des impulsions intermédiaires par rayonnement de freinage des quarks durs traversant le milieu et ceux produits par interaction des partons créés par diffusion dure avec les partons du milieu thermalisé. L'identification des photons directs nécessite la soustraction de la contribution prédominante des photons de désintégration des mésons neutres. Pour identifier les photons créés par le milieu il est nécessaire de comparer les productions de photons dans les collisions proton-proton (pas d'émission thermique ni d'effets nucléaires), proton-noyau (pas d'émission thermique mais effets nucléaires) et noyau-noyau (émission thermique et effets nucléaires).

Dileptons. Les dileptons ou photons virtuels, dont les sources sont similaires à celles des photons directs, jouent un rôle similaire à ces derniers pour contraindre les équations décrivant la dynamique du système issu de la collision des ions lourds. Même si la production de dileptons est supprimée d'un facteur 1/137 par rapport à celle des photons, la masse du photon virtuel procure une source d'information supplémentaire sur la dynamique de l'excitation vectorielle du plasma.

Une autre source de dileptons est la désintégration des mésons vecteurs, des états du quarkonia et des bosons vecteurs de l'interaction faible Z^0 et $W^\pm$. Les analyses en masse invariante des dileptons, du fait de leur faible couplage avec le milieu hadronique, permettent de récolter des informations précises sur l'abondance et la dynamique de ces particules sur leur altération due au milieu en expansion, ainsi bien dans la phase partonique que dans la phase hadronique.

Production de quarks lourds et de quarkonia. L'étude de la production de quarks « lourds » permet de tester avec un degré de liberté supplémentaire le milieu formé dans les collisions d'ions lourds. Du fait de leur grande masse, ils ne peuvent être produits que dans les premières phases de la collision et ils sondent donc le milieu au moment de sa formation. L'énergie de liaison des états du quarkonia, c'est-à-dire des systèmes liés constitués par une paire quark-antiquark ($q\bar{q}$), est du même ordre de grandeur que la température critique du plasma. Il y a donc une grande probabilité de « dissolution » du système lié, due à l'interaction avec le milieu. Cette suppression des états du quarkonia peut être

compensée par la recombinaison de quarks lourds présents dans le QGP en nombre d'autant plus élevé que la température est plus élevée.

La première observable à mesurer est la section efficace totale de production des quarks lourds. Il n'y a pas encore une description théorique satisfaisante de ce processus, bien qu'il existe plusieurs calculs qui ont été effectués à différents niveaux d'approximation : ordre principal, près de l'ordre principal ou même au-delà. Indépendamment de l'ordre d'approximation, la section efficace peut être exprimée comme suit :

$$\hat{\sigma}_{ij}(\hat{s}, m_Q^2, \mu^2) = \frac{\alpha_s^2(\mu^2)}{m^2} \sum_{k=0}^{\infty} \left(4\pi \alpha_s(\mu^2) \right)^k \sum_{l=0}^k f_{ij}^{(kl)}(\eta) \ln^l \left(\frac{\mu^2}{m_Q^2} \right), \quad (7)$$

où $\eta = \hat{s}/4m_Q^2 - 1$ et μ est l'échelle de renormalisation. Pour $k = 0$ nous retrouvons l'approximation de Born $O(\alpha_s^2)$, et pour $k = 1$ l'approximation après l'ordre principal (en anglais *next to leading order, NLO*) $O(\alpha_s^3)$ et l'échelle de renormalisation entre dans la formule. Les calculs basés sur cette formule donnent une réduction de la production par rapport aux collisions proton-proton du 35 % pour $c\bar{c}$ et du 15 % pour $b\bar{b}$.

La perte d'énergie des quarks lourds dans le plasma n'est pas bien connue, et ceci va rendre encore plus difficile l'estimation des effets d'acceptation du détecteur, surtout dans les coupures sur le p_t minimum pour les leptons provenant de leur désintégration, dans la détermination du taux de production des quarks lourds.

Nous avons déjà parlé de la possible suppression du taux de production des quarkonia due à l'effet du milieu (écranage de couleur, en anglais *colour screening*). Au LHC les états de quarkonia ψ' , χ_c et J/ψ sont abondamment produits, ainsi que la famille du Υ . Pour estimer les modifications des taux de production de quarkonia dans les collisions d'ions lourds, il faut pouvoir estimer la production dans des collisions proton-proton, c'est-à-dire sans milieu. L'écran de couleur pourrait ne pas être le seul mécanisme de suppression des quarkonia, mais aussi la restauration de la symétrie chirale pourrait y contribuer pour les quarks plus lourds.

La génération d'états de quarkonium due à la réinteraction des composants dans la phase de plasma et hadronique devrait aussi être prise en considération. Ici encore les calculs ne sont pas en mesure de donner une réponse précise et une augmentation entre 20 et 60 du rapport J/ψ / *charme* par rapport aux collisions proton-proton est prévue.

La physique des collisions proton-proton avec ALICE. Il ressort de la discussion précédente que la plupart des observables doivent être mesurées d'abord dans les collisions proton-proton pour être comparées aux mesures réalisées dans les collisions entre ions lourds. Une déviation par rapport à une loi d'échelle sera la signature d'un comportement collectif propre aux collisions d'ions lourds. La mesure des observables dans le même détecteur permettra de réduire les biais systématiques. Le programme de physique de ALICE pour les collisions proton-proton est donc nécessaire mais est également intéressant en tant que tel, pour porter un regard expérimental nouveau sur plusieurs aspects et processus encore mal compris théoriquement. En effet, grâce à ses excellentes propriétés

trajectographiques et à son excellente résolution en impulsion de 100 MeV/c à 100 GeV/c, ALICE peut effectuer des mesures complémentaires à celles qui seront réalisées par les autres expériences du LHC dédiées à l'étude des phénomènes rares à très grande impulsion transversale. Il y a ainsi un programme d'études spécifiques de ALICE avec les collisions proton-proton motivé en particulier par l'accès aux très petites valeurs de Bjorken- x , un nouveau domaine encore inexploré.

La mesure de la multiplicité des particules chargées nous permet de vérifier la dépendance de la multiplicité avec l'énergie du centre de masse de la collision qui reste encore mal connue à haute énergie. De plus les énergies du LHC permettent de mesurer des événements de très haute multiplicité, jusqu'à $dN_{ch}/dy \approx 60-70$, c'est-à-dire dix fois la valeur moyenne. Plusieurs autres approches permettent d'étendre les études du Tevatron et d'apporter une nouvelle contribution aux problèmes ouverts dans l'interprétation théorique des données. On peut citer la production d'étrangeté, les transferts en rapidité du nombre baryonique, l'étude des corrélations, la production des saveurs lourdes, la diffusion multiple des partons et la production des jets et des photons directs. La physique des processus diffractifs est également accessible à ALICE grâce à la présence du spectromètre de muons situé à grande rapidité ($\eta \geq 4$) et au Calorimètre à Zéro Degré ($\eta \geq 5,1$).

Collisions proton-noyau. Comme nous l'avons déjà dit dans ce qui précède, l'étude des collisions proton-noyau est essentielle pour identifier les effets du milieu nucléaire froid sur les observables. Les collisions proton-noyau pourront également contribuer à l'étude des fonctions de densité de gluons dans les noyaux dans la région très mal connue des grandes valeurs de $Q^2 \geq 1$ GeV et petite valeur de x .

Avec l'augmentation du flux partonique dans les collisions à haute énergie au LHC, la probabilité que deux partons subissent un choc dans le même noyau « cible » devient non négligeable. Ces collisions doubles permettent de mesurer la matière nucléaire à deux positions différentes simultanément. En changeant le système hadronique « cible », il est possible non seulement d'étudier la variation des interactions doubles avec le « flux » de partons de la cible, mais aussi de distinguer les composantes longitudinale et transversale des effets de corrélation dans la structure nucléaire.

Collisions ultrapériphériques (UPC, en anglais Ultra Peripheral Collision). Au LHC les noyaux accélérés avec un facteur $\gamma \approx 2900$, génèrent un très fort champ électromagnétique. Les collisions dont le paramètre d'impact⁸ est supérieur au rayon des deux noyaux, peuvent générer des collisions photon-photon et photon-noyau dans un domaine d'énergie encore inexploré. Avec les collisions photon-photon sont étudiés la spectroscopie des mésons avec $C = +1$, les mésons exotiques tels que les boules de gluons (en anglais *glueballs*) et la production électrofaible, par exemple de paires $W^+ W^-$.

⁸ Le paramètre d'impact est défini comme la distance minimale entre les centres des deux noyaux au moment de l'impact. Le nombre des nucléons qui participent à une collision, et donc le nombre des particules produites sont inversement proportionnels au paramètre d'impact. Les collisions avec le plus petit paramètre d'impact sont dites « centrales ».

Dans les collisions photon-noyau on pourra étudier la production diffractive de mésons et même sonder les distributions de densité partonique à petit x .

Rayons cosmiques. Au delà de la contributions qui sera fournie à l'étude des rayons cosmiques par l'amélioration de la connaissance des interactions à très haute énergie, ALICE peut opérer comme détecteur de muons cosmiques. Les 29 mètres de croûte terrestre au-dessous de ALICE peuvent absorber toute la composante électromagnétique et hadronique de la gerbe générée par les rayons cosmiques, laissant passer seulement les muons avec énergie ≥ 15 GeV, avec une détermination précise de la direction et de l'impulsion des muons, grâce au large volume et aux caractéristiques trajectographiques de la chambre à projection temporelle (en anglais *Time Projection Chamber*, TPC). Ceci positionne ALICE entre les expériences sur le sol, qui sont affectées par la gerbe atmosphérique (KASCADE), et les expériences profondes pour lesquelles le seuil des muons est beaucoup plus élevé, de l'ordre du TeV (MACRO). L'étude des paquets de muons va être particulièrement intéressante grâce à la capacité de la TPC d'analyser les flux de particules à haute densité.

2 Le détecteur ALICE et son modèle de calcul

2.1 Les conditions expérimentales

Le LHC fonctionne avec un cycle annuel alternant plusieurs mois de collisions proton–proton (équivalent à un temps effectif de 10^7 s) et plusieurs semaines (équivalent à un temps effectif de 10^6 s) de collisions d'ions lourds. Un arrêt d'exploitation d'environ quatre mois complète le cycle. Depuis le démarrage du LHC, le programme de physique de ALICE est le suivant :

- Collisions proton-proton à $\sqrt{s} = 2,75, 7$ et 8 TeV en 2010, 2011 et 2012 ;
- Collisions Plomb-Plomb à $\sqrt{s_{NN}} = 2,75$ TeV fin 2010 et 2011 ;
- Collisions proton-Plomb à $\sqrt{s_{NN}} = 5,02$ TeV fin 2012 et début 2013 ;
- Amélioration de la machine en 2013-2014 en vue d'atteindre l'énergie nominale de 14 TeV pour les collisions proton-proton.

A partir du 2014 débutera le cycle d'exploitation nominal, à savoir :

- Collisions proton–proton à $\sqrt{s} = 14$ TeV de mars à septembre ;
- Collisions ions lourds à $\sqrt{s_{NN}} = 5,5$ TeV en octobre pour une ou deux années ;
 - Une période de collisions ions lourds sera utilisée pour réaliser des collisions proton-Plomb;
 - Une ou deux périodes de collisions ions lourds seront utilisées pour réaliser des collisions Argon-Argon, avec des collisions proton-proton à la même énergie.

La luminosité (taux d'interaction par seconde pour une section efficace d'un centimètre carré) des collisions Plomb-Plomb est limitée à la fois par les détecteurs de ALICE et l'accélérateur LHC. En particulier, le temps de dérive de la TPC de $88 \mu\text{s}$ impose une luminosité maximum de l'ordre de $10^{27}\text{s}^{-1}\text{cm}^{-2}$ si l'on veut éviter une trop grande accumulation d'événements empilés (collisions multiples lors d'un même croisement de faisceaux). Une autre limitation vient des chambres de déclenchement du spectromètre MUON, qui saturent à partir d'un taux de déclenchement de 100 Hz cm^{-2} , soit une luminosité de $4 \cdot 10^{28}\text{s}^{-1}\text{cm}^{-2}$. La limite imposée par l'accélérateur est de $0,5 \cdot 10^{27}\text{s}^{-1}\text{cm}^{-2}$, limite fixée par le réchauffement des tuyaux de faisceau résultant des interactions électromagnétiques des ions de Plomb. Les limites précédentes seront plus élevées pour les ions plus légers.

La luminosité moyenne des faisceaux injectés dans l'anneau du LHC est fonction de la durée du remplissage (en anglais *fill*) et de la durée de la mise en route de l'expérience (en anglais *setup*). Ainsi, pour une luminosité initiale à l'injection de $\mathcal{L}^0 = 10^{27}\text{s}^{-1}\text{cm}^{-2}$, une durée de remplissage de 3 heures et une mise en route de 20 minutes, la luminosité moyenne sature après 8 heures à $0,44 \mathcal{L}^0$, $0,35 \mathcal{L}^0$ et $0,29 \mathcal{L}^0$ pour des collisions dans une, deux ou trois expériences (voir Figure 5). Le

réglage du β^* peut « aplatis » la courbe de luminosité avec comme conséquence un temps de vie plus long et donc une luminosité intégrée accrue.

La luminosité nominale du LHC est de $10^{34} \text{ s}^{-1} \text{ cm}^{-2}$ pour les collisions proton-proton mais la valeur optimale pour ALICE, permettant de réduire l'empilement des événements à une valeur acceptable pour tous les détecteurs, est de $\mathcal{L}^0 \approx 10^{29} \text{ cm}^{-2} \text{ s}^{-1}$ avec une limite supérieure à $5 \cdot 10^{30} \text{ cm}^{-2} \text{ s}^{-1}$.

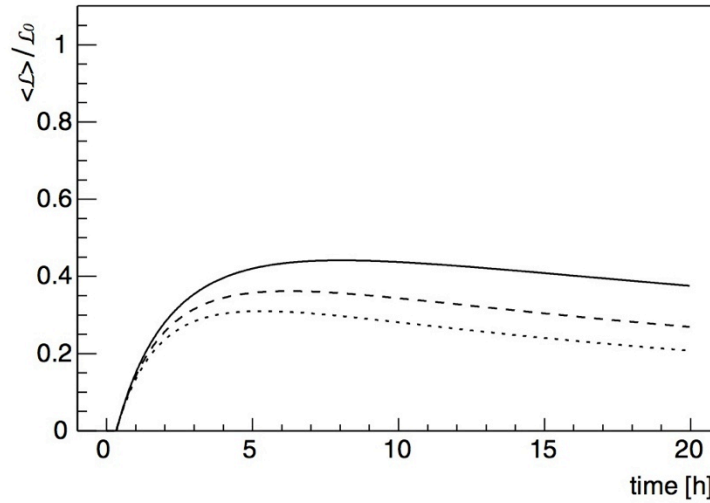


Figure 5 : Luminosité moyenne avec $\mathcal{L}^0 = 10^{27} \text{ cm}^{-2} \text{ s}^{-1}$ et $I_0 = 6,8 \times 10^7$ ions par paquet pour les collisions Plomb-Plomb avec un (trait plein), deux (ligne à tirets) et trois régions d'interaction (ligne pointillée).

Les principaux paramètres de la machine sont résumés dans la Table 1.

Le taux d'interactions est exprimé en fonction de la luminosité et de la section efficace : $R = \mathcal{L} \sigma$. La luminosité est définie, en principe, par les paramètres de l'accélérateur :

$$\mathcal{L} = f N_b \frac{N^2}{2\pi(\sigma_1^2 + \sigma_2^2)} F, \quad (8)$$

où f est la fréquence de révolution, N_b le nombre de particules dans un paquet, N le nombre de paquets dans chaque faisceau, $\sigma_{1,2}$ les dimensions des deux faisceaux au point d'interaction et F un facteur géométrique dû à l'angle de croisement qui, dans le cas de LHC, n'est pas nul. Il est évident que la précision

⁹ La luminosité est inversement proportionnelle à la surface de la zone d'intersection des faisceaux. Celle-ci est proportionnelle à l'émittance des faisceaux, c'est-à-dire à leur surface dans l'espace des phases et au profil des faisceaux au point d'interaction, qui est exprimé par la fonction $\beta(s)$, s étant la position des faisceaux le long de leur trajectoire. La valeur de cette fonction au point d'interaction est indiquée comme β^* . β^* est mesurée en unité de longueur et elle fournit une estimation de la dimension le long des faisceaux de la « zone lumineuse » où ont lieu les collisions. La dispersion des faisceaux double si on s'éloigne d'une distance de β^* du point d'interaction. Donc le nombre d'interactions par croisement des faisceaux est approximativement inversement proportionnel à la valeur de β^* .

avec laquelle les sections efficaces sont déterminées expérimentalement n'excédera pas la précision sur la luminosité. Elle n'est pas meilleure que $\approx 10\%$ du fait de l'indétermination de la dimension des faisceaux au point d'interaction. Une autre façon de mesurer la luminosité dérive du théorème optique :

$$\mathcal{L} = \frac{(1 + \rho^2)}{16\pi} \frac{R_{tot}^2}{(dR_{el}/dt)_{t=0}}, \quad (9)$$

où ρ est le rapport entre les parties réelle et imaginaire de la section efficace élastique. La section efficace totale est mesurée par une expérience dédiée, TOTEM [26]. La détermination de la section efficace inélastique mesurée par ALICE ($R_{el} = R_{tot} - R_{in}$) dépend de l'acceptance du détecteur estimée par des méthodes MonteCarlo. La dissociation électromagnétique peut également être utilisée pour évaluer la luminosité [27].

	proton- proton	Plomb- Plomb
Énergie par nucléon (TeV)	8	2,76
β^*	10	0,55
Valeur efficace du rayon du faisceau au point d'interaction : σ_t (μm)	16	71
Valeur efficace de la longueur du paquet : σ_l (cm)	7,7	7,7
Valeur efficace de la distribution du centre d'interaction : σ_v (cm)	4,3	5,4
Demi-angle de croisement vertical pour la polarité positive (négative) du dipôle du bras di-muon (μrad)	150 (150)	150 (100)
Nombre de paquets	2808	592
Espacement des paquets (μs)	24,95	99,8
Nombre initial de particules par paquet	$1,15 \times 10^{11}$	7×10^7
Luminosité initiale ($\text{cm}^{-2}\text{s}^{-1}$)	$\sim 5 \times 10^{30}$	10^{27}
Taux d'interaction (kHz)	350	8

Table 1 : Paramètres pour les faisceaux proton-proton et Plomb-Plomb pour ALICE au LHC. La grande valeur de σ_t pour les collisions proton-proton dépend de la valeur de β^* qui est modifiée pour ALICE pour réduire la luminosité. Pour les deux premières années d'opération, la valeur de l'énergie est la moitié de celle indiquée. Dans la période de mise au point du collisionneur le nombre de paquets est inférieur à celui indiqué.

Les premières mesures de sections efficaces des processus inélastiques et diffractifs des collisions proton-proton à 2,76 et 7 TeV sont maintenant en cours de publication [28]. La section efficace non élastique a été décomposée en trois termes :

$$\sigma_{NON-EL} = \sigma_{INEL} + \sigma_{DD} + \sigma_{SD}.$$

La mesure a été effectuée en trois temps. La première étape a été la détermination du rapport entre les événements à diffraction unique et double et les événements non diffractifs. La nature des événements a été déterminée par le nombre de traces dans les détecteurs « en avant » (VZERO et FMD) et l'intervalle de rapidité entre les traces. Le rapport trouvé expérimentalement a été corrigé pour l'efficacité de détection grâce à des simulations MonteCarlo avec deux générateurs d'événements différents (PYTHIA6 et PHOJET).

La détermination de la section efficace inélastique a été effectuée à partir de la formule :

$$\frac{dN}{dt} = A \times \sigma_{INEL} \times \mathcal{L}$$

où dN/dt est le taux d'interactions avec au moins une particule dans chaque bras du détecteur V0. La détermination de la luminosité comporte la mesure du flux des protons, qui est faite grâce à des bobines autour du faisceau, et la mesure du profil des faisceaux, qui est réalisée avec un balayage de Van der Meer. Le balayage de Van der Meer est une mesure du taux d'interaction enregistré lors d'un déplacement des faisceaux et elle permet, grâce à la formule (8), de déterminer le « profil » des deux faisceaux. La détermination du facteur A est liée à l'estimation de l'efficacité du détecteur, qui est faite à partir du MonteCarlo avec les deux différents générateurs. Une fois toutes les corrections appliquées, on obtient les résultats reportés dans la Table 2.

$\sqrt{s}$ (TeV)	$\sigma_{INEL}(mb)$	$\sigma_{SD}(mb)$	$\sigma_{DD}(mb)$
2,76	$6,28^{+2,4}_{-4,0}(model) \pm 1,2(lumi)$	$12,2^{+3,9}_{-5,3}(syst)$	$7,8 \pm 3,2(syst)$
7	$73,2^{+2,0}_{-4,6}(model) \pm 2,6(lumi)$	$14,9^{+3,4}_{-5,9}(syst)$	$9,0 \pm 2,6(syst)$

Table 2 : Valeurs des sections efficaces mesurées dans les collisions proton-proton. Les erreurs marquées *model* sont dues à l'utilisation de différentes simulations pour déterminer l'efficacité. L'incertitude marquée *lumi*, correspond à l'incertitude dans la détermination de la luminosité lors des balayages de van der Meer.

2.2 L'organisation hiérarchique de l'expérience ALICE

Ce n'est pas notre intention de donner une description détaillée de l'organisation interne de l'expérience ALICE, mais pour pouvoir comprendre plus aisément la suite de notre travail, il nous est apparu nécessaire d'en présenter ici les grandes lignes.

Le rôle institutionnel du CERN est de construire des accélérateurs de particules et de les mettre à la disposition de la communauté scientifique internationale pour y pouvoir conduire des expériences scientifiques de physique fondamentale. Les projets d'expériences sont présentés par des consortiums d'instituts de recherche et universités, les Collaborations Expérimentales ou simplement Collaborations. Ces projets font l'objet d'un processus d'acceptation par la communauté scientifique internationale avec la participation du CERN. Une fois leur projet accepté, les Collaborations prennent en charge le financement, la construction et l'exploitation du détecteur. Le CERN, et plus

particulièrement son Département de Physique, peut participer en tant qu'institut de recherche à l'une des expériences.

ALICE est le résultat d'une collaboration de plus de vingt ans et de près d'un millier de physiciens, ingénieurs et techniciens provenant de plus de cent instituts de recherche et universités situés dans plus de trente pays sur cinq continents. Plusieurs de ces institutions ont participé à la construction d'une ou plusieurs parties du détecteur, qui ont été ensuite envoyées au CERN pour y être intégrées au dispositif expérimental. Il est donc naturel qu'une organisation flexible mais forcément complexe ait été nécessaire pour gérer les aspects d'ingénierie, financiers et scientifiques d'un tel projet.

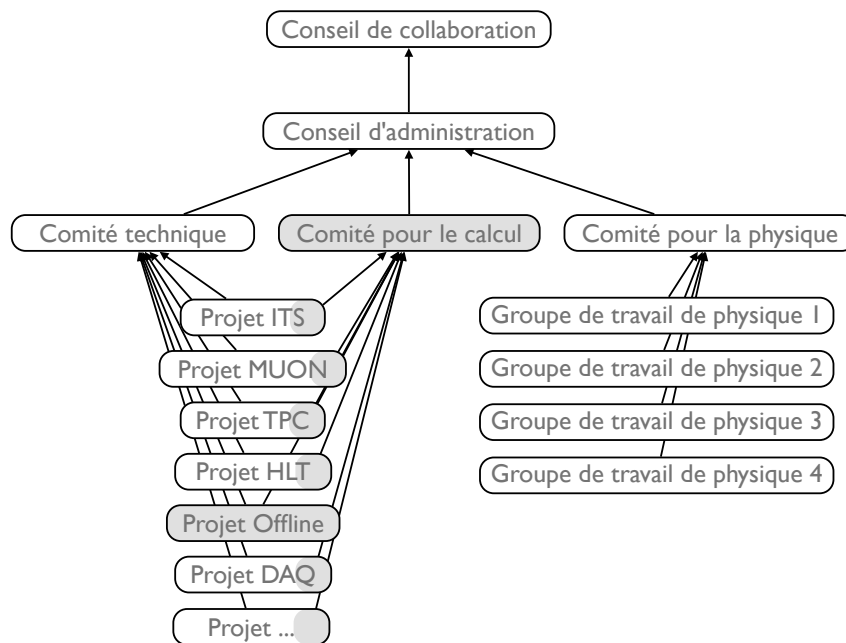


Figure 6 : Organisation de l'expérience ALICE.

La Figure 6 représente l'organisation de l'expérience avec un accent particulier mis sur l'aspect du calcul. Chaque sous-détecteur qui compose ALICE est géré par un projet spécifique. Les aspects d'infrastructure, d'aimants, d'interface avec l'accélérateur, d'alimentation en électricité et en différents gaz, d'assemblage du détecteur et de préparation et gestion du site sont traités par la « Coordination Technique » de l'expérience. Le calcul aussi est structuré comme un projet appelé le projet Offline. Son rôle est de produire l'infrastructure logicielle et de Grille informatique¹⁰ nécessaire pour la simulation, la reconstruction des événements et l'analyse des données. Tout cela va être décrit en détail aux Chapitres 4 et 5. Chaque projet est sous la responsabilité d'un ou plusieurs instituts, le CERN étant responsable du projet Offline. Tous les chefs de projet

¹⁰ La Grille informatique (en anglais, *Grid*) est une infrastructure virtuelle constituée d'un ensemble de ressources informatiques distribuées et potentiellement hétérogènes, qui sont amenées à travailler ensemble indépendamment de leur location géographique à la solution de problèmes de calcul.

participent au Comité Technique (en anglais *Technical Board*) qui est présidé par le coordinateur technique. Le travail d'analyse est coordonné par le Comité pour la Physique (en anglais *Physics Board*), présidé par le Coordinateur pour la Physique, et il s'appuie sur plusieurs sujets, chacun traité dans un Groupe de travail de Physique spécifique (en anglais *Physics Working Group*) dont les coordinateurs siègent au Comité pour la Physique. Les chefs de projet siègent au Comité d'Administration (en anglais *Management Board*), qui est présidé par le porte-parole de l'expérience. L'instance suprême de la collaboration est le Conseil de Collaboration (en anglais *Collaboration Board*), présidé par le Président du Conseil de Collaboration, où siègent les représentants des instituts qui participent à la Collaboration.

La spécificité du projet de calcul réside dans le fait que chaque sous-détecteur doit développer une partie du logiciel qui est intégrée dans l'infrastructure de calcul. Ainsi, le projet de calcul a non seulement une responsabilité de projet mais aussi une responsabilité de coordination de tous les projets. Dans ce sens il peut être considéré comme le pendant « virtuel » de la Coordination Technique. Il y a donc un Comité de Calcul (en anglais *Computing Board*) où siègent les responsables de la partie « logiciel hors-ligne » (Offline) de chaque projet. Le Comité du Calcul est présidé par un Président qui est un membre du Projet Offline.

Les organes de coordination et le conseil d'administration se réunissent chaque mois au cours de la même semaine (en anglais *ALICE mini Week*). Trois fois par année, les membres de la Collaboration sont conviés à se rencontrer au CERN pour une semaine de rencontres et réunions (en anglais *ALICE Week*) pendant laquelle se réunit le Conseil de la Collaboration. Avant chaque semaine ALICE, il y a une semaine de rencontres et de groupes de travail spécifiques pour l'Offline (en anglais *Offline Week*) où tous les membres de la collaboration impliqués dans l'Offline sont conviés.

2.3 Le détecteur ALICE

Le détecteur ALICE (Figure 7) est présenté en détail dans plusieurs documents techniques [29]. Nous n'en donnerons ici qu'une description très succincte.

ALICE est un détecteur universel (en anglais *general purpose*) optimisé pour la détection et l'identification des hadrons, leptons et photons émis à rapidité centrale ($y = 0.5 \ln((E + p_z)/(E - p_z))$, $-0,9 \leq y \leq +0,9$), et pour la détection et l'identification des muons émis à grande rapidité ($-4 \leq y \leq +2,4$). La partie centrale du détecteur, le « baril », est immergée dans un champ magnétique homogène et de faible intensité (0,5 T) généré par un solénoïde opérant à température ambiante. Un champ magnétique dipolaire est généré par un aimant conventionnel situé à grande rapidité en amont du bras di-muon. Le baril consiste en un certain nombre de détecteurs couvrant 2π en *azimut* et de haute granularité destinés à la trajectographie des particules chargées. En partant du point d'interaction et vers des rayons de plus en plus grands, se trouvent le système de trajectographie interne (en anglais *Inner Tracking System, ITS*) composé de trois doubles couches de détecteurs Silicium, puis la chambre à projection temporelle (en anglais *Time Projection Chamber, TPC*), puis le détecteur de radiation de transition (en anglais *Transition Radiation Detector,*

TRD), puis le détecteur du temps de vol (en anglais *Time Of Flight*, TOF), et finalement le spectromètre de photon (en anglais *PHOton Spectrometer*, PHOS, $|\eta| \leq 0,12$, 100° de couverture azimutale), le calorimètre électromagnétique (en anglais *ElectroMagnetic CALorimeter*, EMCAL, 110° de couverture azimutale) et le détecteur pour l'identification des particules à haute impulsion (en anglais *High Momentum Particle Identifier*, HMPID, $|\eta| \leq 0,6$, $57,6^\circ$ de couverture azimutale).

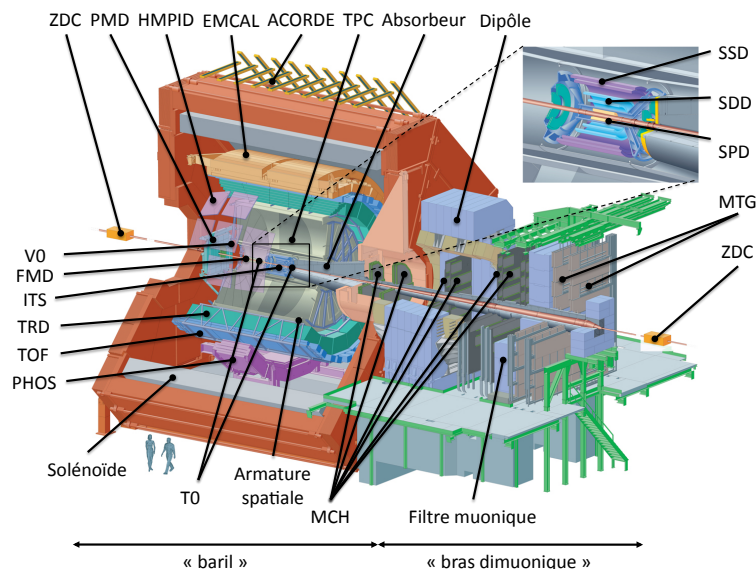


Figure 7 : Représentation du détecteur ALICE dans sa configuration de 2010. La signification des acronymes est donnée dans le texte.

Le baril est complété par un détecteur de rayons cosmiques situé au-dessus de la culasse de l'aimant (en anglais *A COsmic Ray DEtector*, ACORDE). Les détecteurs ITS, TPC, TRD, TOF et HMPID pèsent 77.500 kg et sont supportés par une armature (en anglais *space frame*) à treillis en acier inoxydable, très légère et robuste. Toujours à l'intérieur de l'aimant solénoïdal se trouvent d'autres détecteurs pour la mesure de la multiplicité des particules émises à grande rapidité : le détecteur de multiplicité de photons (en anglais *Photon Multiplicity Detector*, PMD), le détecteur de multiplicité vers l'avant (en anglais *Forward Multiplicity Detector*, FMD, $|\eta| \leq 5,1$), le détecteur du temps de collision (T0) et le détecteur de déclenchement pour les interactions de biais minimum (V0). A l'arrière du baril se trouvent le bras di-muon, constitué d'un absorbeur conique ne laissant passer que les muons avec un minimum d'interactions, six chambres de trajectographie (en anglais *Muon CHambers*, MCH, $-4 \leq \eta \leq 2,4$), un filtre en béton pour arrêter toutes les particules résiduelles sauf les muons, et deux chambres de déclenchement pour les muons (en anglais *Muon TRigger*, MTR). A l'entrée du bras di-muon, un aimant dipolaire génère un champ dipolaire de 0,67 T horizontal et perpendiculaire à l'axe du faisceau. ALICE est complétée par deux calorimètres à zéro degré (en anglais *Zero Degree Calorimeters*, ZDC) situés des deux côtés de ALICE à 115 mètres du point d'interaction à côté et entre les tuyaux de faisceaux. Une extension du calorimètre électromagnétique a été

installée au début de 2012 pour en étendre la couverture azimutale. Pour augmenter encore davantage cette couverture, le plan de mise à niveau de ALICE prévoit l'installation en 2013 d'un calorimètre électromagnétique (DCAL) additionnel installé dans la partie inférieure du détecteur, à côté de PHOS, avec une couverture de 60° en azimut et de 1,4 unités en pseudo-rapacité. Le rôle principal de DCAL sera de mesurer avec EMCAL les corrélations di-jet et hadron-jet.

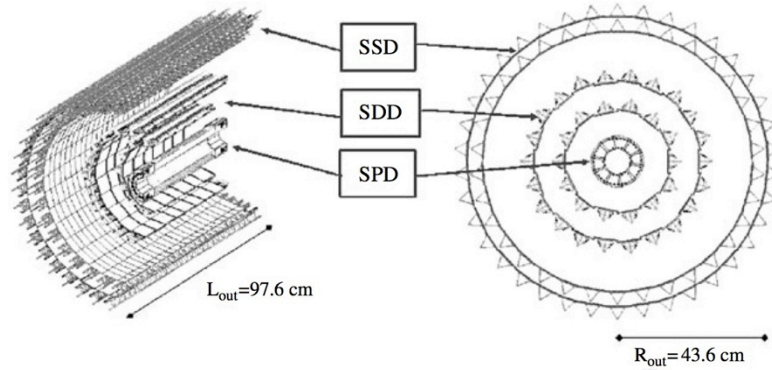


Figure 8 : Plan du système trajectographique interne.

La caractéristique principale du détecteur ALICE est sa capacité à reconstruire avec précision les trajectoires des particules chargées émises en grande quantité lors des collisions d'ions lourds, à rapidité centrale $-0,9 \leq y \leq 0,9$, et à identifier leur nature dans un domaine d'impulsion s'étendant de 100 MeV/c à 100 GeV/c. La couverture en rapidité est étendue vers les grandes valeurs de rapidité $-3,4 \leq y \leq 5,1$ grâce aux détecteurs FMD, V0 et T0. Lors de la conception du détecteur, les estimations sur la densité de particules chargées $dN_{ch}/dy|_{y=0}$, attendue au LHC étaient basées sur les valeurs mesurées au RHIC, soit 3500 particules chargées par unité de rapidité à rapidité centrale. ALICE a donc été optimisée pour cette valeur mais reste performante jusqu'à des densités de l'ordre de 8000 particules par unité de rapidité. La couverture en rapidité choisie est le résultat d'un compromis entre le coût et les besoins imposés par la physique. La valeur du champ solénoïdal est aussi issue d'un compromis. Du point de vue de la physique hadronique, un champ de 0,2 T garantit un seuil en impulsion suffisamment bas pour l'étude des phénomènes collectifs et des corrélations. Un champ plus élevé assure une meilleure résolution en impulsion, nécessaire pour une bonne identification des particules de la famille du γ reconstruites à partir de leurs deux électrons de décroissance et pour l'étude des jets. Vu que la section efficace de production de ces dernière observables est faible, ALICE prendra des données la plupart du temps avec un champ solénoïdal de 0,5 T, et pour un nombre limité de prises de données (en anglais *runs*) avec 0,2 T. La résolution pour les moyennes impulsions doit être de l'ordre de la masse des mésons vecteurs ϕ et ω (~ 700 MeV).

Très schématiquement, ALICE fonctionne avec deux configurations de lecture (en anglais *readout*) intercalées : soit tout le détecteur enregistre des données, soit seulement le spectromètre MUON et la couche de pixels du système de trajectographie centrale enregistrent des données. Le spectromètre MUON est capable d'enregistrer des données avec une fréquence plus élevée que les

détecteurs du baril central, et ainsi ce mode de fonctionnement permet de collecter un maximum d'événements possibles déclenchés par la désintégration des résonances à basse impulsion dans le canal $\mu^+\mu^-$. En effet le système d'acquisition des données de ALICE permet de combiner arbitrairement les détecteurs pendant la prise des données. La fréquence et la quantité des données récoltées sont finalement limitées par la bande passante pour l'enregistrement des données sur bande magnétique, à savoir 4 Go/s.

La détermination très précise de l'origine des traces au plus près du point d'interaction est essentielle pour l'identification du vertex de désintégration des résonances et des particules de saveur lourde. Cela est assuré par les six couches de silicium du ITS.

L'identification des particules π , K et protons avec un pouvoir de résolution supérieur à 3σ est nécessaire pour la mesure des corrélations à deux particules du type Hanbury-Brown et Twiss, pour l'identification des hypérons, des mésons vecteurs ($\phi \rightarrow K^+ K^-$) et des particules (mésons et baryons) de saveur lourde par la mesure du canal de désintégration hadronique, et pour la mesure de l'abondance de particules événement par événement. Cela est assuré pour les particules chargées avec impulsion transversale supérieure à 2,5 GeV grâce au baril central offrant une couverture azimutale de 2π . Pour les impulsions au-delà de 10 GeV/c les mêmes performances sont réalisées par le HMPID, couvrant cependant un angle solide plus réduit.

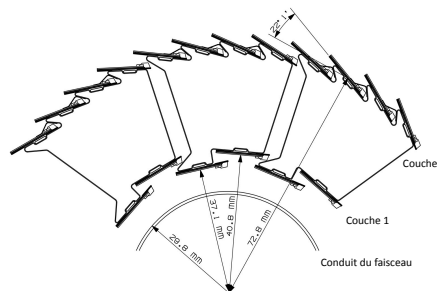


Figure 9: Structure du détecteur SPD.

L'étude des jets nécessite un détecteur déclencheur qui identifie un jet sur une base calorimétrique incluant les particules neutres. Cela permettra l'étude des jets jusqu'à des impulsions de l'ordre de 200 GeV/c.

La mesure des photons directs est conditionnée par la réjection des photons de désintégration des mésons neutres (π^0 et η), ce qui est assuré par la granularité élevée de PHOS et dans une mesure moindre de EMCAL.

Le spectromètre MUON a pour rôle de détecter et d'identifier les muons de décroissance des résonances des quarkonia, charmonium J/ψ and ψ' , et bottomonium Υ , Υ' and Υ'' jusqu'à une très basse impulsion transversale. La gamme de rapidité $-4,0 \leq y \leq -2,4$ a été choisie comme un compromis entre les exigences et le coût. Pour une spectrométrie complète, la résolution en masse doit être meilleure que 70 MeV à la masse du J/ψ (3 GeV) et 100 MeV à la masse du Υ (10 GeV). Pour préserver les chambres du grand flux de particules de bas p_t généré par les

collisions d'ions lourds, un absorbeur est intercalé entre le point d'interaction et les chambres de trajectographie.

Une des caractéristiques de l'expérience ALICE réside dans le fait que le détecteur réunit beaucoup de techniques de détection différentes. Dans la suite, nous donnerons une brève description de chaque sous-élément.

Le système de trajectographie interne (ITS, Figure 8). Les tâches de l'ITS sont la localisation du point d'interaction avec une résolution inférieure à $100\ \mu\text{m}$, la localisation du vertex de désintégration des hypérons et des mésons B et D, la reconstruction de la trajectoire et l'identification des particules d'impulsion inférieure à $100\ \text{MeV}/c$, l'amélioration de la résolution angulaire des particules à grande impulsion qui traversent la TPC et la reconstruction, avec une résolution limitée, des trajectoires des particules qui traversent les « zones mortes » de la TPC. Les deux premières couches de l'ITS vont fournir des données autant aux événements « complets » qu'aux événements enregistrés par le bras di-muon, pour donner une information précise sur le point d'interaction.

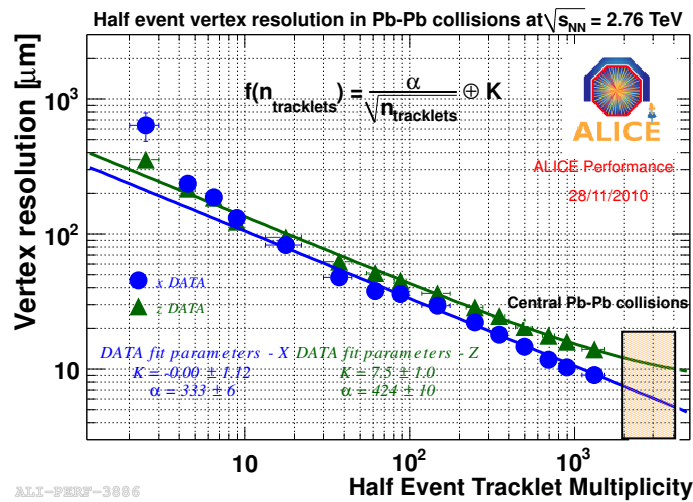


Figure 10 : Résolution de la mesure de la position du vertex déterminée à partir des données de l'ITS. La résolution est estimée en divisant aléatoirement les particules de l'événement en deux groupes et en calculant la différence entre les positions du vertex obtenues à partir des deux « sous-événements ». Les événements utilisés sont les collisions Plomb-Plomb à 2,76 TeV.

L'ITS a été conçu et réalisé dans le but d'accomplir ces tâches. L'acceptance est la même que celle des autres détecteurs du baril central $|\eta| \leq 0,9$, à l'exception de la première couche, pour laquelle $|\eta| \leq 0,98$, ce qui permet une meilleure précision pour la mesure de la multiplicité et offre une continuité avec les autres détecteurs à grande rapidité (FMD, MUON, T0). L'ITS a une longueur de 97,6 cm, un diamètre intérieur de 3,9 cm et un diamètre extérieur de 87,2 cm. L'identification des particules de basse impulsion est réalisée par la mesure de la perte d'énergie spécifique, dE/dx , dans les couches qui fournissent une mesure analogique. Un paramètre d'une importance particulière est la quantité de matériel de l'ITS (en termes de longueur de radiation) : ce budget matériel (en anglais *material budget*) doit être le plus petit possible, afin de minimiser les

perturbations induites sur la trajectoire des particules. Dans le cas de l'ITS, l'épaisseur moyenne est inférieure à 1 % de longueur de radiation, ce qui permet de mesurer l'impulsion des pions entre 100 MeV/c et 3 GeV/c avec une résolution en impulsion inférieure à 2 %. La nécessité de pouvoir fonctionner dans un environnement à haut flux de particules impose une granularité élevée des éléments sensibles, qui doivent être capables de mesurer simultanément jusqu'à 8.000 particules par unité de rapidité, c'est-à-dire environ 15.000 particules dans l'acceptance du système, ce qui à son tour requiert environ 13 millions de canaux électroniques. La résolution spatiale est un autre paramètre critique pour la mesure du vertex de désintégration des particules charmées et des résonances des quarks de saveurs lourdes. Pour cela la résolution spatiale a été particulièrement soignée, soit 12 μm en $r\phi$ pour la première couche de silicium.

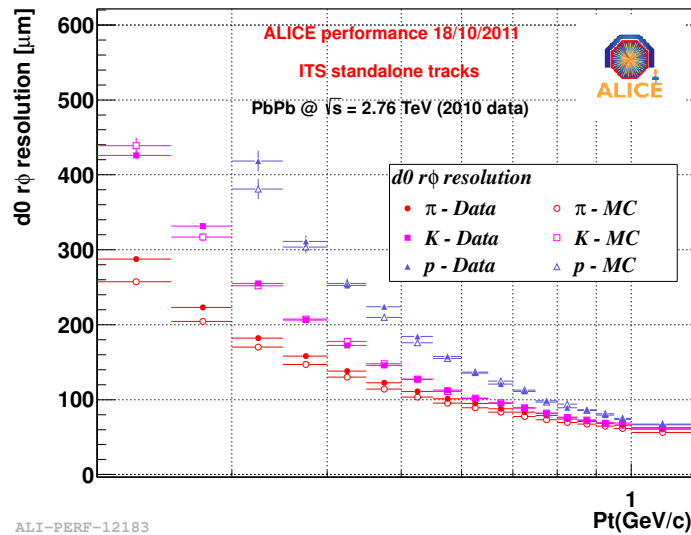


Figure 11 : Résolution du paramètre d'impact (distance du vertex de la ligne du faisceau) dans le plan perpendiculaire aux faisceaux pour les trajectoires reconstruites à partir du seul ITS pour des données réelles et des données simulées.

La partie la plus interne est un détecteur de silicium à pixels (en anglais *Silicon Pixel Detector*, SPD, Figure 9). Avec des rayons de 3,7 et 7,6 cm, une longueur de 28,2 cm et environ 10 millions de canaux électroniques, il est composé de deux couches de détecteurs à pixels de silicium en technologie CMOS (0,25 μm de taille de dispositif, en anglais *feature size*) soudés aux puces de lecture par « bump bonding » (soudure par point). Le choix d'un détecteur à pixels avec lecture binaire permet de détecter jusqu'à 80 particules par centimètre carré, de soutenir une dose de radiation de 0,22 kGy par année et d'offrir un taux de lecture jusqu'à 1 kHz. La dimension de chaque cellule de lecture est de 50 μm en $r\phi$ et 425 μm en z . La dimension du senseur (appelé *échelle*, en anglais *ladder*) est de 12,8 mm en $r\phi$ et 69,6 mm en z , avec une épaisseur de 200 μm . Il y a 80 échelles dans la première couche et 160 dans la deuxième, arrangées comme on peut le voir dans la Figure 9.

La puce de lecture est un circuit intégré développé pour une application spécifique (en anglais *Application Specific Integrated Circuit*, ASIC) programmable qui réunit les signaux de 8192 pixels. Une caractéristique de cette puce est de pouvoir produire un signal de « OU logique rapide » (en anglais *fast-OR*), si un des pixels est actif. Ce signal peut être lu par le système de déclenchement et il permet de sélectionner les événements à très basse ou très haute multiplicité, et il peut être utilisé comme déclencheur de niveau 0 (L0). Le signal « OU logique rapide » peut être délivré en 900 ns, et tout le SPD peut être lu en 256 μ s, avec 10 % de temps mort. Les données sont transférées au système d'acquisition par 20 liens de données des détecteurs (en anglais *Detector Data Link*, DDL).

La deuxième couche de l'ITS est le détecteur silicium à dérive, en anglais *Silicon Drift Detector*, SDD. Il est composé de deux couches avec des rayons de 15,0 et 23,9 cm et des longueurs de 44,4 et 59,4 cm. Il a un total d'environ 133.000 canaux électroniques pour 260 puces de silicium, chacune avec une surface sensible de 70,17 ($r\phi$) $\times$ 75,26 (z) mm² et 300 μ m d'épaisseur, arrangées sur deux couches (84 sur la première et 176 sur la deuxième). La surface sensible est divisée en deux régions de drift dans la direction $r\phi$ et 256 anodes de collection en direction de z . La vitesse de dérive est de 8,1 μ m/ns, et les signaux des anodes sont échantillonnés avec une fréquence de 40,08 MHz, ce qui donne une cellule de détection de 202 ($r\phi$) $\times$ 294 (z) μ m, pour une résolution de 35 ($r\phi$) $\times$ 25 (z) μ m. La résolution pour les traces doubles est en moyenne de 700 μ m avec une efficacité de 70 %. Des injecteurs de charge sont intégrés dans les puces pour contrôler la stabilité de la vitesse de dérive.

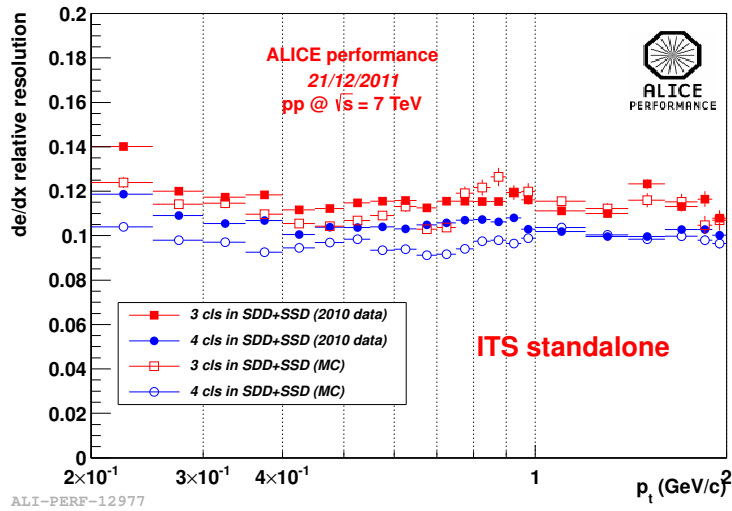


Figure 12 : Résolution relative de la perte d'énergie spécifique pour les traces reconstruites à partir des données du seul ITS en fonction de l'impulsion transversale. Les données utilisées proviennent des collisions proton-proton de 2010 et 2011.

Les puces ont une superposition de 580 μ m en $r\phi$ et z pour obtenir une acceptance totale pour le point d'interaction de $\pm\sigma = 10,6$ cm, et pour une impulsion transversale de $35 \text{ MeV}/c \leq p_t \leq 2,8 \text{ GeV}/c$. L'épaisseur du matériel est le 1,13 % de longueur de radiation, à laquelle il faut ajouter 0,29 % pour les

structures de support, les connections électriques et les systèmes de refroidissement (les « services », en anglais *services*). La lecture est effectuée par trois ASIC : PASCAL (embarqué sur la puce, pré-amplification, stockage analogique et numérisation, en anglais *Analogue to Digital Converter*, ADC), AMBRA (aussi embarqué sur la puce, tampon des données, égalisation de la ligne de base du signal anodique et compression non linéaire des données de 10 à 8 bits) et CARLOS (un pour 6 ou 8 puces, compression des données et suppression des données nulles) qui envoient les données à 12 DDLs.

Avec un taux d'interaction de 1700 Hz et un taux d'événements acceptés de 400 Hz, le SDD est occupé (en anglais *busy*) 10,5 % du temps. Pour des collisions Plomb-Plomb avec $dN_{ch}/d\eta|_{\eta=0} = 8000$ un taux d'occupation de 6,5 et 3,1 % a été simulé pour les deux couches.

La troisième composante de l'ITS est le détecteur silicium à pistes (en anglais *Silicon Strip Detector*, SSD). Il est composé de deux couches avec des rayons de 38,5 et 43,1 cm pour une longueur de 86,2 et 99,8 cm. L'élément de détection est une puce de $73 (r\phi) \times 40 (z)$ mm qui est couverte de chaque côté de 768 pistes de silicium $40 \text{ mm} \times 95 \mu\text{m}$, faisant un angle de 7,5 et 27,5 mrad avec la direction du faisceau, ce qui résulte en un angle *stéréo* de 35 mrad. Ce faible angle réduit les ambiguïtés de détection. Avec une précision de $20 \mu\text{m} (r\phi)$ et $820 \mu\text{m}$ la résolution pour les doubles traces est de $300 \mu\text{m} (r\phi)$ et $2400 \mu\text{m}$. Il y a 748 modules sur la première couche et 950 sur la deuxième, ce qui donne environ 2,6 millions de canaux électroniques analogiques. A chaque puce sont connectés 12 circuits ASIC HAL25 pour l'amplification, le formatage et le stockage du signal qui est envoyé aux circuits FEROM (Front-End ReadOut Modules) qui se trouvent en dehors du détecteur pour la numérisation du signal, et de là au système d'acquisition de données via 16 DDLs. Une fréquence d'événements acceptés de 1 kHz correspond à un taux d'occupation de 10 %.

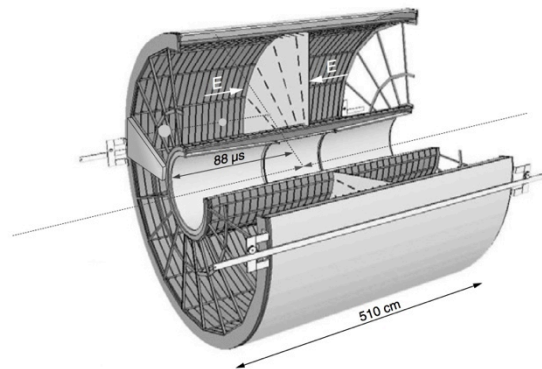


Figure 13 : Plan de la chambre à projection temporelle (TPC) de ALICE.

La résolution en x et z de la mesure de la position reconstruite du vertex de la collision primaire en fonction de la multiplicité des traces est représentée dans la Figure 10. L'estimation de la résolution est obtenue en divisant aléatoirement les traces de l'événement en deux échantillons et en reconstruisant les deux vertex pour les deux échantillons. La différence entre les positions de ces deux vertex est une mesure de la résolution. Un ajustement est effectué avec la fonction indiquée sur la figure. Le paramètre K est un facteur qui tend vers zéro. Le

coefficient α est corrélé à la résolution, qui est également fonction de l'inverse de la racine carrée du nombre de traces détectées dans l'ITS. Avec cette fonction, il nous est possible d'extrapoler la valeur de la résolution à la multiplicité mesurée pour les événements de la classe de centralité¹¹ 0-5 % (boîte orange).

La résolution de la mesure du paramètre d'impact dans le plan perpendiculaire aux faisceaux pour les trajectoires reconstruites à partir des données du seul ITS est représentée dans la Figure 11. La performance mesurée est très proche de celle obtenue à partir des données simulées. Dans cette figure nous pouvons voir la contribution individuelle de chacune des trois couches de l'ITS. La résolution est cependant essentiellement déterminée par la mesure de SPD. Les données utilisées sont les collisions Plomb-Plomb à 2,76 GeV du 2010 (période de prise de données LHC10h).

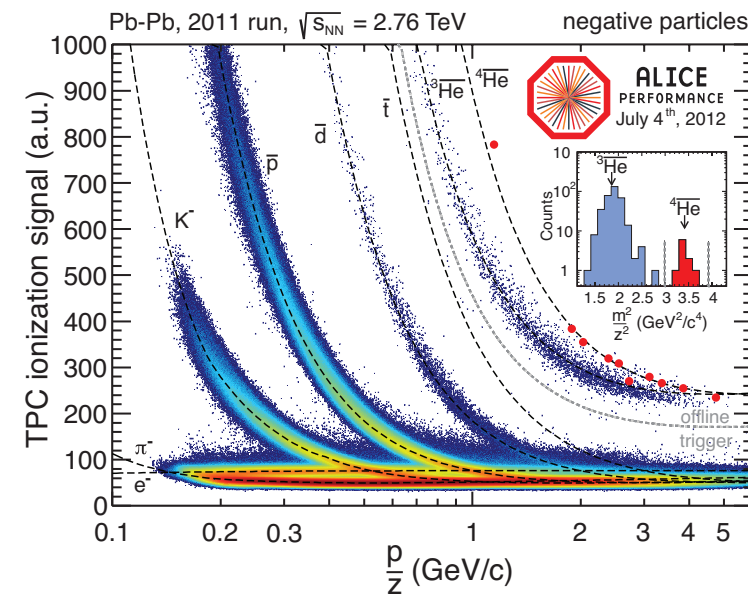


Figure 14 : Distribution de la perte d'énergie spécifique, mesurée par la TPC, des traces de tous les événements incluant au moins une particule identifiée avec une charge électrique supérieure à 1 en valeur absolue. Les collisions Plomb-Plomb à 2,76 TeV de l'année 2011 ont été utilisées. Seules les particules chargées négativement sont représentées. Les dix candidats de noyaux anti-alpha sont indiqués en rouge.

La résolution relative de la mesure de perte d'énergie spécifique dans les deux couches externes de l'ITS (SDD et SSD) pour les trajectoires reconstruites à partir des données du seul ITS est représentée dans la Figure 12 en fonction de l'impulsion transversale. Le fait que la résolution est indépendante de l'impulsion transversale considérée est un élément très important pour

¹¹ On appelle centralité, ou paramètre d'impact, la distance minime entre les trajectoires des centres des particules qui s'entrechoquent dans une collision. Si l'on imagine d'ordonner les collisions en fonction de leur centralité, et puis d'intégrer la distribution normalisée, nous avons une distribution de 0 à 100 %. Une classe de centralité est constituée par tous les événements compris entre deux valeurs de cet intégral. La classe de centralité 0-5 % contient les 5 % des événements à plus petit paramètre d'impact.

l'identification des particules de basse impulsion que le champ magnétique empêche d'atteindre la TPC.

La chambre à projection temporelle (en anglais *Time Project Chamber*, TPC, Figure 13) est le trajectographe principal de ALICE. Le volume actif cylindrique a un rayon interne de 78 cm et un rayon externe et 278 cm pour une longueur de 500 cm. Ce volume, rempli d'un mélange de gaz Ne(90 %)-CO₂(10 %), est divisé en deux zones de dérive par une électrode centrale en Mylar qui délimite un champ électrique de 400 V cm⁻¹, résultant en un temps maximum de dérive des électrons de 90 μs. Comme pour tous les détecteurs de ALICE, une grande attention a été portée sur la minimisation du budget matériel en utilisant des matériaux composites et des structures très légères. Le budget matériel total de la TPC est de moins de 5 % de longueur de radiation.

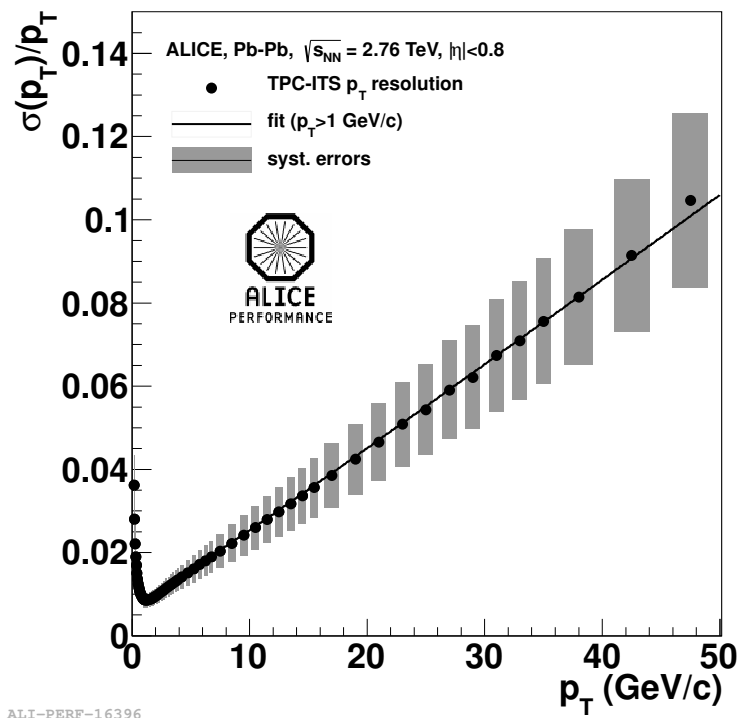


Figure 15: Résolution relative de la mesure de l'impulsion transversale obtenue avec la reconstruction des traces à partir des données combinées de la TPC et de l'ITS. Les événements utilisés proviennent de la deuxième reconstruction des collisions Plomb-Plomb à 2,76 TeV de 2010 (période de prise de données LHC10h).

Aux deux extrémités du cylindre, la TPC est équipée de chambres à fils proportionnelles installées dans 18 secteurs radiaux et couvrant 90 % du volume sensible. Dans chaque secteur, divisé en deux dans le sens du rayon (r 84,1-132,1 cm et 134,6-246,6 cm), sont installées une chambre interne et deux chambres externes à lecture par damiers. Il y a au total 560.000 damiers de détection ($4 \times 7,5$ mm² chambres internes, 6×10 et 6×15 mm² chambres externes) pour optimiser la reconnaissance des trajectoires en fonction de la variation de la densité des particules. Un système de laser est intégré dans la structure pour le calibrage. Cela permet une résolution moyenne en position de ~ 950 μm ($r\phi$) et ~ 1200 μm (z).

Les signaux provenant de 128 damiers sont récoltés par une carte frontale (en anglais *Front End Card*, FEC) qui contient les puces PASA et ALTRO. Les signaux sont amplifiés et mis en forme par le circuit intégré PASA. Ils sont ensuite transférés à la puce ALTRO qui les numérise et les échantillonne à 5-6 MHz. Cette puce, complètement programmable et capable de stocker jusqu'à 8 signaux, contient plusieurs circuits pour le traitement des signaux et notamment l'élimination des signaux nuls, la soustraction de la ligne de base, la suppression de la queue et la réduction du bruit. Jusqu'à 25 FEC sont connectés à une unité de contrôle de lecture (en anglais *Readout Control Unit*, RCU) qui prend en charge la connexion avec le système d'acquisition des données.

La TPC de ALICE est la plus grande jamais réalisée, et elle opère à des fréquences et densités de particules jamais atteintes auparavant. De plus, le programme de physique de ALICE impose plusieurs contraintes très importantes sur les performances de ce détecteur. Une résolution en impulsions à partir de 100 MeV/c de 1 % est nécessaire pour les observables hadroniques « molles ». Pour les électrons, la précision doit être de 1,5 % à 5 GeV/c pour discriminer les membres de la famille du γ en combinaison avec le TRD. La résolution se dégrade pour les impulsions élevées, mais à 100 GeV/c une résolution de 10 % est encore possible avec un champ magnétique de 0,5 T, en combinant les mesures de la TPC avec celles de l'ITS et du TRD. La TPC contribue à l'identification des particules en mesurant la perte d'énergie spécifique dans la région à basse impulsion ($\leq 1\text{ GeV}$) et dans la région de la montée relativiste (en anglais *relativistic rise*, $\geq 10\text{ GeV}$) où, avec une résolution de $\leq 7\%$, il est possible de distinguer, au moins sur une base statistique, les différentes particules et d'obtenir une bonne réjection e/π .

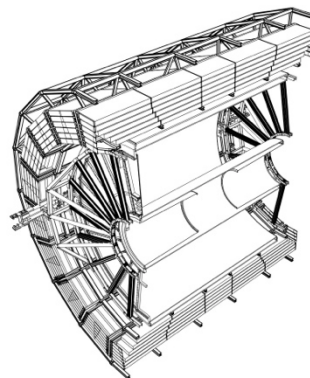


Figure 16 : Plan du détecteur à transition de radiation.

Dans la région des impulsions intermédiaires, il est nécessaire d'utiliser les informations provenant du TOF pour compléter l'identification des particules. Une excellente connexion des traces avec les autres détecteurs du baril, en particulier ITS et TRD qui lui sont limitrophes, est requise pour augmenter la résolution en impulsion : une amélioration d'un facteur cinq peut être atteinte pour les traces de 10 GeV/c avec l'aide des autres détecteurs. La couverture en rapidité est de $|\eta| \leq 0,9$ pour les traces qui sortent du cylindre extérieur, mais jusqu'à $\pm 1,5$ pour les traces qui traversent au moins 30 % de la longueur de la TPC et sortent par les côtés. Pour reconnaître et reconstruire la trajectoire des candidats électrons identifiés dans le TRD, la TPC doit être capable de

fonctionner à une fréquence de 200 Hz. C'est la fréquence de fonctionnement la plus élevée jamais utilisée pour une chambre à projection temporelle. L'expérience opérationnelle de ces deux années montre que les effets de charge d'espace restent acceptables, ce qui est en accord avec les résultats des simulations effectuées en phase de projet.

La distribution de la perte d'énergie spécifique, mesurée par la TPC, des traces de tous les événements incluant au moins une particule identifiée avec une charge électrique supérieure à 1 en valeur absolue est représentée dans la Figure 14. Seules les particules chargées négativement sont représentées. L'ensemble des événements collectés avec les collisions Plomb-Plomb à 2,76 TeV de l'année 2011 a été utilisé. Les différentes espèces de particules ont une perte d'énergie spécifique différente pour une même impulsion sur une large portion du spectre. Cela permet de les identifier en combinant les informations de la TPC et d'autres détecteurs. Dix candidats de particules anti-alpha, c'est-à-dire de noyaux d'anti-hélium, ont été identifiés.

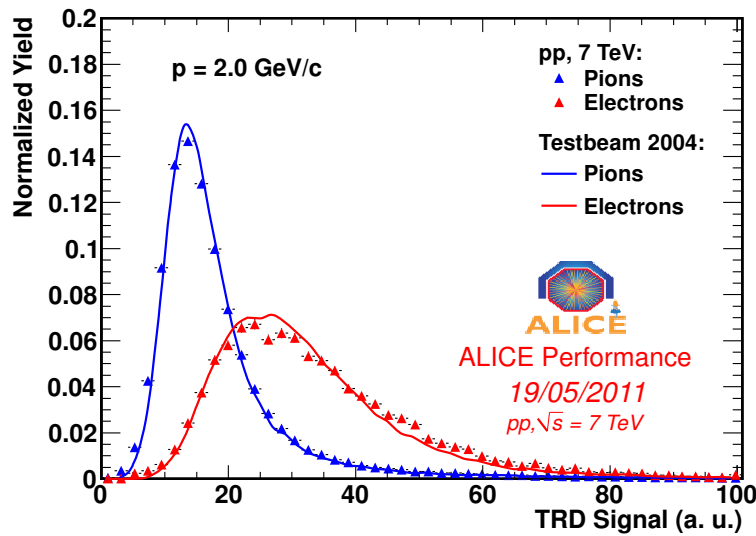


Figure 17 : Charge déposée dans le TRD par les électrons et les pions d'impulsion de 2 GeV/c. Pour les électrons, le dépôt de charge moyen est plus élevé en raison de la contribution supplémentaire de l'absorption du rayonnement de freinage (Bremsstrahlung) et du rayonnement de transition dans le gaz des chambres à dérive.

La résolution relative de la mesure de l'impulsion transversale est représentée dans la Figure 15 en fonction de l'impulsion transversale. Les événements utilisés proviennent de la deuxième reconstruction des collisions Plomb-Plomb à 2,76 TeV de 2010 (période de prise de données LHC10h). Seules les particules dont la trajectoire est entièrement contenue dans la TPC sont considérées ($|\eta| < 0,8$).

Le détecteur de radiation de transition (TRD, Figure 16) a pour but l'identification dans le baril des électrons avec $p_t \geq 1$ GeV/c, où la TPC n'est plus capable de les distinguer des pions. Ce détecteur permet à ALICE d'étudier les désintégrations des mésons vectoriels légers et lourds et le continuum di-leptonique. Il permet aussi, en conjonction avec la trajectographie de la TPC et la détermination du paramètre d'impact de l'ITS, d'identifier les électrons produits par la

désintégration des mésons de charme ou beauté ouverts et de distinguer les J/ψ prompts de ceux provenant de la désintégration du méson B. De plus, le TRD étant assez rapide, il peut être utilisé comme déclencheur pour enrichir les événements avec des événements incluant un γ émis dans le baril.

Le TRD occupe l'espace radial $2,9 \leq r \leq 3,7$ m pour une longueur de 7 m. Il y a 18 secteurs radiaux coïncidant avec les secteurs de la TPC. Chaque secteur comprend six couches avec, dans chaque couche, cinq modules en z pour un total de $18 \times 5 \times 6 = 540$ modules. Chaque module est une chambre multi-proportionnelle à fils remplie de Xe/CO₂ (85 %/15 %) avec un radiateur de 4,8 cm d'épaisseur, et de 12 à 16 rangées de damiers dans la direction z, chaque rangée avec 144 damiers en $r\phi$ pour une dimension typique de damier de $0,7 \times 8,8 \text{ cm}^{-2}$ et un totale de $1,16 \times 10^6$ damiers. La résolution en $r\phi$ est $\leq 600 \mu\text{m}$ et dépend de la multiplicité et de l'impulsion. La résolution est de $[2,5 \oplus (0,5-0,8) \cdot p_t \text{ (GeV)}] \%$ selon la multiplicité.

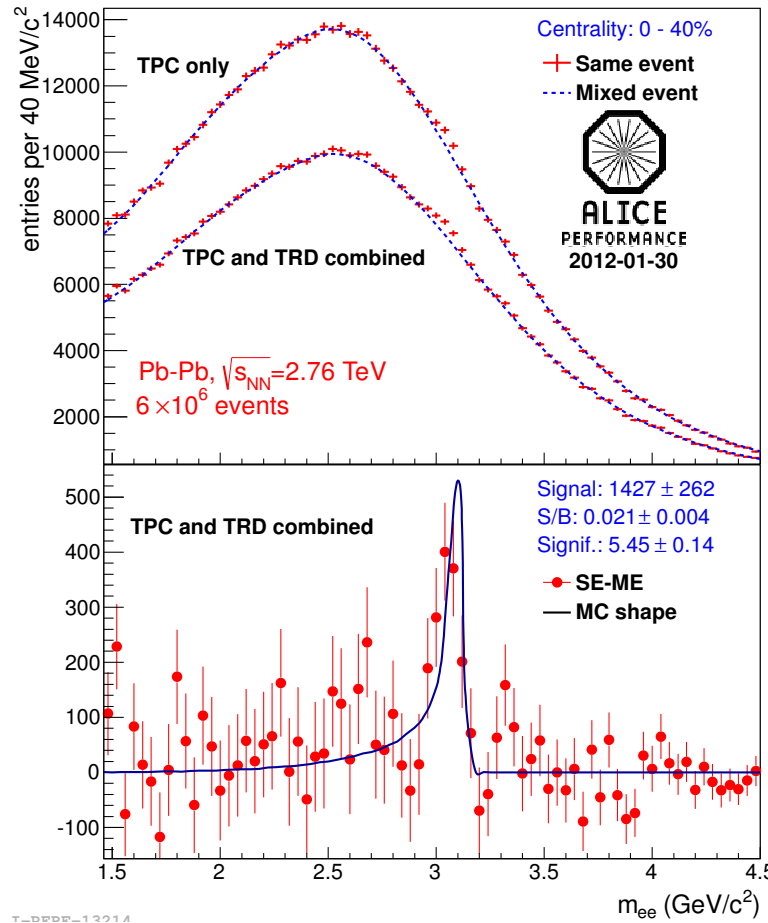


Figure 18: Signal du méson J/ψ dans la distribution de masse invariante des paires d'électrons identifiés dans les collisions Plomb-Plomb pour une classe de centralité entre 0 et 40 %. L'apport du TRD à la réduction du bruit de fond obtenue en réduisant la contamination des hadrons mal identifiés comme électrons γ est démontrée.

L'exigence principale pour l'étude des mésons vecteurs est un pouvoir de discrimination e/π supérieur d'un facteur 100 à celui du système ITS-TPC pour

$p_t \geq 3$ GeV/c. L'exigence d'une bonne résolution en impulsion est déterminée par le besoin de connecter les traces du TRD avec celles de la TPC. Pour cela une résolution en impulsion de 5 % à 100 GeV/c est suffisante. Cela va assurer aussi une bonne sélectivité pour la fonction déclencheur. La minimisation du budget matériel est un problème pour un détecteur à radiation de transition, mais pour le TRD l'épaisseur a été contenue à 15 % de longueur de radiation. La granularité est déterminée par les exigences de résolution des traces doubles (z) et la résolution en impulsion (φ). A $dN_{ch}/dy|_{y=0} = 8000$ l'occupation maximale est de 34 % et l'efficacité de détection est de 80 %.

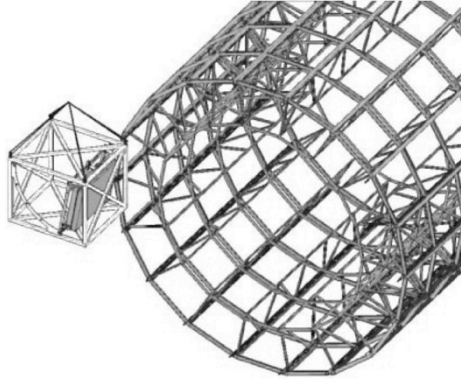


Figure 19: Plan du détecteur de temps de vol avec détail de l'installation dans l'armature spatiale.

Les signaux de 18 damiers sont acheminés à un module multi-puce (en anglais *Multi-Chip Module*, MCM) embarqué qui contient deux ASICs, un pour la pré-amplification, formatage et sortie du signal et l'autre avec l'ADC, une mémoire tampon et le processeur des mini-traces (fraction de trajectoire reconstruite dans le TRD). Tous les MCMs envoient leur information des mini-traces à l'unité globale de trajectographie (en anglais *Global Tracking Unit*, GTU) qui produit un signal pour le processeur central de déclenchement (en anglais *Central Trigger Processor*, CTP).

Le pouvoir discriminant du détecteur pour distinguer les électrons des pions est représenté sur la Figure 17 par le dépôt de charge pour les électrons et les pions de 2 GeV/c d'impulsion. Pour les électrons, le dépôt de charge moyen est plus élevé en raison de la contribution supplémentaire due à l'absorption du rayonnement de freinage (Bremsstrahlung) et du rayonnement de transition dans le gaz des chambres à dérive. Les distributions mesurées à partir des données sont en bon accord avec les mesures effectuées en 2004 sous faisceaux tests de pions et d'électrons. Pour déterminer la pureté des échantillons de particules identifiées, les électrons provenant de la conversion de rayons gamma et les pions provenant de la désintégration de K^0 ont servi de référence. Plusieurs coupures topologiques ont été utilisées pour s'assurer de la qualité des données. L'analyse a été effectuée avec les données proton-proton à 7 TeV provenant de la deuxième reconstruction de la période de prise de données LHC10d du 2010.

Le pouvoir discriminant du TRD pour distinguer les électrons des hadrons améliore le rapport signal sur bruit dans l'identification du méson J/ψ , comme il est montré dans la Figure 18. En comparant la distribution de masse invariante des paires d'électrons identifiés par la TPC (courbe supérieure) et par la TPC et

le TRD (courbe inférieure), nous observons que l'élimination des hadrons faussement identifiés par la TPC comme électrons permet de réduire le bruit de fond de 30 % sans altérer le signal.

Le détecteur de temps de vol (en anglais *Time Of Flight*, TOF, Figure 19) couvre le même intervalle d'acceptance que les autres détecteurs du baril, $|\eta| \leq 0,9$ et 2π et permet d'identifier les particules chargées avec $0,2 \leq p_t \leq 2,5$ GeV/c en combinaison avec les autres détecteurs. L'exigence d'identifier les particules jusqu'à $dN_{ch}/dy|_{y=0} = 8000$ avec 10-15 % d'occupation a motivé l'introduction de 10^5 canaux indépendants. Cela demande une résolution intrinsèque de 40 ps.

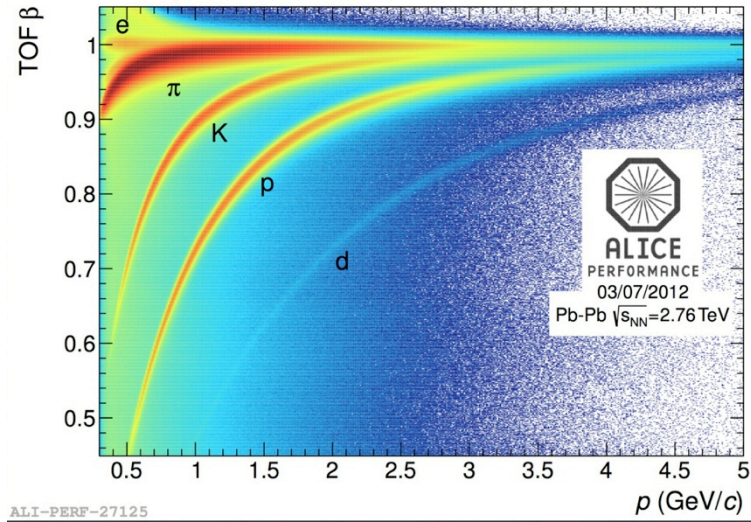


Figure 20 : Distribution des particules détectées en fonction de leur vitesse relative (β) mesurée par le TOF et de leur impulsion. Les données utilisées sont les événements Plomb-Plomb de 2011. Le bruit de fond est dû à des erreurs d'appariage entre le TOF et la TPC.

Le détecteur est composé de 18 secteurs radiaux avec un rayon entre 370 et 399 cm et une longueur de 7,45 m. L'unité de base est une bande de $1.200 (r\phi) \times 74 (z)$ mm² placée dans une chambre à plaque résistive à interstices multiples (en anglais *Multi-gap Resistive Plate Chamber*, MRPC) rempli de gaz à haute pression et soumis à un champ électrique élevé. Le passage d'une particule génère une avalanche dans le gaz qui est détectée par les 96 électrodes collectrices de 25×35 mm² de chaque bande. Ces bandes sont assemblées dans des chambres (modules) courtes (15 bandes) et longues (19 bandes) inclinées et intercalées de façon à ne pas laisser d'espace mort et à être perpendiculaires à une ligne joignant le point d'interaction au centre de chaque bande. Dans chaque secteur il y a un module court central flanqué par deux modules longs, pour un totale de 90 ($18 + 4 \cdot 18$) modules et 157.248 électrodes collectrices. Le budget matériel du détecteur est de 20 % de longueur de radiation.

Les signaux venant de huit détecteurs sont récoltés et mis en forme par un ASIC (NINO) qui transmet les informations du front de montée (en anglais *leading edge*) du signal et du temps total au-dessus du seuil (en anglais *Time Over Threshold*, TOT) qui dépend de la charge totale, au convertisseur de temps à haute performance (en anglais *Time to Digital Converter (TDC) Readout Module*, TRM). Celui-ci à son tour l'achemine vers le module de lecture des données (en

anglais *Data Readout Module*, DRM), qui code le signal et l'envoi via des DLL au système d'acquisition de données.

La distribution des particules en fonction de leur vitesse relative ($v/c = \beta$) et de leur impulsion est représentée dans la Figure 20. Les divers types de particules de masse différente sont clairement identifiables. Le bruit de fond est dû par des erreurs d'appariage entre le TOF et la TPC.

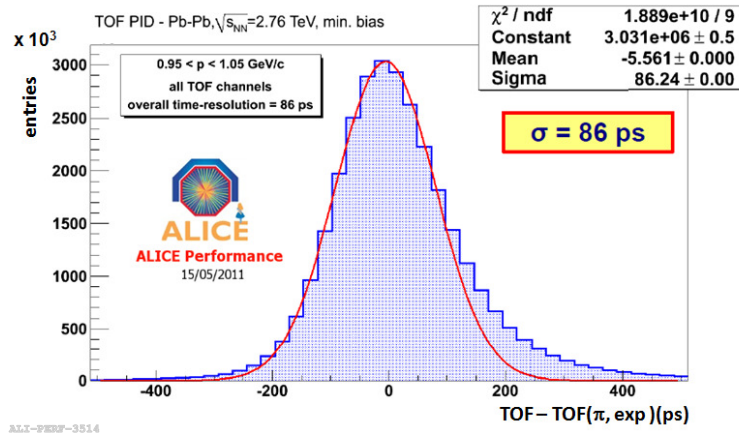


Figure 21 : Résolution de la mesure du temps de vol par le TOF. Des pions sont sélectionnés dans un intervalle étroit d'impulsions (0,95 GeV/c < p < 1,05 GeV/c).

La Figure 21 nous montre la performance du détecteur. Des pions provenant de collisions Plomb-Plomb ont été sélectionnés dans un étroit intervalle d'impulsion (0,95 GeV/c < p < 1,05 GeV/c). Le temps de vol « théorique » est calculé pour chaque particule en intégrant la fonction $1/\beta \, dt$ et il est soustrait du temps de vol mesuré par le TOF. La résolution du détecteur est de 86 ps. La partie de droite de la distribution est non gaussienne pour des raisons qui ne sont pas encore comprises.

Le calorimètre électromagnétique (en anglais *Electro Magnetic CALorimeter* EMCAL, Figure 22) couvre un arc de 107° en φ et a une acceptance $|\eta| \leq 0,7$, ses dimensions étant fortement contraintes par la nécessité de s'adapter entre le TOF et le solénoïde en r et entre PHOS et le HMPID (voir plus loin dans ce document) en φ . L'unité de base est 12.288 tours (en anglais *tower*) de $6 \times 6 \times 25(z) \, \text{cm}^3$ ($\Delta\eta \times \Delta\varphi = 0,014 \times 0,014$) à géométrie approximativement projective vers le point d'interaction. Chaque tour est composée par un empilement de 76 couches de Plomb et scintillateur, où la collection de la lumière est effectuée par 6×6 fibres optiques à décalage de longueur d'onde (en anglais *wavelength shifting fiber*) longitudinales, (configuration dite à Shashlik) regroupées sur la surface de $5 \times 5 \, \text{mm}^2$ d'un photodiode à avalanche (en anglais *Avalanche Photo Diode*, APD). Les tours sont arrangées 2×2 dans des modules, assemblés à leur tour dans dix « supermodules » de $12 \times 24 = 288$ unités et deux de 12×8 . Les supermodules ont une ouverture angulaire de $\Delta\varphi = 20^\circ$ et $\Delta\varphi = 7^\circ$ respectivement et ils suivent la segmentation azimutale de TPC, TRD et TOF. Un système de calibrage à diodes à émission de lumière (en anglais *Light Emission Diodes*, LED) est intégré dans le détecteur.

Le calorimètre électromagnétique a deux fonctions principales. La première est de fournir un signal de déclenchement rapide et non biaisé pour des jets, avec une réjection du bruit d'un facteur 10 pour les collisions Plomb-Plomb et ~ 3.000 pour les collisions proton-proton. Cela va permettre une augmentation du nombre des jets enregistrés jusqu'à ~ 500 pour les collisions proton-proton et 50 dans les cas Plomb-Plomb. La deuxième est de mesurer l'énergie totale des jets avec une résolution intrinsèque en énergie de $(6,9/\sqrt{E} \oplus 1,4)\%$ et en position de $1,5 \oplus 5,3/\sqrt{E_{MIP}}$ afin de pouvoir les connecter aux traces mesurées avec les autres détecteurs du baril. Cela permet une résolution de l'énergie du jet de 30 % dans les cas des jets dans les collisions proton-proton et jusqu'à 20 % pour les jets à haute énergie dans les collisions des ions lourds.

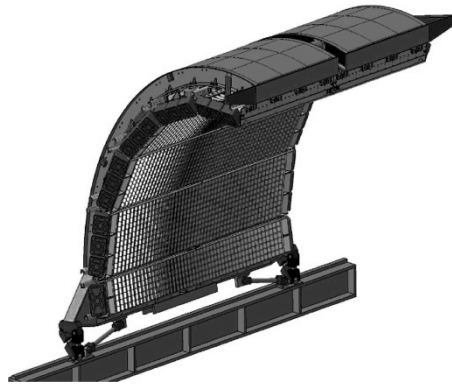


Figure 22 : Plan du calorimètre électromagnétique.

Un amplificateur de charge (en anglais *Charge Sensitive Preamplifier*, CSP) est embarqué avec l'APD. Les signaux de 32 tours sont acheminés après amplification à un circuit électronique frontal (en anglais *Front End Electronics*, FEE) programmable pour les façonner, les échantillonner à 10 MHz et les numériser. La sortie est un signal à 14 bits qui est envoyé à une puce ALTRO que nous avons déjà vue pour la TPC. Selon l'énergie de la particule, le signal final à 10 bits est traité avec amplification basse (250 MeV – 250 GeV) ou haute (16 MeV – 16 GeV). Le circuit FEE fournit aussi un signal « OU logique rapide » pour chaque module, qui est acheminé à un circuit de déclenchement « régional » (en anglais *Trigger Region Unit card*, TRU) qui numérise les signaux et les transmet à une matrice de portes électroniques programmables (en anglais *Field-Programmable Gate Array*, FPGA) pour le calcul du signal de déclenchement de niveau 0 ou 1.

La comparaison entre la résolution de la mesure de l'énergie par le calorimètre obtenue à partir de données de faisceaux test d'électrons mono-énergétiques et celle obtenue à partir des données simulées (Figure 23) indique un accord satisfaisant. La dépendance de la résolution avec l'énergie est décrite par une fonction empirique :

$$\frac{\sigma}{E}(\%) = \sqrt{\frac{a^2}{E^2} + \frac{b^2}{E} + c^2}.$$

La résolution en énergie est un paramètre essentiel pour un calorimètre électromagnétique, en particulier pour la détection des jets. Le terme

« constante » de la résolution relative du calorimètre électromagnétique de ALICE est d'à peine 1,6 %.

La Figure 24 nous montre la distribution de la masse invariante des paires rayon gamma rayon gamma dans le calorimètre électromagnétique de ALICE dans le cas des collisions proton-proton à 7 TeV pour un intervalle d'impulsion transversale entre 16 et 18 GeV/c. Comme nous pouvons le voir, le pic de masse combinée du π^0 est clairement visible et une courbe gaussienne lui a été ajustée. Il est possible aussi de voir très clairement le pic de masse de la particule η autour de 548 MeV.

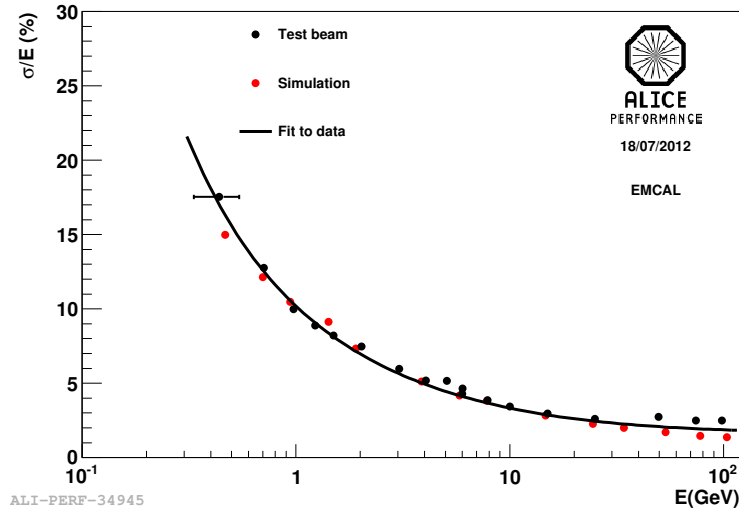


Figure 23 : Comparaison des résolutions en énergie de EMCAL obtenues à partir de données simulées et à partir des données de faisceau test d'électrons mono-énergétiques.. La fonction ajustée aux

données est $\frac{\sigma}{E}(\%) = \sqrt{\frac{a^2}{E^2} + \frac{b^2}{E} + c^2}$.

L'identificateur de particules à haute impulsion (en anglais *High Momentum Particle Identifier Detector HMPID*, Figure 25) a été conçu pour la mesure des hadrons avec $p_t \geq 1$ GeV/c, et il est optimisé pour identifier les particules à haut p_t et avec un grand angle d'ouverture. Son rôle est d'améliorer la discrimination π/K et K/p au-delà de ce qui est possible avec la perte spécifique d'énergie (ITS et TPC) ou le temps de vol (TOF) pour $1 \text{ GeV/c} \leq p_t \leq 3$ ou 5 GeV respectivement. Le détecteur est composé de sept modules carrés de $1,5 \times 1,5$ m dans la position à « deux heures » couvrant 5 % de la surface du baril (voir Figure 25).

Chaque module est formé de 5 couches indépendantes. Quand une particule chargée provenant du centre du baril traverse une couche, elle rencontre une couche de 15 mm de fluoro-hexane (C_6F_{14}) qui agit comme « radiateur » Tcherenkov avec $n = 1,2989$ à $\lambda = 175 \text{ nm}$, ce qui donne un seuil de $p_{th} \geq 1,21 \cdot \text{masse}$.

Les photons produits par effet Tcherenkov traversent une fenêtre de quartz de 5 mm fermant une chambre proportionnelle à fils multiples (en anglais *Multi Wire Proportional Chamber*, MWPC) remplie de méthane (CH_4) à température et pression ambiantes. Dans chaque chambre, le signal est collecté par 6 cathodes de $64 \times 40 \text{ cm}^2$ avec $76 \times 50 = 3.840$ damiers photo-cathodiques de $8 \times 8,4 \text{ mm}^2$

couverts de iodure de césium (CsI). L'anode est réalisée par des fils à 2 mm de distance de la cathode, espacés de 4,2 cm l'un de l'autre et à une tension de 2050 V fournissant un gain de 10^5 .

Dans chaque chambre des puces ASIC GASSIPLEX-07 embarqués réalisent la mise en forme et le filtrage du signal de 16 canaux. Les signaux de trois GASSIPLEX-07 sont ensuite numérisés à 12 bits et envoyés à une puce ASIC DILOGIC pour la soustraction de la ligne de base du signal et l'élimination des canaux avec signal nul. Ensuite, le signal est envoyé au système d'acquisition de données via les DDLs. Le HMPID est capable de fonctionner à une fréquence de 5 kHz et une occupation du 12 % à $dN/d\eta|_{\eta=0} = 8.000$. Ceci est amplement suffisant, vu que le HMPID peut fonctionner seulement pour les traces détectées par la TPC.

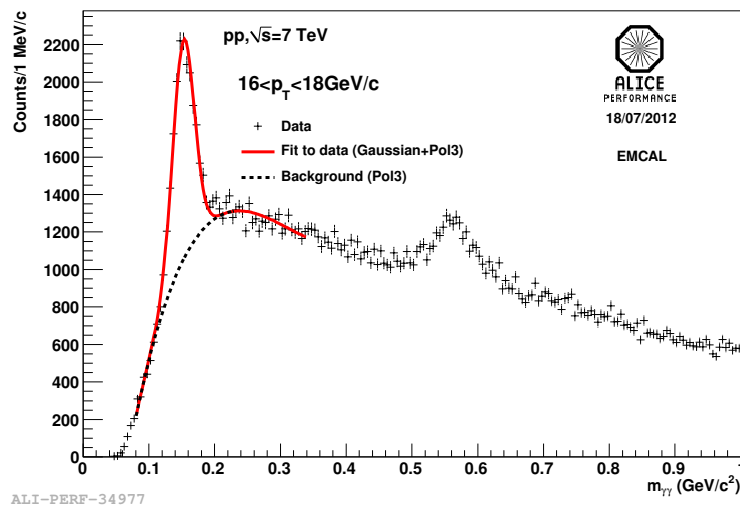


Figure 24 : Distribution de la masse invariante des paires rayon gamma rayon gamma dans EMCAL dans le cas des collisions proton-proton à 7 TeV pour un intervalle d'impulsion transversale entre 16 et 18 GeV/c.

La distribution de l'angle Tcherenkov reconstruit à partir des photons Tcherenkov formant un anneau dans le HMPID (Figure 26) permet de distinguer très clairement les différentes particules dans un domaine cinématique assez grand.

Le pouvoir de séparation du HMPID des différentes espèces de particules en unités d'écart type moyen est représenté dans la Figure 27 en fonction de l'impulsion transversale de particules. Nous pouvons conclure que le HMPID fournit une séparation de deux écarts types moyens entre pions et kaons jusqu'à 4 GeV/c et entre kaons et protons jusqu'à 6 GeV/c.

Le spectromètre de photons (en anglais *PHOTon Spectrometer*, PHOS) est un calorimètre électromagnétique à haute résolution optimisé pour l'étude des photons et des π^0 . Son rôle est d'identifier et d'étudier les photons thermiques. En combinaison avec l'EMCAL il peut contribuer à l'étude des jets et en particulier des corrélations γ -jet et π^0 -jet. Cela permet l'investigation de la phase initiale de l'évolution du plasma à travers la mesure du spectre des photons et di-photons prompts et des corrélations de Bose-Einstein entre photons. L'identification des

photons se base sur l'analyse du profil de la gerbe électromagnétique, sur le temps de vol et sur un détecteur pour particules chargées qui va être installé devant le PHOS (en anglais *Charged Particle Veto*, CPV). Ce cahier des charges pose des demandes assez exigeantes sur les performances du détecteur. D'une part une grande résolution en énergie temps et position est requise, et d'autre part une très grande gamme dynamique, ce qui comporte un bas bruit.

Pour pouvoir satisfaire ces exigences, le PHOS est composé de cinq modules avec une ouverture de 20° placés sur le fond du baril à 460 cm du point d'interaction pour une couverture totale de $|\eta| \leq 0,12$ et $\Delta\phi = 100^\circ$. Chaque module est composé de 59×64 cristaux $22 \times 22 \times 180 \text{ mm}^3$ de tungstate de Plomb (PbWO_4) à haute densité (8 g cm^{-3}) refroidis à -25° pour augmenter la production de lumière par scintillation, pour un total de 17.920 cristaux. Un système de contrôle à LED est intégré dans chaque module. La résolution en énergie est $\sim 1,05 \oplus 1,70/\sqrt{E}$ et en position entre 1 et 4 mm selon l'énergie et l'angle du photon.

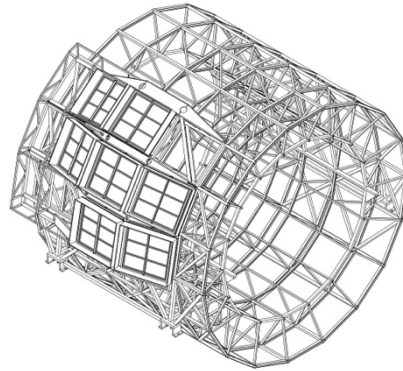


Figure 25 : Plan de l'identificateur de particules à haute impulsion.

Le CPV, qui n'est pas encore installé, est composé de cinq chambres MWPC avec un mélange Ar (80 %)-CO₂ (20 %), chacune couvrant la face côté faisceau d'un module, et dotée de 7168 damiers de $22 \times 10,5 \text{ mm}^2$.

Dans chaque cristal, la lumière est détectée par une photodiode à avalanche (APD) très peu sensible aux particules chargées qui traversent les cristaux. L'électronique pour la lecture et le traitement des signaux des cristaux est la même que celle du EMCAL. Cela permet une gamme dynamique de 100 GeV avec deux niveaux de gain, et une résolution en temps de $\sim 1 \text{ ns}$ à 1 GeV. Des cartes de déclenchement réunissent les signaux de 28×16 cristaux.

Pour le CPV, la puce GASSIPLEX développée pour le HMPID est utilisée. Le signaux de chaque chambre sont acheminés à 32 séries de modules multi-puce (en anglais *Multi-Chip Modules*, MCM) et une mémoire « à colonne » pour lecture et écriture (en anglais *Column-Memory and Read-Write*, CMRW) qui forment les données pour le système d'acquisition.

Le spectre de masse invariante des paires de photons entre 1 et 2 GeV/c produits dans les collisions proton-proton à 2,76 TeV avec (distribution supérieure) et sans (distribution inférieure) bruit de fond autour du pic du π^0 est représenté dans la Figure 28.

Le spectromètre MUON vers l'avant (en anglais *Forward Muon Spectrometer*, MUON) a pour tâche la détection des paires $\mu^+\mu^-$ avec $-4,0 \leq \eta \leq -2,5$ pour la spectrométrie des mésons vectoriels à saveur lourde J/ψ , ψ' , Υ , Υ' et Υ'' , ainsi que du ϕ , et du continuum di-muon avec charge opposée jusqu'à $10 \text{ GeV}/c^2$. L'étude du continuum va permettre la spectroscopie des saveurs lourdes ouvertes. En outre, l'étude des coïncidences e (TRD)- μ (MUON) va permettre l'étude des saveurs lourdes dans l'intervalle $-2,5 \leq \eta \leq -1$. Le « bras » di-muon est un système de détection indépendant du baril et peut enregistrer des événements avec les autres détecteurs du baril ou avec un sous-ensemble composé de ZDC, SPD, PMD, T0, V0 et FMD qui lui permet une fréquence d'acquisition jusqu'à 1 kHz .

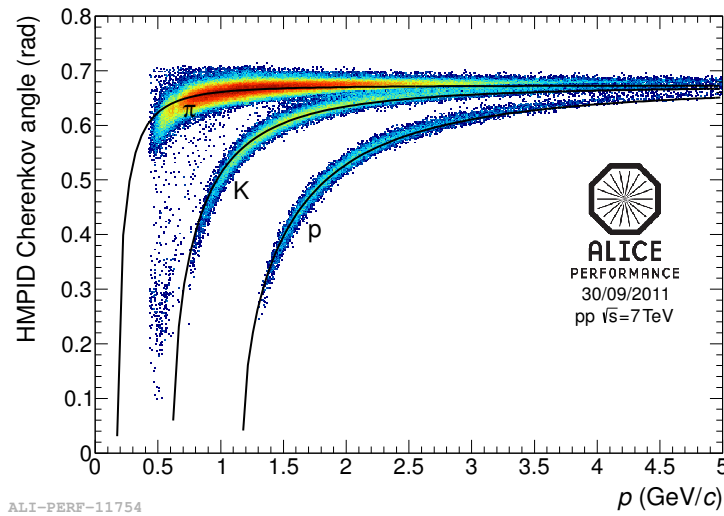


Figure 26 : Distribution de l'angle Tcherenkov en fonction de l'impulsion dans l'HMPID pour les particules primaires chargées émises dans les collisions proton-proton à 7 TeV . Les courbes indiquent les distributions théoriques attendues pour les pions, les kaons et les protons.

Les demandes imposées par la physique incluent la capacité de reconstruire les traces dans un environnement à haute multiplicité et une grande acceptance pour enrichir la statistique, particulièrement à bas p_t où la majorité des mésons vectoriels sont prompts et ne viennent pas de la désintégration des résonances de saveur lourde ouverte. Les muons avec impulsion $\geq 4 \text{ GeV } c^{-1}$ peuvent être facilement « filtrés » par un blindage qui absorbe les autres particules. Cela, avec l'exigence pour les bas p_t , situe naturellement un tel détecteur dans la région à grande rapidité. La demande de pouvoir résoudre les différents éléments des familles des mésons vectoriels à saveurs lourdes se traduit dans une demande de résolution en masse combinée allant de 2% à $3 \text{ GeV}/c^2$ à 1% à $10 \text{ GeV}/c^2$, qui à son tour impose les contraintes sur la précision spatiale des chambres et l'intensité du champ magnétique. Un système de déclenchement permet d'enrichir la statistique récoltée.

Le bras di-muon couvre un angle polaire entre 2° et 9° et il est composé d'un système passif pour le filtrage des muons et du système de détection proprement dit. Le système passif comprend trois éléments : i) un « absorbeur » conique de béton, acier et carbone de 10 longueurs de radiation entre $z = 90$ et 503 cm pour absorber toutes particules autres que les muons ; ii) une « chemise » d'acier, plomb et tungstène autour du tube du faisceau pour toute la longueur du détecteur pour arrêter les

particules produites à $\theta \leq 2^\circ$ qui pourraient « émerger » dans le détecteur ; iii) un mur en béton d'un mètre d'épaisseur entre la dernière chambre trajectographique et les chambres du système de déclenchement à ~ 15 m du centre d'interaction.

Les muons qui traversent l'absorbeur passent dans un aimant où ils sont déviés par un champ magnétique dipolaire horizontal de 0,7 T, avec un intégral de champs de 3 T m.

Le système de détection trajectographique (en anglais *Muon Tracker*, MTR) est composé de cinq paires de chambres à damiers cathodiques remplies d'un mélange de gaz 80 % Ar/20 % CO₂, deux avant le dipôle, une dans le dipôle et deux après, respectivement à 5,36, 6,86, 9,83, 12,92 et 14,22 m du centre d'interaction. Les chambres ont une segmentation avec des damiers de forme et dimension différentes placés dans des quadrants pour les deux premières chambres, et dans des lamelles (en anglais *slats*) pour les trois autres. La dimension des plaquettes va de 4×6 à 5×100 mm² avec un totale de $1,09 \times 10^6$ canaux qui assurent une résolution spatiale verticale (dans le plan de la déviation magnétique) de 70μ et une occupation inférieure à 5 % pour les collisions Plomb-Plomb. Le détecteur est complété par un système de déclenchement (en anglais *Muon Trigger*, MTR) qui consiste en deux paires de chambres à plaques résistives (en anglais *Resistive-Plate Chambers*, RPC) placées après le filtre muon en fer. Ces chambres, réalisées avec des électrodes de bakélite lues par des bandes segmentées, ont une résolution spatiale d'un centimètre. Cela permet de pouvoir appliquer une coupure en p_t d'une ou deux GeV/c, tout en sachant que l'impulsion minimale d'un muon pour arriver jusqu'aux chambres de déclenchement est de 4 GeV/c.

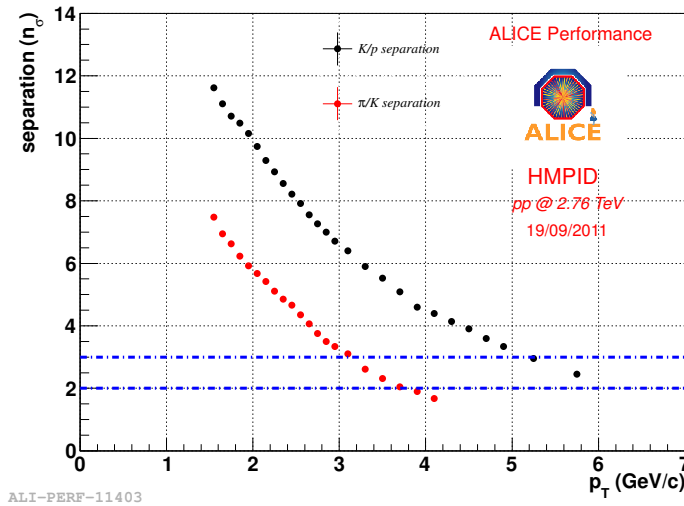


Figure 27 : Pouvoir de séparation de l'HMPID en unités d'écart type moyen en fonction de l'impulsion transversale, obtenu pour des particules produites dans les collisions proton-proton à 2,76 TeV. Les lignes pointillées indiquent les niveaux de séparation à deux et trois écarts types moyens. Les valeurs obtenues sont compatibles avec les spécifications de conception du détecteur.

Les signaux de 16 damiers des chambres trajectographiques sont acheminés vers une puce MANAS pour amplification, filtrage et mise en forme. Les signaux de quatre MANAS sont envoyés aux cartes d'entrée MANU pour la numérisation et

l'élimination des zéros. Jusqu'à 26 MANUs envoient leurs signaux aux systèmes de concentration et de lecture de groupe (en anglais *Concentrator ReadOut Cluster Unit System*, CROCUS), qui, deux par chambre, assurent la liaison avec le système d'acquisition de données. Les chambres de déclenchement sont équipées avec des discriminateurs à double seuil connectés à un circuit de déclenchement local qui détermine l'impulsion du muon en 600 – 700 ns en comparant la position verticale dans les deux chambres.

Dans le spectre de masse invariante des paires de muons détectées par le spectromètre MUON, les pics de masse du J/ψ et du ψ' sont clairement visibles dans la Figure 29. Les données utilisées sont les résultats de la deuxième reconstruction des collisions proton-proton à 2,76 GeV du 2011.

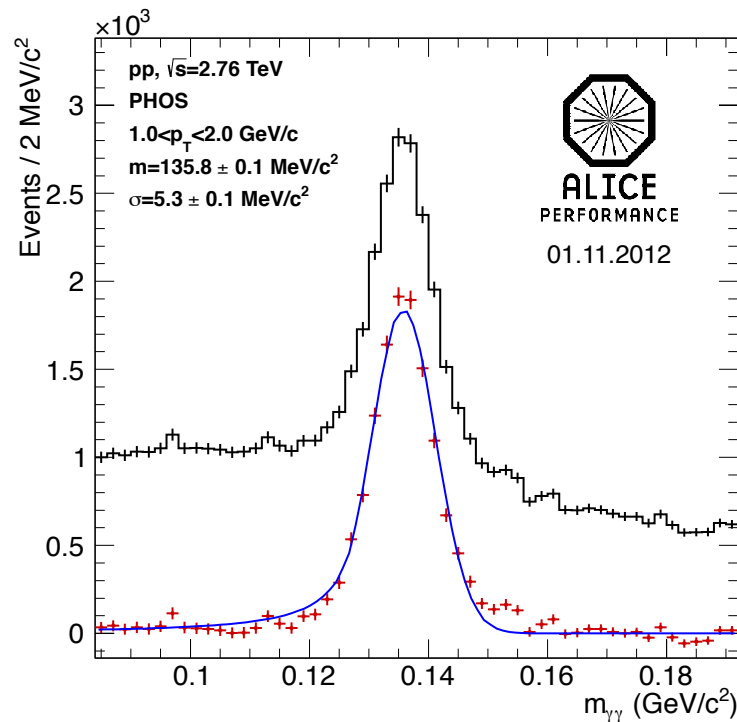


Figure 28 : Spectre de masse invariante mesuré par PHOS des paires de photons entre 1 et 2 GeV/c produits dans les collisions proton-proton à 2,76 TeV avec (distribution supérieure) et sans (distribution inférieure) bruit de fond autour du pic du π^0 . Les données utilisées sont le résultat de la troisième reconstruction de la période de prise de données LHC11a en 2011.

Le calorimètre à zéro degré (en anglais *Zero Degree Calorimeter*, ZDC) a pour fonction de déterminer la centralité de la collision des ions lourds. Dans une collision d'ions lourds, une partie de l'énergie est utilisée dans la collision par les nucléons « participants », et le reste est emporté par les nucléons « spectateurs » qui ne participent pas à la formation de la « boule de feu ». La mesure de l'énergie des spectateurs, qui continuent presque dans la même direction du faisceau, permet une évaluation de la centralité de la collision. Le ZDC est composé de deux ensembles de trois unités de détection positionnés de chaque côté du vertex d'interaction. Les trois éléments de chaque groupe sont : un calorimètre neutronique, un protonique et un électromagnétique. Les

contraintes imposées par le petit espace à disposition et la grande dose de radiation attendue, jusqu'à 10^4 Gy j^{-1} , ont dicté le choix du calorimètre : un absorbeur de grande densité avec des fibres de quartz à effet Tcherenkov pour la détection des particules chargées et le transport de la lumière jusqu'au photomultiplicateurs. Le calorimètre neutronique est placé à ± 116 du centre entre les deux lignes de faisceau et il est composé d'un alliage de tungstène de densité $17,61 \text{ g cm}^{-3}$ avec une dimension de $7,04 \times 7,04 \times 100 \text{ cm}^3$ et 1.936 fibres parallèles au faisceau. Le calorimètre à proton est composé de cuivre avec une dimension de $12 \times 22,4 \times 150 \text{ cm}^3$ et 1.680 fibres parallèles au faisceau. Il est placé à la même hauteur que le calorimètre neutronique mais à côté de la ligne du faisceau, où les particules de charge positive sont déviées après la collision. Le calorimètre électromagnétique est placé à $\pm 7 \text{ m}$ du centre et 8 cm au-dessous du faisceau. Il est composé de Plomb avec une dimension de $7 \times 7 \times 21 \text{ cm}^3$ et des fibres inclinées de 45° par rapport à la direction du faisceau et espacées de 3 mm .

La résolution des calorimètres hadroniques est inférieure à 10% , et pour le calorimètre électromagnétique est inférieure à 1% . Les calorimètres fournissent une contribution au système de déclenchement de niveau 1 (L1) avec définition de trois classes de centralité de la collision.

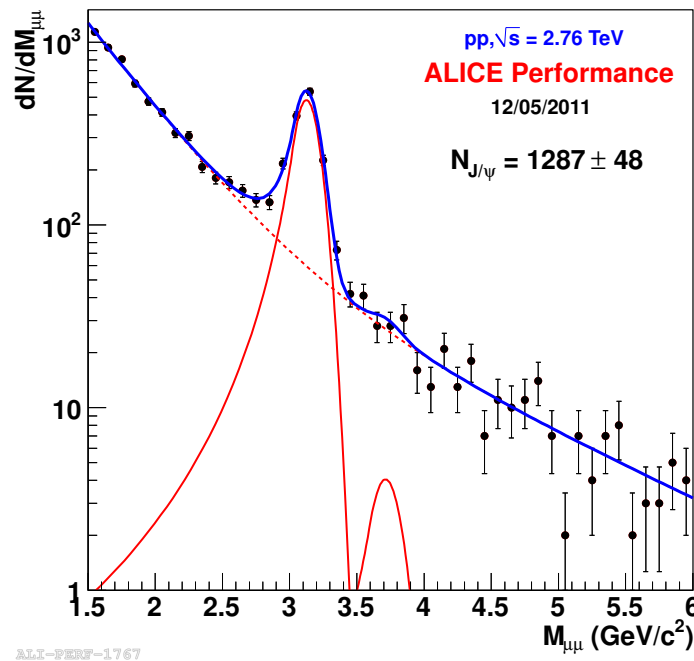


Figure 29 : Spectre de masse invariante des paires de muons détectées par le spectromètre MUON indiquant clairement les pics de masse du J/ψ et du ψ' . Les données utilisées sont celles des collisions proton-proton à $2,76 \text{ GeV}$.

Les événements sont caractérisés en fonction de l'énergie mesurée dans les calorimètres neutroniques et de l'énergie détectée (unité arbitraire) dans le calorimètre électromagnétique (Figure 30). Notons que si le calorimètre neutronique peut être calibré en énergie grâce au signal d'un neutron isolé, la calibration absolue pour le calorimètre électromagnétique n'est pas possible. Les données utilisées sont les collisions Plomb-Plomb de 2011 (LHC11h). Les régions

de centralité sont déterminées par la simulation. Dans les collisions les moins centrales, l'énergie détectée par les deux systèmes calorimétriques est très limitée. En augmentant la centralité, le nombre de neutrons détectés augmente rapidement à cause de la fragmentation des noyaux et de l'évaporation des fragments. Autour d'une centralité de 50 %, le nombre des participants augmente et donc le nombre de neutrons non affectés par la collision diminue. En revanche, le nombre de particules produites augmente rapidement et avec lui le signal dans le calorimètre électromagnétique, surtout à cause des pions neutres. Pour les collisions les plus centrales, tous les nucléons participent à la collision, et donc le signal dans les calorimètres neutroniques est nul et le signal dans le calorimètre électromagnétique est maximum.

Le détecteur de multiplicité photonique (en anglais *Photon Multiplicity Detector*, PMD) a pour rôle la mesure de la multiplicité et de la distribution spatiale des photons dans un intervalle de rapidité $2,3 \leq \eta \leq 3,5$ à 360 cm du centre. Cette position a été choisie parce que l'espace entre le détecteur et le centre d'interaction est relativement libre de câbles et de circuits de refroidissement (les « services ») de TPC et ITS. Le détecteur va participer à l'étude des fluctuations événement par événement, de l'écoulement et de la formation du condensat chiral désorienté (en anglais *Disoriented Chiral Condensate*, DCC) et il peut fournir une information sur l'énergie électromagnétique transversale et sur le plan de réaction.

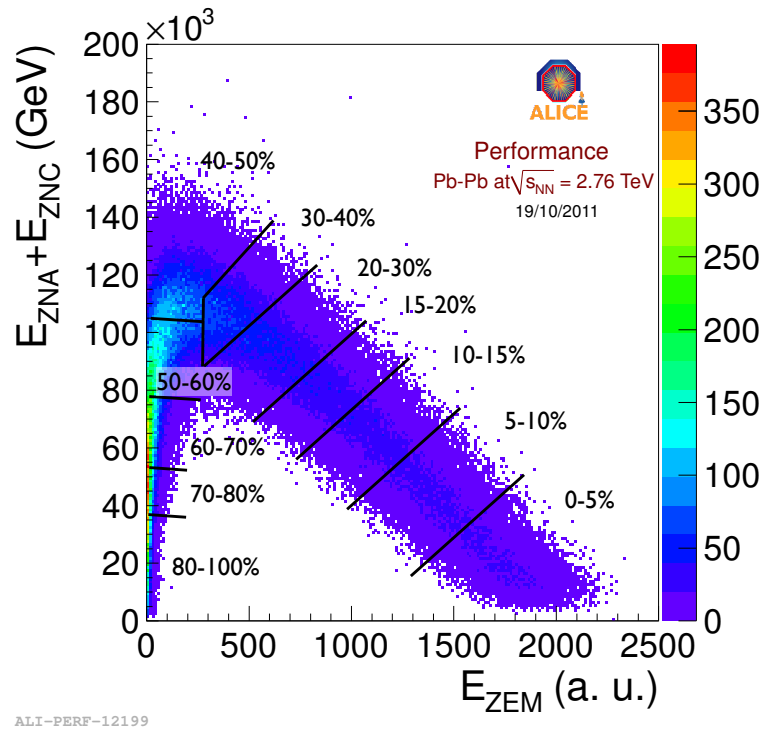


Figure 30 : Distribution de l'énergie détectée par ZDC dans le calorimètre à neutrons en fonction de l'énergie détectée (unité arbitraire) dans le calorimètre électromagnétique. Les régions de centralité surimposées sont déterminées par la simulation.

Le détecteur est composé de deux plans identiques, chacun avec 110.592 cellules hexagonales (structure en nid d'abeille) de 3 mm de côté et 5 mm d'épaisseur,

remplies d'une mixture Ar-CO₂ (70 %-30 %) avec les parois métalliques jouant le rôle de cathode à -1.400 V et un fil central jouant le rôle d'anode. Entre les deux plans il y a 15 mm de Plomb dont le rôle est de convertir les photons et 5 mm d'acier inoxydable pour soutenir les cellules. Le premier plan de cellules face au centre d'interaction agit comme un veto pour les particules chargées (efficacité de détection 96 %), tandis que le deuxième plan détecte les électrons provenant de la conversion des photons dans le plomb et l'acier. Le gaz choisi est insensible aux neutrons, tandis que les parois métalliques des cellules réduisent le danger d'interférence due aux électrons δ . L'occupation prévue pour une collision d'ions lourds à $dN/d\eta_{ch}|_{\eta=0}$ est de 28 % dans le deuxième plan, avec une efficacité pour le photon du 54 %. L'électronique est la même que celle utilisée pour les chambres trajectographique du bras di-muon.

La dépendance de l'énergie moyenne des photons détectés par PMD en fonction de l'énergie du faisceau (Figure 31) indique que les données de ALICE reproduisent celles de UA5 à 900 GeV.

Le détecteur de multiplicité avant (en anglais *Forward Multiplicity Detector*, FMD) mesure la multiplicité des particules chargées dans $-3,4 \leq \eta \leq -1,7$ et $1,7 \leq \eta \leq 5,1$. Le FMD va pouvoir étudier la multiplicité des événements et le flux d'énergie grâce à sa segmentation azimutale, et leurs fluctuations événement par événement. En combinaison avec le SPD il couvre la gamme $-3,4 \leq \eta \leq 5,1$ et, comme il y a recouvrement partiel avec le SPD dans la région centrale, on peut vérifier la consistance des mesures des deux détecteurs.

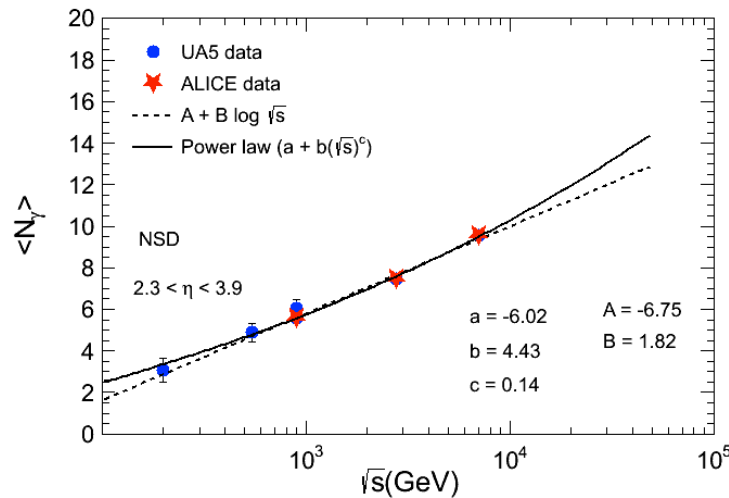


Figure 31 : Dépendance de l'énergie moyenne des photons détectés par le PMD à l'énergie du faisceau dans les collisions proton-proton.

Le détecteur consiste en trois anneaux de rayon 4,2-17,2 cm, divisés en 20 secteurs azimutaux, chacun avec 512 bandes concentriques de silicium, et deux anneaux de rayon 15,4-28,4 cm avec 40 secteurs et 256 bandes respectivement. Les anneaux petits sont placés à -62,8, 83,4 et 340 cm et les grands à $\pm 75,2$ cm.

La segmentation du détecteur permet de limiter l'occupation dans les collisions d'ions lourds à 2,2 particules même dans la bande intérieure de l'anneau petit plus proche du centre. L'information sur la multiplicité est fournie à la fois par le rapport entre bandes occupées et bandes libres et par l'énergie déposée.

Les bandes de silicium sont reliées à deux circuits « hybrides » semi-circulaires qui contiennent les puces VA1 pour l'amplification et la formation du signal et la carte avec les ADCs, qui assurent une fréquence de lecture de 10 MHz. Les signaux numérisés sont acheminés à des puces ALTRO, également utilisées pour la TPC. A la vue de son temps de lecture ($\sim 13 \mu s$), le FMD ne peut pas fournir de signal pour le système de déclenchement.

Un exemple de performance de FMD pour la détermination du plan de réaction est représenté dans la Figure 32 en fonction de la centralité de la collision. Chaque événement est divisé en trois sous-événements : FMD, TPC $0,2 < \eta < 0,8$ et TPC $-0,9 < \eta < -0,2$ et un coefficient de corrélation angulaire pour chaque harmonique est calculé. Ce coefficient est utilisé pour la correction de la non-uniformité de l'acceptance azimutale du FMD (voir [30]).

Le détecteur V0 fourni un signal de déclenchement de biais minimum pour les détecteurs du baril dans les collisions proton-proton en rejetant les interactions faisceau-gaz. Il fournit aussi deux signaux de déclenchement pour la centralité des collisions Plomb-Plomb. En tant que détecteur, il fournit une évaluation de la centralité, il participe au contrôle de la luminosité et il contribue à filtrer le bruit dans le bras di-muon dans les collisions proton-proton.

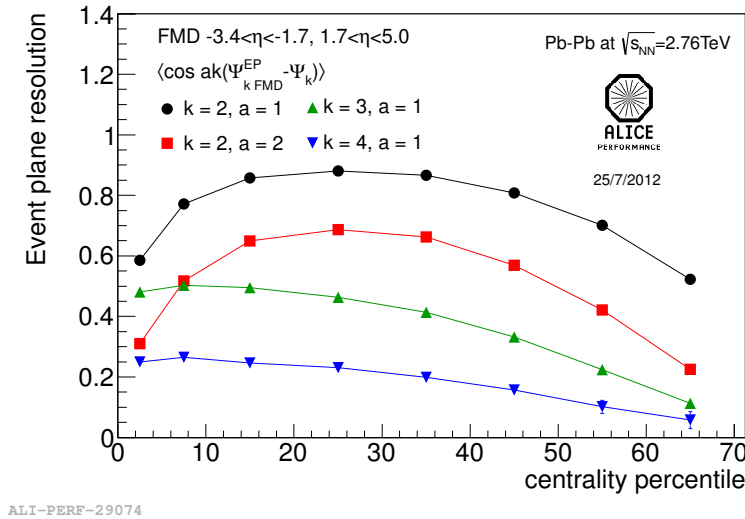


Figure 32 : Moyenne du cosinus de l'harmonique d'ordre k de la distribution du plan de réaction calculé à partir des données de FMD et deux sous-événements sélectionnés à partir des données de la TPC en fonction de la centralité.

Le détecteur est composé de deux disques de rayon interne 4,6 cm et externe 38 cm placés de chaque côté du point d'interaction respectivement à -90 cm contre l'absorbeur du spectromètre MUON et à 340 cm du côté opposé, couvrant une région de rapidité de $-3,7 \leq \eta \leq -1,7$ et $2,8 \leq \eta \leq 5,1$ respectivement. Chaque disque est composé de quatre anneaux divisés en 8 secteurs azimutaux de 45° , sauf les deux anneaux externes du disque à -90 cm qui sont divisés en 16 secteurs.

Les exigences pour le fonctionnement de ce détecteur sont une gamme dynamique de 1 à 1000 fois le signal d'une particule au minimum d'ionisation

(en anglais *Minimum Ionizing Particle*, MIP) pour accommoder à la fois les signaux des collisions proton-proton et des ions lourds. En outre une résolution temporelle de 1 ns est nécessaire pour discriminer les collisions faisceau-gaz.

Pour répondre à ces exigences, le détecteur est composé de scintillateurs de 2 cm d'épaisseur avec des fibres WLS intégrées avec un pas de 10 mm. A la sortie du détecteur, les fibres WLS rejoignent des fibres normales qui acheminent la lumière vers les photomultiplicateurs placés dans le volume magnétique du solénoïde.

Les signaux sortant des photomultiplicateurs sont acheminés à une carte d'interface de canal (en anglais *Channel Interface Unit*, CIU) qui contient la même chaîne électronique que le TOF (puce NINO et TDC). En plus, il y a une unité d'interface et de concentration des canaux (en anglais *Channel Concentration and Interface Unit*, CCIU) qui se charge de recevoir les signaux du CIU et de les envoyer, après codage, au système d'acquisition des données.

La distribution de la densité des particules chargées par unité de rapidité est représentée dans la Figure 33 en fonction de la rapidité des particules et pour différentes classes de centralité des collisions Plomb-Plomb à 2,76 TeV. Les mesures du FMD et V0 ont été obtenues avec la technique des vertex déplacés, c'est-à-dire en utilisant des collisions entre un paquet du faisceau principal et des paquets « satellites » à $\pm 37,5$ cm du point d'interaction principal. L'accord entre les différents détecteurs est assez satisfaisant.

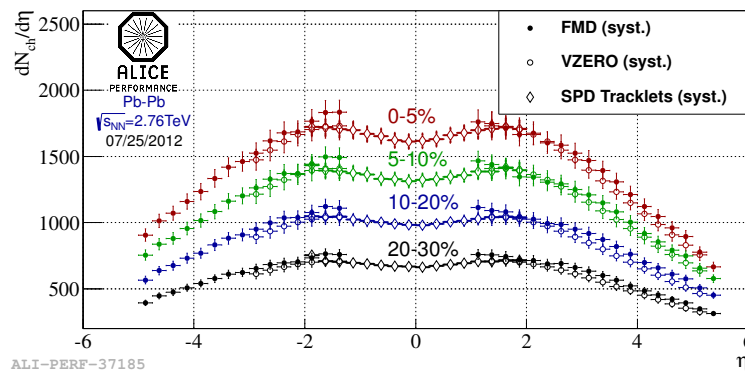


Figure 33 : Distribution de la densité des particules chargées par unité de rapidité mesurée avec le SPD, le FMD et le V0 pour les collisions Plomb-Plomb à 2,76 TeV. Plusieurs classes de centralité sont représentées.

Le détecteur du temps initial T_0 mesure le temps de l'interaction indépendamment de la position exacte du point d'interaction. Cela donne le moment initial pour le TOF et fournit un signal de réveil (en anglais *wakeup*) au TRD avant le signal de déclenchement de niveau 0 (en anglais *Level 0 trigger*, L0). De plus, le T_0 mesure la position du centre de l'interaction et permet d'estimer la multiplicité de l'événement et de fournir trois niveaux de centralité. La demande de précision en temps est de 50 ps, qui se traduit en une précision d'estimation du point d'interaction de 1,5 cm.

Le détecteur est composé de deux rangées de 12 compteurs Tcherenkov. La première rangée est placée à -70 cm du vertex, dû à la présence de l'absorbeur

du bras di-muon, et couvre l'intervalle de rapidité $-3,3 \leq \eta \leq -2,9$. Le deuxième détecteur est placé à +350 cm et couvre l'intervalle de rapidité $4,5 \leq \eta \leq 5$.

Chacun des 24 compteurs est composé d'un radiateur Tcherenkov de quartz de 3 cm de diamètre et 3 de longueur, attaché à un photomultiplicateur du même diamètre. L'efficacité de déclenchement est de 48 % pour les collisions inélastiques proton-proton, mais pratiquement de 100 % pour les collisions d'ions lourds.

Les signaux sont envoyés à la fois au TRD, où ils sont traités par l'électronique pour obtenir le signal de réveil et à la chaîne électronique de traitement du T0, qui est très semblable à celle du TOF.

La performance du T0 en termes de résolution et stabilité est représentée dans la Figure 34. La partie inférieure de la figure montre la différence entre le temps de l'interaction détecté par le détecteur de droite et celui de gauche corrigé par la position du vertex principal de la collision. La résolution de cette mesure de temps est inférieure à 23 ps. La distribution supérieure montre l'évolution de la résolution sur plusieurs périodes de prise de données. Aucune dérive de la résolution n'est observée.

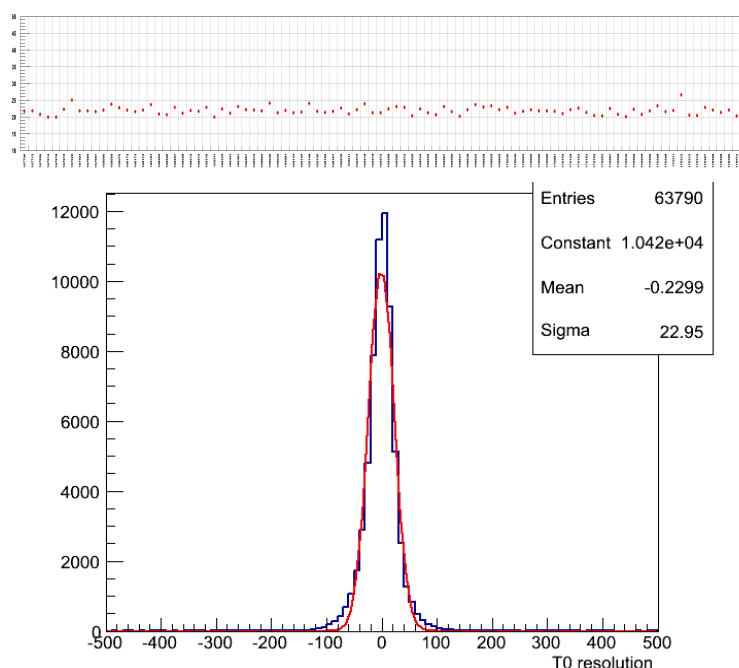


Figure 34 : Différence entre les temps de détection du signal de l'interaction mesurés par chacun des deux détecteurs T0, et corrigés par la position du vertex d'interaction. La distribution supérieure montre l'évolution de la résolution sur un certain nombre de périodes de prise de données.

Le détecteur de rayons cosmiques (en anglais *A COsmic Ray DEtector*, ACORDE) fournit un signal de déclenchement de niveau 0 pour la détection de rayons cosmiques d'une énergie initiale entre 10^{15} et 10^{17} eV. La détection des rayons cosmiques est importante pour l'alignement et le calibrage des détecteurs, et ouvre aussi la possibilité d'une étude du spectre des muons cosmiques, comme

expliqué plus haut. Le seuil pour les muons cosmiques dans ALICE est de 17 GeV, et la limite de reconstruction de l'impulsion est autour de 2 TeV avec un champ magnétique de 0,5 T. Le système de déclenchement de ALICE permet d'acquérir des événements cosmiques entremêlés aux collisions pendant le fonctionnement de l'accélérateur. Une fréquence d'environ 115 Hz a été mesurée pour les muons et d'environ 2 Hz pour les événements multiples.

Le détecteur est composé de 60 doubles couches de scintillateur $1,88 \times 0,2 \text{ m}^2$ disposées en six rangées, deux sur le vertex du dipôle et deux de chaque côté sur les faces supérieures de l'octogone.

Les signaux des scintillateurs sont envoyés à quatre modules de coïncidence, chacun regroupant 30 scintillateurs. Si un module observe un signal en coïncidence, un signal de niveau 0 est généré pour le système de déclenchement dans une fenêtre en temps de 450 ns.

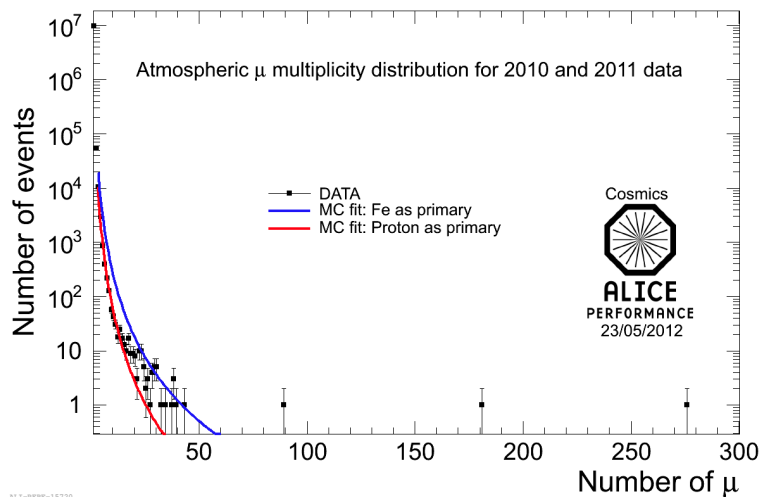


Figure 35 : Distribution de la multiplicité des muons détectés dans ACORDE comparée aux distributions simulées pour des protons et des noyaux de Fer comme rayons primaires.

La distribution de la multiplicité des muons détectés dans ACORDE mesurée en 2010 et 2011 est comparée aux courbes simulées pour des protons et des noyaux de Fer comme rayons primaires (Figure 35). Il ressort clairement de cette comparaison que les multiplicités inférieures sont plus proches de la courbe simulée pour les protons, tandis que pour les multiplicités plus élevées les données tendent vers la courbe simulée pour le Fer. Ce comportement est attendu pour une composition du rayonnement cosmique où la masse des rayons augmente avec leur énergie, résultat confirmé par plusieurs expériences précédentes. Cependant, on peut observer qu'en 2010 et 2011 nous avons enregistré des événements avec un nombre étonnamment élevé de muons. En particulier, la plus haute multiplicité reconstruite par le TPC correspond à une densité de muons de $18 \mu/\text{m}^2$.

L'énergie estimée du rayon cosmique primaire pour de tels événements est au moins de $3 \times 10^{16} \text{ eV}$ (10.000 fois l'énergie du LHC), en supposant que le cœur de la gerbe soit à l'intérieur de ALICE et que la particule primaire soit un noyau de fer. Selon les connaissances actuelles sur les rayons cosmiques, un événement

avec une telle multiplicité est attendu une fois tous les 4-5 ans. Comme de tels événements ont été trouvés dans les 10 premiers jours de prise de données, une étude plus approfondie sera nécessaire pour comprendre s'ils sont dus aux rayons cosmiques standards, et la multiplicité élevée serait tout simplement une fluctuation statistique, ou s'ils résultent d'un mécanisme de production différent. Une étude détaillée de ces événements n'a pas montré de comportement inhabituel dans les autres variables mesurées.

Il est intéressant de noter que trois expériences LEP (L3, ALEPH et DELPHI) ont également constaté un excès d'événements de haute multiplicité qui n'ont pas été expliqués par les modèles de MonteCarlo. L'espoir de ALICE est de trouver et d'étudier un grand nombre de ces événements d'une manière plus quantitative pour bien comprendre leur nature.

2.4 Le traitement des données

Comme décrit précédemment, les données récoltées par les détecteurs sont acheminées vers deux systèmes, le système de déclenchement (en anglais *Trigger*), qui décide si l'événement doit être enregistré ou non, et le système d'acquisition des données (en anglais *Data Acquisition system*, DAQ), qui se charge du codage des données provenant des différents sous-systèmes, de leur assemblage et de leur enregistrement sur support permanent. L'ensemble de ces deux systèmes est appelé « traitement en-ligne » (en anglais *Online Computing*, ou simplement *Online*). Une fois enregistrées, les données sont traitées de façon asynchrone pour « reconstruire » les trajectoires et l'identité des particules (reconstruction) et extraire les informations physiques des événements (analyse) par un système de traitement « hors-ligne » (en anglais *Offline Computing*, ou simplement *Offline*).

Pour une expérience comme ALICE, cette distinction classique est assez schématique et trop simpliste. En effet il existe un système « en-ligne » appelé déclencheur de haut niveau (en anglais *High Level Trigger*, HLT) censé reconstruire et identifier les traces de façon synchrone avec l'acquisition des données. Cela permet de générer un signal de déclenchement basé sur les caractéristiques « physiques » de l'événement et d'écrire sur support permanent les événements dans un format beaucoup plus économe en espace de stockage que celui utilisé par le système d'acquisition de données, au prix d'une perte relative d'information. Les algorithmes utilisés dans le HLT sont soit les mêmes, soit une version simplifiée et plus rapide des algorithmes « hors-ligne ». Le HLT peut être aussi utilisé pour « comprimer » les données, en réalisant en-ligne une partie du traitement des données.

Les signaux mesurés par les détecteurs sont « dé-calibrés » dans le sens que les différents éléments de détection ne génèrent pas le même signal pour une même particule, et donc ils doivent être « calibrés ». Cela est réalisé hors-ligne. Cependant pour permettre de traiter les données assez rapidement après leur acquisition, une version simplifiée des algorithmes de calibrage est en fonction « en-ligne ». Ce système peut produire, après chaque période continue de prise de données (en anglais *run*), des informations de calibrage suffisant pour une reconstruction assez rapide des événements dans le but de produire des résultats de physique assez rapidement.

2.5 Le traitement des données en-ligne

Comme on peut le voir dans la Table 1, le taux d'interactions est de 8 kHz pour les interactions Plomb-Plomb et de 350 kHz pour les interactions proton-proton à la luminosité nominale. Il n'est pas possible d'enregistrer toutes les interactions pour deux raisons. En premier lieu la fréquence maximale avec laquelle la plupart des détecteurs peuvent enregistrer les données est généralement inférieure à 1 kHz. La deuxième raison est que la taille d'un événement est de $\sim 0,5$ Mo pour les interactions proton-proton et de ~ 5 Mo pour les interactions Plomb-Plomb. Ainsi, pour enregistrer toutes les collisions, il faudrait pouvoir faire face à un flux de données de 170 Go/s dans un cas et de 40 Go/s dans l'autre. Non seulement cela engendrerait des coûts prohibitifs en termes d'infrastructure et de supports de stockage, mais cela serait également très inefficace parce que la pluparts des événements enregistrés seraient sans intérêt physique, et donc à écarter de toute analyse.

Déclencheur	Taux moyen
Centraux	70 Hz
Semi-centraux	70 Hz
Biais minimal	10 Hz
EMCAL	30 Hz
UPC ¹²	20 Hz
MUON	420 Hz
Test	10 Hz
PHOS	10 Hz
Temps mort	33 %

Table 3 : Exemple d'un mélange de classes de déclenchement. Ce mélange a été utilisé pendant la prise de données des collisions Plomb-Plomb en 2011. Dans la colonne de gauche nous avons le type d'événement enregistré et dans celle de droite la fréquence attendue.

Il est donc nécessaire de déclencher l'enregistrement des données sous certaines conditions physiques. C'est le rôle du système de déclenchement (en anglais *Trigger System*). Le principe de base en est très simple : certains détecteurs fournissent un signal qui permet à un circuit électronique de décider si l'événement doit être sauvegardé ou non, et ce circuit renvoie des signaux aux détecteurs qui déclenchent l'enregistrement des données. La réalisation pratique de ce principe est compliquée par le fait que les différents détecteurs ont des temps de traitement des signaux différents, et pour chaque détecteur les différentes phases de construction des signaux ont une durée très variable. Donc le principe de base est implémenté à travers différents « niveaux » (en anglais *Level*) chacun avec une échelle de temps différente.

Le traitement des signaux dans chaque détecteur est subdivisé en plusieurs phases, par ordre croissant de durée et les données sont passées à la phase suivante seulement si un signal du niveau précédent est reçu. Les différents

¹² Quand deux noyaux se croisent à une distance telle que les interactions qui ont lieu seulement avec les champs de photons virtuels de deux noyaux, nous parlons de collisions ultrapériphériques (UPC en anglais *Ultra Peripheral Collisions*).

niveaux correspondent aussi à des décisions de plus en plus lentes, soit parce que la logique est plus complexe, soit parce que des détecteurs plus « lents » sont impliqués. Cela permet de « masquer » partiellement le temps de traitement et de décision en perdant le minimum d'événements intéressants. Dans le cas idéal, le temps nécessaire pour prendre une décision de premier niveau correspond à l'inverse de la fréquence d'arrivée des données. Ce premier niveau de décision réduit la fréquence des événements présentés à l'entrée du deuxième niveau, qui pourra dans ces conditions prendre des décisions plus complexes, et donc nécessitant un temps de traitement plus long, sans perdre de données, et ainsi de suite. Suivant cette logique, n'importe quelle complexité décisionnelle peut être implémentée en construisant un système à plusieurs niveaux. En pratique, le système final est le résultat d'un compromis entre les demandes de la physique et le coût et la complexité du système.

Le système de déclenchement de ALICE peut sélectionner plusieurs types d'événements avec des taux d'échantillonnage adaptés aux limitations imposées par les systèmes DAQ et HLT. Le défi pour le système de déclenchement de ALICE est de satisfaire les exigences d'un grand nombre de détecteurs avec des temps de traitement des données très différents et deux modes de fonctionnement distincts : proton-proton et Plomb-Plomb.

Le système de déclenchement de ALICE est composé d'un niveau 0 (L0) qui envoie un signal aux détecteurs $1,2 \mu\text{s}$ après la collision. Ce signal étant trop rapide pour prendre en compte tous les détecteurs, un deuxième signal dit de niveau 1 (L1) prend en compte les détecteurs plus « lents » et est envoyé $6,5 \mu\text{s}$ après la collision.

Le temps de « lecture » du signal est particulièrement long pour deux détecteurs, la TPC ($\pm 88 \mu\text{s}$ autour du signal de déclenchement) et le SDD ($\pm 10 \mu\text{s}$). Pendant la durée de la lecture, d'autres collisions peuvent avoir lieu et rendre les données difficiles à interpréter du fait de l'impossibilité de décider à quel événement appartiennent les traces reconstruites, ce qui est connu sous le nom de phénomène d'empilement (en anglais *pile-up*). Pour cela un niveau additionnel à ± 88 et $\pm 10 \mu\text{s}$ a été introduit (L2), appelé protection passé-futur. Ce niveau de déclenchement permet d'enregistrer une collision seulement si d'éventuelles collisions ayant lieu pendant cette durée respectent certaines conditions.

Les signaux de déclenchement provenant des différents détecteurs sont traités par le processeur central de déclenchement (en anglais *Central Trigger Processor*, CTP) qui les analyse et renvoie les signaux de déclenchement des différents niveaux aux détecteurs. Le CTP peut recevoir jusqu'à 24 entrées de niveau 0, 20, de niveau 1 et 6, de niveau 2. L'unité de base de la logique du CTP est la « classe ». Une classe de déclenchement est composée d'un ensemble d'entrées provenant des détecteurs aux niveaux 0 et 1, de leur combinaison logique requise à chaque niveau, des détecteurs qui doivent être lus par le système d'acquisition des données lorsque les entrées ont la configuration requise (grappe de détecteurs, en anglais *cluster*) et du facteur d'échantillonnage. Il est possible de définir jusqu'à 50 classes et six grappes, chaque grappe définissant une protection passé-futur, c'est-à-dire une logique de niveau 2, spécifique. Le même événement peut déclencher plusieurs classes.

Le CTP doit fonctionner avec un budget de temps très serré, mais il doit également produire des données qui gardent une trace des conditions ayant amené la décision quelle qu'elle soit. Ainsi, pour chaque signal de déclenchement accepté, l'identificateur de l'événement ainsi que la description des classes de déclenchement activées sont envoyés à tous les détecteurs. De la même façon, toutes les interactions dans la même orbite du faisceau sont classées en trois niveaux de centralité et enregistrées pour aider à l'identification des événements empilés (liste d'interaction, en anglais *interaction record*). De façon générale, toutes les entrées et sorties sont comptées par des compteurs (en anglais *scalars*) dont les valeurs sont utilisées pour la détermination des sections efficaces.

Dans la Table 3 nous avons une configuration typique pour le système de déclenchement qui a été utilisé pour la prise de données Plomb-Plomb en novembre 2011. La colonne de gauche nous montre les classes d'événements qui vont être sélectionnés par des combinaisons logiques de signaux de déclenchement provenant de détecteurs. La colonne de droite nous donne la fréquence attendue pour ces signaux, ce qui est une combinaison de l'occurrence « naturelle » des signaux rares, pour lesquels on veut intégrer toute la luminosité fournie par la machine, et d'une réduction (en anglais *downscaling*) qui est opérée par le CTP pour les événements plus fréquents (du type « biais minimal ») pour ne pas épuiser la fenêtre de temps ouverte pour l'acquisition d'un événement.

Le « temps mort » c'est la pourcentage du temps pendant lequel le détecteur est occupé (en anglais *busy*) pour enregistrer un événement et donc il ne peut pas en accepter un autre.

Faisceaux	Taille des événements en Mo		Facteur de compression
	Sans compression	Avec compression	
proton-proton	1,1	0,72	35 %
proton-Plomb	1,0	0,75	25 %
Plomb-Plomb	7,5	6,50	13 %

Table 4 : Dimensions moyennes des événements mesurées pendant les deux premières années de fonctionnement. La taille effective de chaque événement change de façon substantielle selon les modalités de la collision.

Le système *d'acquisition des données* (DAQ) a pour but la collecte des données provenant des différents détecteurs, leur assemblage en événement, c'est-à-dire en structure de données qui contient toutes les informations enregistrées pour une collision, et finalement leur enregistrement sur un support de stockage permanent, c'est-à-dire des bandes magnétiques. Le format des données utilisé par le DAQ est appelé donnée brute (en anglais *raw data*, RAW) pour indiquer que c'est le point de départ de la chaîne de traitement hors-ligne. Des raisons technologiques et budgétaires limitent pour ALICE la bande passante du DAQ vers le système de stockage à 4 Go s⁻¹. Cela doit être considéré comme l'enveloppe avec laquelle il faut être capable de collecter les différents types de données nécessaires au programme de physique de l'expérience. La durée totale pour la prise des données en mode proton-proton est de huit mois et d'un mois

en mode Plomb-Plomb (10^7 s et 10^6 s en tenant compte du coefficient de disponibilité de la machine).

La Table 4 nous montre les dimensions moyennes des événements enregistrés pendant les deux premières années d'exploitation du détecteur. Les valeurs reportées sont des moyennes et la dimension des événements est très variable en fonction des conditions de la collision, qui peuvent générer plus ou moins de particules secondaires, de l'énergie des collision et de la luminosité de la machine qui peut provoquer plus ou moins de collisions dans le même événement. La compression reportée dans la table consiste dans le remplacement effectué en ligne des données brutes avec les clusters calculés par le déclencheur de haut niveau pour la TPC.

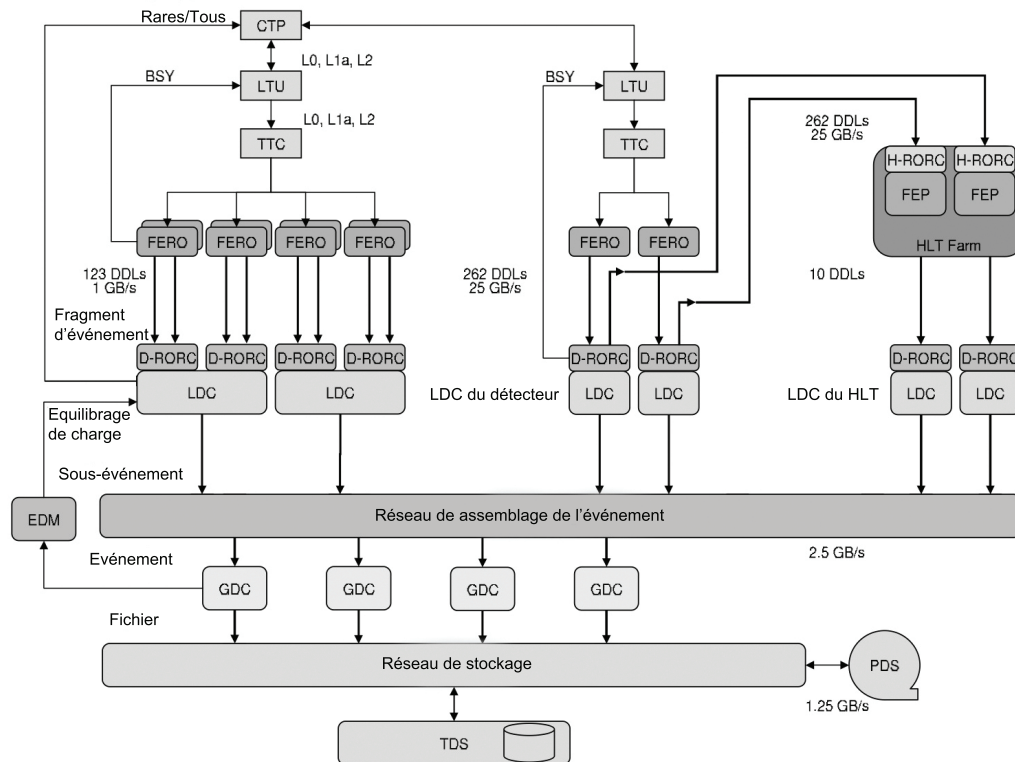


Figure 36 : Plan du système d'acquisition de données de l'expérience ALICE.

Dans le cas proton-proton, la taille d'un événement contenant une seule collision est de 200 ko après compression. Cette valeur doit être modifiée en tenant compte de l'empilement des événements qui pour les collisions proton-proton peut aller jusqu'à un facteur 20. La valeur d'empilement tolérable dépend aussi de la possibilité de filtrer les événements du bruit grâce à l'HLT, mais cela demandera beaucoup d'expérience avec le détecteur. C'est la même chose aussi pour la fréquence des collisions, que peut avoir une valeur instantanée approchant les 1000 Hz au début du cycle pour décroître après. Ici aussi il faut tenir compte des collisions du spectromètre MUON. Tout ça doit être évidemment adapté aux conditions du faisceau et aux choix de la coordination de physique de l'expérience. Les valeurs moyennes typiques que nous retiendrons

sont un facteur d'entassement de cinq, avec une fréquence de 100 Hz, ce qui donne 100 Mo s^{-1} et 10^9 événements proton-proton par année.

La Figure 36 donne une vue d'ensemble du système d'acquisition des données. Les données provenant de détecteurs sont envoyées depuis les unités de lecture d'entrée (en anglais *Front End Readout Unit*, FERRO) aux cartes électroniques de réception et lecture du DAQ (en anglais *DAQ ReadOut Receiver Card*, D-RORC) par le biais de connections à fibre optique (en anglais *Detector Data Link*, DDL) capables de transférer jusqu'à 200 Mo s^{-1} . Les D-RORC, à base de composants FPGA, sont installées dans le bus PCI¹³ des ordinateurs qui assemblent les données d'un certain nombre de détecteurs en sous-événements (en anglais *Local Data Concentrators*, LDC). Chaque D-RORC a deux interfaces DDL. Les données de certains détecteurs sont traitées par le système HLT. Pour ces détecteurs, une seule des interfaces reçoit les données du détecteur, la deuxième les copie directement dans la ferme HLT pour pouvoir les traiter en temps réel.

Les LDC sont des PCs (en anglais *Personal Computers*) standard qui assemblent les données reçues en sous-événements. La communication entre le D-RORC et le processeur central (en anglais *Central Processing Unit*, CPU) du LDC est réalisée par accès direct à la mémoire (en anglais *Direct Memory Access*, DMA). Les données collectées par les LDCs sont ensuite envoyées aux concentrateurs globaux de données (en anglais *Global Data Concentrator*, GDC).

Les GDCs sont aussi des PCs standard qui ont pour rôle d'assembler les sous-événements en un événement qui est ensuite envoyé vers un système de disques tampon. De là, les fichiers contenant les événements sont envoyés vers un autre serveur de disque situé au centre de calcul du CERN pour être copiés sur bande magnétique.

Le processeur central de déclenchement (en anglais *Central Trigger Processor*, CTP) envoie les décisions des différents niveaux ainsi que les données du déclencheur à l'unité de déclenchement locale (en anglais *Local Trigger Unit*, LTU) de chaque détecteur, qui est elle-même interfacée à un système de temporisation, déclenchement et contrôle (en anglais *Timing, Trigger and Control*, TTC). L'électronique FERRO de chaque détecteur ou le D-RORC peuvent envoyer un signal « occupé » (en anglais *busy*) au CTP via la LTU grâce à un circuit de retenue à « contre pression » (en anglais *back pressure*) qui peut arrêter l'arrivée des nouvelles données au LDC si le système n'est pas prêt à les recevoir.

Ce système évite de perdre les événements rares. Quand le DAQ n'est pas prêt à recevoir d'événement, le système de déclenchement privilégie les événements rares aux dépens d'événements plus fréquents ou de basse priorité. Ainsi, aucun événement rare n'est perdu lors d'une surcharge du DAQ.

La topologie des DDL est fixe, chaque LDC reçoit les DDL provenant d'un seul et même détecteur. Les fragments d'événements de ce détecteur sont assemblés en sous-événements dans le LDC. Par contre, la topologie entre LDC et GDC est dynamique. Les deux systèmes sont reliés par le même réseau d'assemblage des

¹³ Un bus est un système de communication entre les composants d'un ordinateur. Le *Peripheral Component Interconnect* (PCI) est un standard de bus local (interne) permettant de connecter des cartes d'extension sur la carte mère d'un ordinateur.

événements (en anglais *Event Building Network*). Quand tous les sous-événements d'un même événement sont prêts dans les LDCs des détecteurs de la grappe active, l'administrateur de la destination des événements (en anglais *Event Destination Manager*, EDM) choisi un GDC libre comme destinataire. Les LDCs envoient leurs sous-événements à ce GDC qui les assemble en un événement complet, les introduit dans une structure de données Root (voir la Section 4.2 plus loin pour une description de Root) et les copie sur le disque tampon au centre de calcul pour être sauvegardés sur bande magnétique (en anglais *Permanent Data Storage*, PDS). Toutes les communications réseaux utilisent le protocole standard TCP/IP [31]. En cas de panne de la liaison entre la zone de l'expérience et le centre de calcul, un serveur de disque (en anglais *Temporary Data Storage*, TDS) est situé près de l'expérience avec une capacité suffisante pour absorber le flux de données pendant quelques heures de collisions d'ions lourds.

Le logiciel qui gère le système d'acquisition des données (*Data Acquisition and Test Environment*, DATE [32]) a été entièrement développé par le projet Online de ALICE. Il s'appuie sur une infrastructure distribuée, gérée par les interactions des processus qui tournent sur des plateformes UNIX. Les caractéristiques principales du système d'acquisition des données sont l'utilisation de composants et de protocoles standards, sa flexibilité et son potentiel de croissance permettant de faire face à d'inévitables nouvelles exigences.

Avant sa mise en fonction, le système avait été extensivement simulé pendant sa réalisation, et des campagnes de test en conditions réalistes ont été effectuées chaque année (en anglais *Data Challenges*).

Le système de déclenchement de haut niveau (en anglais *High Level Trigger*, HLT) a pour but d'augmenter la quantité d'événements rares qui peuvent être récoltés par le détecteur. Bien que le système de déclenchement puisse identifier, au moins dans certains cas, les candidats pour les événements rares, le temps à disposition pour prendre une décision est limité, ce qui à son tour limite la sélectivité et augmente la contamination de l'échantillon avec des « faux positifs » (en anglais *fakes*). Cela a pour effet d'augmenter le flux des données, qui est néanmoins limité à $\sim 4 \text{ Go s}^{-1}$, ce qui amène à la perte d'événements rares. Pour pallier ces problèmes, le système HLT est constitué d'une grande ferme d'ordinateurs qui reçoit les données en temps réel pour pouvoir accomplir trois fonctions. Avec sa grande puissance de calcul, l'HLT peut reconstruire les traces et donc identifier avec plus de précision les événements rares (fonction de déclenchement). Deuxièmement, dans un événement rare, par exemple contenant une désintégration électronique d'une résonance à saveurs lourdes, l'HLT peut sélectionner pour écriture seulement la partie « intéressant », c'est-à-dire la partie « autour » des deux traces (en anglais *Region Of Interest*, ROI). Enfin, l'HLT peut « comprimer » les données très efficacement, par exemple en transformant les informations brutes des détecteurs trajectographiques du baril en parties de trajectoire, ou bien en appliquant des algorithmes de compression avec ou sans perte d'information, qui demandent une grande puissance de calcul. A partir de la prise de donnée des collisions Plomb-Plomb 2011, le HLT est utilisé pour effectuer la reconnaissance des grappes de signaux de la TPC définissant les points de passage des particules (en anglais *clusters*). Le remplacement des

signaux bruts par les paramètres des clusters permet un gain d'espace d'un facteur quatre pour les signaux de la TPC et d'un facteur plus de deux pour les événements centraux Plomb-Plomb. Cela permet d'utiliser la bande passante du système d'acquisition des données pour enregistrer le double d'événements. Un système plus poussé de compression est maintenant à l'étude pour la mise au niveau du détecteur prévue pour les années 2017-2018. Il s'agit de reconnaître les clusters de la TPC, d'exécuter une reconstruction en-ligne et de garder uniquement les clusters qui ont une grande probabilité d'appartenir à une trace « intéressante ». Les études préliminaires conduites indiquent qu'une compression d'un facteur 20 est possible avec cette méthode.

L'HLT est le seul système où une décision de déclenchement peut être prise en temps réel en connaissance de l'information de tous les détecteurs et basée sur des « observables » physiques tels que l'impulsion et le type des particules.

Il est évident que l'utilisation du HLT amène à la « perte » d'une partie de l'information pour garder l'information « intéressante » dans un budget de largeur de bande (en anglais *bandwidth*) limité. L'information perdue ne peut en aucun cas être récupérée, et donc il est important de s'assurer que les décisions du HLT sont celles auxquelles on s'attend et que de l'information intéressante n'est pas perdue parce que les algorithmes de sélection sont imparfaits, ce qui amènerait à introduire un biais. Pire encore, l'information peut être perdue simplement parce qu'elle est « inattendue », c'est-à-dire pas prévue par la théorie, ce qui amènerait à manquer une découverte scientifique. Pour cela le HLT peut fonctionner en trois régimes. « Régime A » (en anglais *Mode A*), le HLT est simplement éteint. « Régime B » (*Mode B*), le HLT fonctionne « en parallèle » au système d'acquisition classique : ses décisions ainsi bien que les données « comprimées » qu'il produit sont enregistrées avec les données normales. « Régime C » (*Mode C*), les décisions de déclenchement du HLT sont respectées et les seules données enregistrées pour les détecteurs qui y participent sont celles qui sortent du HLT. Évidemment le Régime B est utilisé pour vérifier le fonctionnement du HLT en comparant ses décisions avec le flux effectif des données.

Du point de vue architectural, la ferme HLT est composée d'un millier d'ordinateurs PC commerciaux connectés en une ferme à multiprocesseurs symétriques (en anglais *Symmetric Multi Processing*, SMP) où tous les processeurs (nœuds) sont du même type. Un certain nombre de ces ordinateurs ont une interface D-RORC dans leur bus PCI pour recevoir les données provenant des détecteurs par le biais d'un autre D-RORC, comme expliqué plus haut. D'autres nœuds ont des interfaces D-RORC de sortie qui injectent les données dans des LDC pour les écrire. Du point de vue du DAQ, le HLT est un détecteur qui injecte ses données dans des LDC. La ferme est organisée selon une hiérarchie qui reflète la structure physique du détecteur. Dans le cas de la TPC, les nœuds du premier niveau traitent les sous-secteurs en parallèle pour l'identification des points de passage des traces (en anglais *clusters*), les nœuds du deuxième niveau identifient les traces dans chaque secteur, les nœuds du troisième niveau joignent les segments de trace provenant de secteurs ou détecteurs différents et les nœuds du quatrième niveau opèrent la décision finale de déclenchement. La communication entre les différents algorithmes est

réalisée par accès direct et partage de la mémoire, et la gestion du flux des données est effectuée par un mécanisme de souscription-publication (en anglais *publisher-subscriber*), où chaque algorithme souscrit à différents producteurs et ne démarre que les producteurs qui lui ont notifié la présence de données. Lui-même est producteur et il notifiera à ses souscripteurs quand il aura fini de produire ses données. Les fonctions de contrôle et de transmission des données sont donc séparées et le système est piloté par les données (en anglais *Data Driven*).

Le logiciel du HLT a deux composantes. La première est l'infrastructure de communication qui réalise l'échange des données, la communication entre les processus et le pilotage du système par les flots de données. La deuxième partie est composée par les algorithmes de reconstruction des traces, d'identification des particules et d'analyse. Cette partie est intégrée dans le logiciel de traitement des données hors-ligne (AliRoot).

L'environnement en-ligne de ALICE est complété par deux autres systèmes : le système de contrôle de l'expérience (en anglais *Experiment Control System, ECS*) et le système de contrôle du détecteur (en anglais *Detector Control System, DCS*).

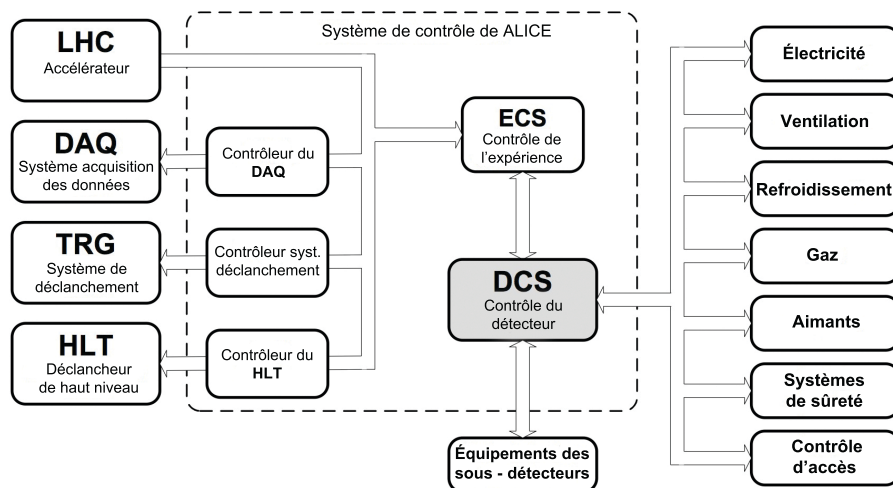


Figure 37 : Plan du système de contrôle de l'expérience ALICE.

Comme on peut le voir dans la Figure 37, le système DCS remplit plusieurs fonctions de contrôle et d'archivage pour les sous-systèmes qui composent le détecteur et pour les systèmes « externes » qui sont indispensables pour le fonctionnement du détecteur. Le rôle du DCS vis-à-vis des détecteurs qui composent ALICE est de contrôler différents paramètres tels que les réglages des alimentations de haute et basse tension et la configuration de l'électronique d'entrée (FEE) par le biais d'un dispositif virtuel d'entrée (en anglais *Front End Device, FED*) qui communique avec un protocole appelé DIM. Le DCS surveille et archive un grand nombre de paramètres. Quand un de ces paramètres sort d'un certain intervalle de valeurs, le DCS initie une série d'actions prédéterminées et il prévient l'opérateur lorsqu'une situation d'alarme est activée.

En plus de l'électronique des détecteurs, le DCS surveille et contrôle une série de systèmes. Plusieurs détecteurs ont besoin d'un flux régulier de gaz. Le DCS contrôle les systèmes d'alimentation et de purification du gaz via le module de

contrôle des gaz (en anglais *Gas Control System*, GCS). Les différentes baies pour l'alimentation électrique des éléments du détecteur sont aussi sous contrôle du DCS, qui reçoit les informations de l'état de l'alimentation électrique primaire de l'expérience. Le DCS reçoit également l'information sur l'état du système de ventilation et contrôle de la salle expérimentale et de ses annexes, mais il ne peut effectuer aucune action de contrôle. De la même manière, le DCS reçoit les informations sur l'état du courant électrique dans les aimants et l'accès du personnel à la zone expérimentale. De plus, il y a dans la caverne toute une série de détecteurs de conditions environnementales qui ne sont pas liés à un détecteur en particulier, telles que la pression, la température et l'intensité du champ magnétique à différents endroits. Tous ces paramètres sont collectés par le DCS.

Une des fonctions principales du DCS est de garantir la sûreté du personnel et des équipements. Pour cela, une partie majeure du DCS est constituée du système de sûreté du détecteur (en anglais *Detector Security System*). Comme cette partie ne joue pas un rôle dans la manipulation des données, nous nous contenterons ici de mentionner son existence.

L'infrastructure informatique du DCS est basée sur le logiciel PVSSII, qui est un produit commercial pour la supervision de l'acquisition des données et du contrôle (en anglais *Supervisory Controls And Data Acquisition* SCADA). Sur cette plateforme logicielle les expériences LHC ont bâti un socle de logiciel commun pour le contrôle des détecteurs, qui a été ensuite spécialisé pour chaque expérience.

PVSSII est un système distribué et commandé par événement, implémenté en langage orienté objet. Différents composants ont été intégrés au système de base : un automate fini (en anglais *Finite State Machine*), des interfaces aux senseurs matériels et logiciels pour l'acquisition des données sur l'état de la machine et pour la configuration des différents composants et des bases de données pour l'archivage des mesures et des configurations qui peuvent être téléchargées vers les modules électroniques du détecteur. Le système a une interface graphique très riche, non seulement pour la visualisation des données, mais aussi pour la programmation graphique de procédures. PVSSII enregistre les données seulement quand il y a un changement qui excède un seuil prédéfini. Cela a l'avantage que dans des conditions de fonctionnement « normales » le trafic de données est nul. Le même mécanisme est aussi à la base du système de génération et traitement des alarmes, qui signalent des conditions anormales de fonctionnement qui requièrent l'attention d'un opérateur.

Le rôle du système de contrôle de l'expérience (ECS, en anglais *Experiment Control System*) est de coordonner les activités des quatre domaines d'activité (en anglais *Activity Domains*) en-ligne : DCS, HLT, DAQ et Trigger. Les différents domaines d'activités sont indépendants et ils interagissent entre eux et avec le monde extérieur sous le contrôle du ECS. Pendant l'acquisition des données, le détecteur ALICE peut être partagé « verticalement » en plusieurs partitions. Chaque sous-détecteur peut être actif seulement dans une des partitions, même si un détecteur peut agir à la fois comme déclencheur dans une partition et comme détecteur proprement dit dans une autre. Les différents domaines sont actifs, mais indépendants dans les différentes partitions. L'ECS gère aussi les

partitions, leur création et opération, et les relations entre les domaines et les sous-détecteurs associés. Le cœur du système est un automate fini associé à une base de données. Une interface graphique permet de gérer les opérations de l'agent de contrôle des partitions (en anglais *Partition Control Agent*, PCA) implémenté à partir de l'automate fini.

2.6 Le traitement des données hors-ligne

Dans cette section, nous donnons un aperçu du modèle de calcul et de traitement des données de l'expérience ALICE. Ce modèle de calcul et ses composantes, conçus et réalisés par l'auteur et son équipe dans le cadre du projet Offline, constituent le sujet principal de ce travail de thèse et seront donc décrit en détail dans les chapitres suivants. Il nous a cependant semblé important d'en donner d'abord une vue d'ensemble pour compléter ce chapitre sur la description du détecteur ALICE. Comme tout sous-système qui compose le détecteur, la conception du système de calcul hors-ligne de l'expérience ALICE a fait l'objet d'un Rapport de Conception Technique (*Computing Technical Design Report* [33]).

Les principales composantes du calcul pour une expérience de Physique des Hautes Énergies sont la simulation, la reconstruction, le calibrage et l'analyse.

2.6.1 La simulation

La conception d'un détecteur moderne de physique des particules nécessite un énorme effort d'ingénierie. Des millions d'éléments de détection doivent être positionnés avec la précision du micron (10^{-5} cm). Les données sont collectées par des millions de canaux électroniques qui, ensemble, consomment plusieurs dizaines de kW d'électricité. La température doit être rigoureusement contrôlée et plusieurs détecteurs nécessitent un flux de gaz constant, ou une alimentation électrique à très haute tension, etc.

Mais, surtout, les détecteurs doivent satisfaire à des critères très stricts de « performance » physique, c'est-à-dire qu'ils doivent être en mesure de détecter les particules et d'en mesurer les propriétés (position, vitesse, énergie, charge) avec une précision donnée pour obtenir les résultats de physique pour lesquels ils ont été construits.

Pour s'assurer que les détecteurs satisfont les exigences de la physique, avant d'être construit un détecteur est *simulé* par ordinateur, parfois pendant de nombreuses années. Les programmes de simulation employés sont des programmes complexes permettant de simuler les collisions entre particules élémentaires dans les conditions expérimentales prévues. Ces programmes sont ensuite capables de « transporter » les particules produites lors de la collision à travers la matière du détecteur. Les moindres détails de l'interaction du rayonnement nucléaire avec le matériel du détecteur sont reproduits, avec leurs fluctuations statistiques intrinsèques, par des méthodes dites de MonteCarlo. La réponse des éléments sensibles au passage des particules est reproduite, ainsi que le traitement électronique prévu pour les signaux électriques générés par les éléments des détecteurs, et enfin ces signaux sont numérisés et stockés dans des fichiers qui ont le même format que les fichiers générés par le détecteur. Des millions d'événements sont ainsi « simulés » pour évaluer les performances du

détecteur et développer les programmes de reconstruction et d'analyse des données. Comme les particules injectées dans le détecteur sont connues, il est possible d'évaluer avec quelle précision les programmes de traitement des données sont capables de « reconstruire » les paramètres et les identités des particules à partir des signaux enregistrés. Cela est une aide importante à la conception et à la compréhension des détecteurs.

Une fois que le détecteur est construit, la simulation reste indispensable, car elle permet d'estimer l'efficacité du détecteur pour les différents signaux qui font l'objet d'études. Pour chaque type d'événement observé, la simulation permet d'évaluer le taux d'événements « perdus » pour différentes raisons telles que les limitations géométriques du détecteur, la concurrence avec d'autres signaux (le « bruit de fond » ou simplement « fond ») ou encore les limitations intrinsèques de précision des détecteurs. Ainsi sont évaluées les « corrections » à appliquer au taux observé d'événements pour estimer le taux effectif.

La question importante est évidemment de savoir jusqu'à quel point la simulation est fiable. En effet, la « validation » des programmes de simulation est une activité fondamentale, qui fait l'objet d'un programme spécifique auquel sont dédiées d'importantes ressources humaines et matérielles. La simulation est typiquement une activité limitée par la disponibilité de ressources de calcul plutôt que d'entrée et sortie (en anglais *CPU-bound*), du moment que les données d'entrée se limitent à la description de la configuration du détecteur.

2.6.2 La reconstruction

Le résultat de l'activité d'un détecteur est un ensemble de signaux électriques générés par le passage des particules à travers la matière, les « données brutes » ou RAW. Le traitement de ces données visant à déterminer les caractéristiques des particules qui ont généré les signaux, à savoir leur masse, charge, énergie et direction, s'appelle la « reconstruction ». C'est une étape fondamentale avant l'analyse qui a pour but de comparer les résultats des collisions avec la théorie, aboutissement du travail des chercheurs. Dans un certain sens, ce processus a des points communs avec le traitement d'images, puisque les signaux sont localisés dans l'espace et le temps, et qu'ils constituent une sorte de matrice de « pixels » à partir de laquelle sont identifiées les traces, c'est-à-dire la trajectoire d'une particule à travers l'espace-temps. Ce processus est aussi appelé en anglais « tracking » ou reconstruction des traces.

La reconstruction est une procédure dont les modalités et les résultats dépendent du type de détecteur. Les détecteurs dits trajectographiques privilégient la détermination de la position des particules. Si ces détecteurs peuvent également mesurer la quantité d'énergie déposée, une estimation de la vitesse de la particule est possible grâce à la formule dite de Bethe-Bloch qui relie ces deux quantités. Si le détecteur trajectographique est immergé dans un champ magnétique, il sera possible de déterminer en plus l'impulsion de la particule, via la mesure du rayon de courbure de la trajectoire qui est inversement proportionnelle à l'impulsion de la particule et directement proportionnelle au champ magnétique. Ces deux mesures peuvent fournir une indication sur la masse de la particule et donc sur son identité, le signe de la charge étant déterminé par le sens de la déflexion par rapport à la direction du

champ magnétique. Ces détecteurs ne sont pas en mesure de détecter directement les particules neutres, mais seulement par la détection de leurs produits de désintégration.

Les détecteurs du type dit « calorimétrique » ou « calorimètres » privilégient la mesure de l'énergie en absorbant plus ou moins complètement la particule détectée. La résolution spatiale et angulaire est normalement moins bonne que pour les trajectographes, mais, en revanche, les calorimètres sont sensibles aux particules neutres aussi bien qu'aux particules chargées. Ces détecteurs peuvent mesurer des flots d'énergie et ils contribuent à la caractérisation des traces en association avec les détecteurs trajectographiques. Parfois il n'est pas possible de distinguer les différentes particules composant un flot d'énergie (jet) dans un calorimètre. Cependant, dans le cas de la détection des jets, la capacité des calorimètres d'intégrer naturellement l'énergie de plusieurs particules qui appartiennent au même jet peut se révéler un atout pour leur détection.

Hormis ces deux familles de détecteurs, des systèmes spécialisés mesurent le temps de vol ou l'impulsion des particules en utilisant des effets tels que l'effet Tcherenkov, ou permettent de discriminer les électrons des autres particules grâce à la radiation de transition. Nous en avons vu des exemples dans la description des composants du détecteur ALICE.

L'aperçu très rapide de la typologie des détecteurs que nous venons d'exposer a pour but d'illustrer le fait que la reconstruction est habituellement une procédure très complexe où les informations de différents types de détecteurs sont associées grâce à des procédures statistiques sophistiquées. Les résultats de ce processus sont les traces, c'est-à-dire un ensemble de paramètres d'identification d'une trajectoire dans l'espace avec leurs incertitudes statistiques et les corrélations entre eux, ou l'identification des jets de hadrons produits par la fragmentation d'un parton créé dans la collision.

Les traces et les jets, ainsi que d'autres informations auxiliaires résultant de la reconstruction, sont stockés événement par événement dans des fichiers appelés Données de Récapitulation de l'Événement (en anglais *Event Summary Data*, ESD). Habituellement, les données brutes sont reconstruites à plusieurs reprises, à mesure que la connaissance et la compréhension du détecteur, ainsi que les algorithmes de reconstruction, s'améliorent. Selon la complexité des algorithmes déployés, la reconstruction peut être une activité limitée par l'entrée et sortie (en anglais *I/O bound*) ou par la capacité de traitement (en anglais *CPU bound*).

2.6.3 Alignement et calibrage

Deux éléments très importants dans la reconstruction sont l'alignement et l'étalonnage (calibrage) des détecteurs. Chaque élément du détecteur est caractérisé par une relation spécifique entre l'intensité du signal émis et la quantité physique qui est à l'origine du signal (énergie, temps, vitesse, charge). Pour obtenir une précision optimale, tous les éléments de détection doivent être « calibrés » en vue de donner le même signal en réponse à la même quantité physique à mesurer. Il faut ici distinguer entre calibrage relatif, qui a pour but de s'assurer de l'uniformité de la réponse des éléments d'un détecteur mesurant la même quantité physique, et calibrage absolu, visant à établir la relation

fonctionnelle entre le signal du détecteur et la valeur de la quantité mesurée. L'ensemble des procédures visant à établir les paramètres de calibrage est appelé calibration. C'est une partie du traitement des données qui peut s'avérer très complexe et qui peut difficilement être formalisée théoriquement. Seules la connaissance des processus de détection spécifiques de chaque détecteur et l'expérience peuvent guider le physicien à obtenir les performances optimales du détecteur. C'est un processus d'apprentissage très long, qui peut durer des années pendant lesquelles la précision des mesures fournies par le détecteur augmente au fur et à mesure que la compréhension du processus de calibrage progresse.

Le calibrage des détecteurs se fait à partir de plusieurs sources d'information. Chaque module du détecteur est testé en l'exposant à un faisceau d'énergie et composition connues (faisceau de test ou, en anglais *Test Beam*), et cela permet d'obtenir une première série de paramètres de calibrage. Certains détecteurs ont un mécanisme pour injecter un signal déterminé, par exemple sous forme d'une impulsion laser pour déterminer le calibrage relatif et la stabilité du détecteur pendant la prise des données.

La détermination précise de la position spatiale des signaux créés par une particule dépend de la connaissance de la position des éléments de détection. Celle-ci est en principe déterminée par la géométrie du détecteur, mais la position réelle de chaque élément de détection peut différer légèrement de sa position idéale. Après la construction du détecteur, la position de chaque élément est déterminée avec les méthodes de la topométrie de précision et de la métrologie géodésique. Cette procédure est appelée « alignement ».

Mais les procédures que nous venons de décrire brièvement ne permettraient pas de parvenir à la précision ultime possible avec un détecteur sans l'utilisation des données elles-mêmes. En effet, les étapes finales aussi bien du calibrage que de l'alignement sont effectuées à partir des données en utilisant des procédures récursives. Cela nécessite parfois des sessions de prise de données dédiées, ou des événements spécifiques qui sont mélangés aux autres événements. Nous en donnerons ici quelques exemples.

Dans le détecteur ALICE, nous avons des prises de données sans champ magnétique où les traces sont des lignes droites. La minimisation des « résidus », c'est-à-dire des distances des signaux à la trajectoire de la particule qui les a générés, obtenue par le déplacement des éléments de détection, permet d'en déterminer avec précision la position réelle. C'est un problème de minimisation sous contrainte avec parfois plusieurs milliers de degrés de liberté qui demande des procédures spécifiques et des ressources de calcul non négligeables.

Nous avons vu que la TPC intègre un système laser qui, pendant la prise des données, permet d'injecter des rayons laser dans le gaz du détecteur. Ces événements « laser » génèrent des signaux sur des plans à une distance connue du plan des électrodes dans la direction de dérive des électrons. Cela permet d'évaluer l'évolution de la vitesse de dérive pendant une prise de données, et donc d'utiliser pour chaque événement la valeur la plus appropriée de ce paramètre pour la reconstruction des traces.

La mesure des masses connues avec grande précision, telles que celles du pion ou du kaon, peut aussi fournir des indications sur le calibrage d'un détecteur.

Alignement et calibrage sont des procédures spécifiques de chaque détecteur qui se développent graduellement avec l'expérience et la compréhension du détecteur.

2.6.4 L'Analyse

Une fois identifiés les traces ou les jets, et déterminée leur précision statistique, l'analyse peut commencer. Dans son essence, c'est un processus dans lequel les résultats expérimentaux sont comparés avec la théorie pour la vérifier ou la falsifier. C'est également lors de l'analyse que des phénomènes inattendus sont découverts, nécessitant de nouveaux développements théoriques pour leur explication. Le premier passage d'analyse s'effectue à partir des ESD, et en même temps les informations sur les traces et les jets sont écrites d'une façon plus compacte dans des fichiers appelés Données des Objets d'Analyse (en anglais *Analysis Object Data*, AOD), qui seront utilisés dans les passages successifs d'analyse. Cette procédure est appelée « filtrage » parce que seule une partie des informations contenues dans les ESDs est écrite dans les AODs. Grâce à cela, les AODs sont beaucoup plus petits que les ESDs et ils peuvent être lus et analysés plus rapidement. Comme le processus de filtrage implique des choix qui dépendent de la physique et parfois même du type d'analyse, il est courant de répéter le filtrage pour optimiser la statistique au fil du progrès du processus d'analyse.

Il est important de mentionner ici les fichiers des « étiquettes » (en anglais *Tag Files*). Ce sont des fichiers qui contiennent un résumé très restreint pour chaque événement, ainsi que le fichier où se trouve l'événement. Il y a des étiquettes pour les RAW, les ESDs et les AODs. La fonction des étiquettes est de permettre d'accéder rapidement à une certaine sélection d'événements sans devoir lire en entier les fichiers correspondants.

Lors de l'analyse, les informations contenues dans les ESDs ou les AODs sont utilisées pour produire des distributions statistiques mono- ou multi-dimensionnelles. L'identification des événements intéressants se fait via la sélection sur la valeur de certaines variables calculées à partir des paramètres des traces, qui, dans le jargon du physicien, sont appelées « coupures ». Au-delà de l'identification des classes d'événements intéressants, l'activité d'analyse vise à déterminer la fréquence avec laquelle ces événements se produisent. C'est une opération très délicate, car l'efficacité du détecteur n'est jamais de 100 % et, en particulier, est différente pour les diverses classes d'événements. Cette mesure est néanmoins fondamentale parce qu'elle permet de déterminer la section efficace de chaque type de réaction, qui est une des prévisions fondamentales de la théorie.

La simulation et la reconstruction sont organisées de manière centralisée, l'analyse, en revanche, est à la fois une activité organisée au niveau central et une activité personnelle (dite aussi « chaotique ») effectuée par chaque physicien qui veut tester une hypothèse ou une idée. L'analyse est habituellement une activité dominée par l'entrée et sortie des données, où les entrées sont les ESDs ou les

AODs et où la sortie est constituée de différents types de distributions statistiques.

Toutes ces activités requièrent une utilisation intensive des ressources informatiques ainsi que le développement de programmes spécifiques et d'une infrastructure matérielle et de gestion spécifique. Dans les prochains chapitres, nous décrirons en détail ces éléments tels qu'ils ont été conçus par l'auteur de cette thèse et réalisés par son équipe au cours des dix dernières années.

2.7 Le processus de traitement des données

Les activités décrites plus haut sont exécutées continuellement sur les données et doivent être coordonnées avec les périodes de prise de données, mais aussi avec les exigences imposées par les conférences de physique. L'élaboration d'un modèle de calcul qui puisse intégrer toutes ces exigences et à la fois garder assez de souplesse pour faire face aux imprévus qui accompagnent toute activité expérimentale a été l'une des activités majeures de l'auteur à partir de 2005.

Comme les autres expériences du LHC, ALICE utilise un modèle informatique distribué hiérarchique et multi-niveaux qui a pour but d'optimiser l'utilisation des ressources informatiques distribuées et d'en équilibrer la charge. Ces ressources sont reliées et rendues accessibles grâce à la technologie de la Grille informatique, que nous décrirons en détail dans le Chapitre 5. Pour la compréhension de ce qui suit, il sera suffisant de noter que la structure de calcul distribué de ALICE comprend environ 80 centres de calcul dans presque autant de pays différents. Le plus important de ces centres est basé au CERN, appelé aussi Tier 0¹⁴. A lui seul il fournit 30 % des ressources informatiques de ALICE. Cinq autres centres de grande taille (Tier 1), dont la caractéristique principale est d'offrir un stockage sur bandes magnétiques, et un grand nombre de centres de taille généralement inférieure appelés Tier 2, complètent le dispositif de la Grille. Tous ces centres sont reliés par la Grille informatique et participent ensemble au traitement des données de l'expérience.

Dans ce qui suit, nous faisons une description du modèle de traitement des données tel qu'il a été défini dans le Rapport Technique sur le Calcul [33]. Ce modèle a été suivi d'assez près pendant les premières années de prise de données : même si par pragmatisme nous avons dû nous en écarter de temps en temps, il reste la référence pour le traitement des données.

Le rôle du Tier 0 est de stocker les données brutes provenant du détecteur, d'effectuer la première reconstruction des données et de stocker les données de calibrage et les ESDs.

Une ferme d'ordinateurs d'environ 150 nœuds travaillant en parallèle grâce au système PROOF (*Parallel Root Facility*, utilitaire pour le calcul parallèle faisant partie du système Root, voir section 4.2) est chargée de la reconstruction rapide, du calibrage et de l'analyse préliminaire des données résidant sur le disque tampon du Tier 0.

Les données brutes, les ESDs et les AODs provenant de la première reconstruction sont également migrés vers les centres Tier 1 pour y être archivés

¹⁴ Ici le nom anglais *tier* prend le sens de « couche d'un modèle de conception ».

sur bande. C'est aussi dans les Tier 1 que sont effectuées les reconstructions ultérieures ainsi que l'analyse planifiée. Les données brutes sont distribuées de façon à en avoir deux copies, une au CERN et une autre dans un des Tier 1. Les centres Tier2 sont utilisés la plupart du temps pour la simulation et l'analyse utilisateur, et ils gardent sur disque des répliques des ESDs et AODs.

Bien que les stratégies de traitement varient entre les données proton-proton et ions lourds, leur point commun est la nécessité de disposer rapidement des résultats du premier passage de reconstruction. Ce processus doit donc démarrer immédiatement après (pour les collisions d'ions lourds) ou pendant (pour les collisions proton-proton) la prise de données. Cette action rapide garantit un potentiel de découvertes précoces et elle permet de déterminer rapidement les conditions de prise de données. Les tâches d'alignement et de calibrage sont effectuées à flux tendu sur les événements stockés sur les disques du Tier 0.

Les données proton-proton collectées, dont la reconstruction nécessite moins de ressources de calcul que les données ions lourds, sont traitées quasi en-ligne pendant la prise des données (sept mois selon notre modèle d'« année standard de prise de données », en anglais *Standard Data Taking Year*) par le Tier 0 au CERN. Les données stockées temporairement sur les disques de l'élément de stockage (en anglais *Storage Element*) du CERN sont calibrées et reconstruites dès qu'elles sont disponibles. Les données reconstruites sont conservées dans le système de stockage à long terme local au Tier 0 et distribuées aux Tier 1s. Les reconstructions suivantes sont exécutées de façon distribuée dans les Tier 1s sur une période de six mois.

En raison de leur taille beaucoup plus importante, la reconstruction en-ligne des données ions lourds nécessiterait des ressources informatiques excessives. La première reconstruction dure entre deux et trois mois et commence immédiatement après la fin du calibrage, pendant la fermeture annuelle de l'accélérateur. Cette première reconstruction doit être terminée bien avant le début de la prochaine prise de données d'ions lourds (au moins six mois avant) pour être en mesure de définir, à partir de l'analyse des données, les conditions expérimentales (réglages des détecteurs et du système de déclenchement). Comme dans le cas des collisions proton-proton, les reconstructions suivantes sont exécutées de façon distribuée dans les Tier 1s sur une période de six mois.

3 Le modèle de développement de AliRoot

L'objectif de l'infrastructure de calcul de ALICE est d'une part de fournir les outils logiciels pour la simulation, le calibrage, la reconstruction et l'analyse des données, et d'autre part de gérer le processus de traitement des données et les ressources de calcul de l'expérience qui permettent de simuler, calibrer, reconstruire et analyser les données.

L'ensemble des codes qui participent au traitement hors-ligne des données de l'expérience fait partie d'une infrastructure logicielle cohérente connue sous le nom de AliRoot [34] et qui sera décrite dans le Chapitre 4.

L'ensemble du logiciel qui gère de façon centralisée le traitement des données dans les différents centres de calcul fournissant des ressources à l'expérience ALICE est connu sous le nom générique de intergiciel (en anglais *middleware*, logiciel du milieu). Une partie de ce intergiciel est développée par des projets internationaux, européens et américains et est commune à toutes les expériences LHC, tandis qu'une autre partie est développée par l'expérience ALICE pour son usage propre. Nous présenterons la Grille informatique dans le Chapitre 5.

Le présent chapitre décrit le développement, la maintenance et le déploiement de AliRoot. Dans un contexte plus général, nous commençons avec des considérations sur le développement des programmes informatiques dans le domaine de la Physique des Hautes Énergies.

3.1 Le logiciel dans la Physique des Hautes Énergies

L'étude des collisions de particules et noyaux à haute énergie requiert la manipulation de grandes quantités de données, ce qui ne peut se faire qu'avec l'aide des ordinateurs. Les programmes utilisés à cette fin ne sont pas disponibles sur le marché et dans la plupart des cas doivent être développés par les physiciens eux-mêmes. Cela représente une activité majeure, en termes de ressources et de coûts, dans le cadre de la préparation et de la mise en œuvre d'une expérience de Physique des Hautes Énergies.

La complexité des équipements expérimentaux d'aujourd'hui et des procédures pour extraire les signaux physiques des données brutes récoltées par les détecteurs nécessite l'écriture de programmes très complexes qui peuvent atteindre plusieurs millions de lignes de code. Ces programmes doivent ensuite être développés et entretenus pendant des dizaines d'années par des équipes distribuées et multinationales, en constant renouvellement.

La Physique des Hautes Énergies partage les problèmes liés au développement et à la maintenance d'importantes suites de logiciels avec d'autres disciplines et champs d'application. Il est donc tout à fait naturel de s'interroger sur les rapports entre la théorie du développement du code, élaborée par la science de l'information, et la pratique des physiciens programmeurs. Un travail sur le logiciel d'une expérience de Physique des Hautes Énergies ne serait pas complet sans quelques remarques à ce sujet.

En général, la Physique des Hautes Énergies réussit très bien à développer ses propres logiciels, même si ce succès est modulé en fonction des différentes

expériences. Malgré ce bilan honorable, la Physique des Hautes Énergies n'a jamais prétendu utiliser, et n'a jamais utilisé à grande échelle, les méthodes du génie logiciel traditionnelles spécifiquement conçues par les experts informaticiens afin de faciliter la production de programmes de grande envergure. La Physique des Hautes Énergies n'a jamais non plus développé une école indépendante de la pensée sur ses pratiques de développement de logiciel. Toutefois, l'opinion de l'auteur est que des méthodes « intuitives » de génie logiciel ont été appliquées sans jamais être formalisées. Certaines d'entre elles ont été novatrices, et même en avance sur leur temps.

La réticence des physiciens programmeurs à reconnaître cette dimension « théorique » de leur travail a créé une relation complexe et peu efficace avec les ingénieurs informaticiens, ce qui est l'une des contradictions caractérisant le calcul dans le domaine de la Physique des Hautes Énergies.

3.1.1 Le génie logiciel classique

Le génie logiciel (en anglais *Software Engineering*) et la « crise du logiciel » qu'il a pour but de résoudre ont été des questions fondamentales au cœur de l'activité de développement des logiciels depuis que le concept de logiciel existe.

La rédaction de grands programmes a toujours été reconnue comme une activité problématique. Déjà en 1958 des travaux sur le sujet [35] ont été présentés lors d'une conférence sur la production de grands programmes informatiques. Il y a quarante ans, une première conférence a même été spécifiquement organisée pour discuter du fait que « les projets de logiciels ne respectent pas les délais, dépassent leur budget, et leurs produits ne sont pas fiables » [36]. Lors de cette conférence, le terme *Software Engineering* (génie logiciel) a été introduit. Le précepte fondamental du génie logiciel est que « les solides méthodes de l'ingénierie doivent être appliqués à la production des logiciels afin de la transformer en une activité prévisible et efficace ». Comme on peut le voir, c'est moins une définition qu'un objectif, d'ailleurs basé sur l'hypothèse douteuse de l'existence et du champ d'application des « solides méthodes de l'ingénierie ».

La littérature sur le génie logiciel est vaste, allant de la gestion et du contrôle des procédés aux méthodes formelles pour démontrer qu'un programme est correct, jusqu'à la psychologie et à la philosophie. En termes assez généraux, la stratégie élaborée par le génie logiciel classique pour atteindre son objectif de prévisibilité et d'efficacité a été de répartir le processus de production des logiciels en une série d'étapes clairement définies et de formaliser le transfert d'informations d'une étape à l'autre.

Malheureusement, il semble inévitable que tout programme, une fois testé dans des conditions réelles, doive être modifié afin de satisfaire les demandes des utilisateurs. Pour faire face à cette réalité, le génie logiciel traditionnel a envisagé une série de stratégies qui ont pour but de limiter au maximum le changement, ou tout au moins de l'encadrer dans un processus très formel et « prévisible », toujours à la recherche d'un parallèle utopique entre ce qui est supposé être la pratique de l'ingénierie et le développement proprement dit du logiciel.

Étant donné la présence inéluctable de « boucles de rétroaction » dans le processus de développement du logiciel, le génie logiciel a essayé d'en minimiser le nombre et la portée en prescrivant aux programmeurs professionnels de

commencer par la description la plus complète et exhaustive possible de leur projet.

Ce processus a tendance à devenir rapidement lourd et à introduire des surcoûts qui peuvent être assez importants. En raison de leur rigidité et de leur formalisme, les méthodes du génie logiciel classique sont connues sous le nom de Procédés de Haute Cérémonie (en anglais *High Ceremony Process*, HCP).

En théorie, cette stratégie pourrait être payante si son objectif, c'est-à-dire la réduction de la fréquence des changements, était atteint. Malheureusement, un tel processus tend à créer une situation où le coût du changement augmente plus rapidement que la réduction de la probabilité du changement. Plus le processus est formel, plus le moindre changement a des coûts élevés. Donc la fréquence des changements est effectivement réduite, à l'inverse des coûts. Si la réalité ne suit pas le « Plan », et c'est pratiquement toujours le cas, le coût de l'élaboration d'une stratégie alternative peut être simplement prohibitif. L'échec du « Plan » laisse alors le projet sans aucun plan, sans guide dans un territoire inconnu où les échelles de temps et les coûts deviennent imprévisibles. Il a été souvent constaté dans la pratique que plus le projet est géré par un plan lourd et détaillé, plus grande est la probabilité que les divergences avec la réalité apparaissent précocement. Cela se traduit par l'aphorisme « trop de processus tue le processus ».

Ce constat de la « faillite » du génie logiciel classique devrait être néanmoins tempéré par une analyse de ses aspects positifs. Même si, à ce stade, nous ne sommes pas en mesure d'identifier une école de génie logiciel qui ait su capturer l'essence du développement des logiciels ou qui ait été capable de proposer une stratégie adaptée à la plupart des cas, nous pouvons affirmer que la réflexion collective sur le développement des logiciels, motivée par les travaux sur le génie logiciel, a été inestimable pour comprendre cette activité qui est au cœur de la révolution de la technologie de l'information et à laquelle elle a certainement contribué à de nombreux égards.

Néanmoins, s'il est vrai que le génie logiciel offre des ouvertures intéressantes dans un domaine très complexe, l'industrie du logiciel est encore à la recherche de stratégies efficaces pour le développement de ses produits. C'est l'opinion de l'auteur que cela est en partie dû au fait que l'activité de développement du logiciel est plus compliquée qu'on ne le pensait au départ, et c'est seulement récemment que les recherches sur ce sujet ont commencé à explorer au-delà de l'analogie avec l'ingénierie. La tentative de concevoir des recettes pratiques, même si c'est nécessaire, sera donc prématurée par rapport au niveau actuel de compréhension.

En réalité, cette affirmation sera plus ou moins vraie selon le domaine d'application du logiciel en question. Dans certains domaines d'application, il y a une chance raisonnable de saisir l'essence du problème et de parvenir à une description assez complète du logiciel, ce qui assurera une certaine stabilité pendant sa production et mise en œuvre. Malheureusement, ces domaines d'application sont rares, et presque aucun ne concerne la recherche dans le domaine de la physique.

L'utilisation des méthodes du génie logiciel classique dans l'industrie a attiré l'attention des physiciens programmeurs. Au CERN, par exemple, chaque nouvelle méthode et formalisme apparus au cours des années ont été essayés. La SASD de Yourdon [37], la méthode entité-relation [38], l'OOADA de Booch [39], l'OMT de Rumbaugh [40], l'OL de Shlaer-Mellor [41], l'ESA PSS-05 [42], l'UML [43]), jusqu'aux dernières méthodes orientées objet. Les outils de support pour ces méthodes (tels que ADAMO [44], I-Logix Statemate [45], OMW [46], Omtool [47], StP [48], Rational Rose [49], ObjecTime [49], pour ne citer que les plus importants) ont régulièrement suscité de l'intérêt pour tomber rapidement dans l'oubli en dépit du temps et des ressources investies.

Les résultats obtenus à l'aide de ces méthodes ont été loin d'atteindre le succès et la robustesse de CERNLIB [50] (et notamment de PAW [51] et GEANT3 [52]), développé au fil des ans avec des méthodes d'« amateur », ou de Root [53], l'héritier spirituel de CERNLIB.

Nous interprétons ce constat par le fait qu'au fil du temps la Physique des Hautes Énergies a élaboré une méthodologie de développement logiciel qui était à la fois suffisante pour ses besoins et raisonnablement efficace. Donc, adopter les méthodes formelles du génie logiciel revenait à remplacer une méthode existante, adaptée à notre environnement bien que jamais formalisée ou simplement décrite, par une autre méthode, suffisamment discutée dans la littérature, mais inappropriée.

3.1.2 Génie logiciel et Physique des Hautes Énergies

Le cœur de toute méthode de génie logiciel est la tentative de contrôler l'impact et le coût du changement. Dans le génie logiciel HCP ce but est atteint par une description complète, exhaustive et aussi détaillée que possible du système avant que le développement proprement dit ne commence, ou du moins avant que le système ne soit déployé. Curieusement, l'idée que les systèmes logiciels puissent être conçus a priori, puis mis en œuvre en suivant le plan initial a séduit plusieurs personnes qui semblent ignorer le fait qu'il s'agit là d'un modèle impliquant une simplification grossière de la réalité.

Même en dehors du monde de la recherche, les clients, les exigences, la technologie et le marché sont en perpétuel mouvement, avec une vitesse de changement qui a tendance à s'accroître au rythme d'Internet et de la mondialisation. Moins prévisible et plus rapide est le changement, moindre sera la pertinence de la conception initiale, et plus nous devons accepter de la changer.

Dans un certain sens, la Physique des Hautes Énergies a anticipé cette situation car elle a toujours vécu dans une situation très dynamique où le changement était inévitable. Certains besoins évoluent et de nouveaux besoins ne se manifestent seulement qu'après le déploiement des logiciels et leur exploitation.

Si le changement est inévitable, l'économie du changement doit être réexaminée. L'axiome qu'un changement dans un projet entraîne une catastrophe doit être contesté via le développement d'une nouvelle forme de génie logiciel qui évite une augmentation exponentielle avec le temps du coût du changement. Dans une telle méthodologie, même le report d'un développement peut être une bonne chose, car la nécessité de mettre en œuvre une certaine fonctionnalité, bien que

prévue en principe, pourrait ne jamais se présenter. Si l'économie du changement est mieux maîtrisée, commencer par un système minimaliste pour ensuite le faire évoluer pourrait bien être une stratégie gagnante, beaucoup plus efficace que de dépenser beaucoup de temps à concevoir le système complet alors que les besoins sont encore mal connus.

L'histoire du développement du logiciel laisse à penser qu'au fil des ans la Physique des Hautes Énergies a développé une méthode originale qui, compte tenu des contraintes et des objectifs, a remporté un succès certain en étant manifestement capable de maîtriser le coût et la dynamique du changement.

Pour ne citer qu'un exemple, les expériences de l'accélérateur Large Electron Positron (LEP) ont été conçues dans les années 70 quand les ordinateurs étaient des « mainframes » (ordinateurs centraux de grande taille) et quand l'ensemble des besoins était estimé inférieur à la puissance d'un seul ordinateur portable actuel. Pourtant les logiciels de traitement de données de LEP ont été portés sur toutes les plateformes de calcul disponibles pendant une période de plus de vingt ans qui a vu la fin des mainframes, l'invention d'Internet et de la toile, l'introduction des « minis » et « superminis » tels que les VAX¹⁵, l'invention des postes de travail personnels comme les Apollos et enfin la révolution due à l'apparition du « Personal Computer » (ordinateur personnel) sous Windows et Mac OS.

Pendant toutes ces années les physiciens travaillant aux expériences du LEP ont vu leur investissement « protégé » et ils ont pu, en même temps, exploiter presque sans interruption chaque nouvelle avancée technologique en termes de matériel et logiciel de calcul.

Ce qui est difficilement appréciable, c'est que la méthode de développement du logiciel propre à la Physique des Hautes Énergies, bien qu'appliquée avec succès, est utilisée « instinctivement » et transmise par voie « orale » sans jamais avoir été formalisée. Cette situation est loin d'être idéale, car la connaissance doit être formalisée afin d'être correctement transmise et qu'elle puisse évoluer.

La principale caractéristique de cette méthode de développement est l'absence apparente de projet. En effet, les projets de logiciel les plus réussis ont eu pour point de départ une idée originale très précise qui a ensuite guidé le développement. Développeurs et utilisateurs « savent ce qu'ils veulent » car ils « savent ce dont ils auraient eu besoin dans le passé. » Cela peut sembler tout sauf précis, mais au grand dam des tenants des méthodes HCP, cette méthode est aussi précise qu'elle peut l'être sans imposer de contraintes inutiles et sans inventer des détails qui très probablement seront amenés à devenir rapidement obsolètes.

Le « scénario » et le but sont clairs, bien que parfois difficiles à exprimer et formaliser. Le développement est guidé par une « idée » partagée, qui trouve son origine dans l'expérience collective de la communauté et non dans un « plan » formel.

Cette idée initiale, presque une métaphore, devient propriété de la communauté des physiciens utilisateurs, et guide l'évolution du système. Le code est écrit en

¹⁵ Ordinateur introduit à la fin des années 70 par Digital Equipment Corporation.

général rapidement et en conformité avec les objectifs du projet, et il est rapidement livré à la communauté des utilisateurs, qui sont très actifs à l'essayer car il répond à leurs besoins immédiats.

Ensuite, le développement procède par cycles, les caractéristiques les plus importantes étant celles sur lesquelles les développeurs travaillent d'abord parce qu'elles sont les plus demandées par la communauté. Dans la plupart des cas, la dynamique tire le projet dans la bonne direction car il y a une culture commune et un objectif partagé par la communauté qui était contenu dans l'idée initiale.

Les utilisateurs expriment ouvertement leur opinions, et, le cas échéant, leur mécontentement, mais de manière constructive, et ils continuent à utiliser le produit. Les développeurs sont à l'écoute et travaillent sur les fonctionnalités les plus demandées. Les utilisateurs les plus avancés fournissent directement des parties du code, suivant le style et la philosophie du produit, qui la plupart du temps peuvent être simplement « insérées », sans changement ou presque, pour faire partie de la base commune du code. Tous les développements sont mis en commun sur la base de la philosophie « Open Source » (code source libre [54]) et la communauté se sent collectivement responsable pour le logiciel. Les développements parallèles se rejoignent souvent pour donner naissance à un seul produit et, lorsque le besoin s'en fait sentir, une restructuration ou même une refonte massive est mise en route. Les nouvelles versions apparaissent assez fréquemment, et la version en cours de développement est constamment à la disposition des utilisateurs cobayes ou de ceux qui ont désespérément besoin des nouvelles fonctionnalités.

La communauté se sent responsable et guide le développement du logiciel. Le progrès est basé sur le dialogue et le respect mutuel. L'essentiel de la communication se fait par courriel et il n'est pas rare que certains des principaux développeurs ne se rencontrent que quelques fois dans leur vie. Des refontes majeures sont réalisées avec le souci de protéger l'investissement des utilisateurs, tout en ouvrant la voie à des évolutions ultérieures.

Ce n'est pas la description d'un avenir de rêve. Cela a été l'histoire de l'évolution de la bibliothèque des programmes du CERN (CERNLIB [55]) de la fin des années 70 au début des années 90 quand elle a été mise hors exploitation, et c'est maintenant le quotidien du développement de Root.

Il est intéressant de noter que d'autres produits pour lesquels une description complète des besoins a été réalisée avant de commencer le développement, soit n'ont jamais vu le jour (GEM dans les années 70) soit ont connu des difficultés et des retards (par exemple GEANT4 [56]) ou encore ont été simplement rejetés par les utilisateurs (par exemple LHC ++ [57]).

3.1.3 Le cycle de développement

Il ne fait aucun doute que, dans le processus de développement du logiciel en Physique des Hautes Énergies, l'exigence de stabilité est transférée du produit lui-même au processus qui génère le produit, dont la procédure de publication est le cœur.

Ainsi, par exemple, les utilisateurs ne s'attendent pas forcément à ce que le produit satisfasse toutes leurs exigences, mais que le processus de soumission de nouvelles exigences et de suivi de ces exigences soit prévisible et fiable. Ils sont même disposés à accepter un certain degré de « frustration » dû à d'éventuels bogues, à condition que leurs réactions et remarques soient rapidement et sérieusement prises en compte. C'est en quelque sorte une « virtualisation » de la production du logiciel, où le centre d'intérêt se déplace du code au processus de production.

Un des points délicats de la gestion de ce processus est la publication (en anglais *release*) de nouvelles versions d'un produit (y compris la première!). Comme les exigences ne sont pas totalement définies au début du développement, au fur et à mesure que la compréhension des besoins s'améliore et que des bogues sont découverts et corrigés, de nouvelles versions du logiciel, parfois avec des modifications substantielles, sont appelées à être publiées.

Dans le cas du développement du logiciel, l'adoption de la part des utilisateurs d'une nouvelle version est facilitée par une « suspension momentanée » de la naturelle défiance envers toute nouveauté qui pourrait perturber le travail en cours. L'exploitation de cette « confiance momentanée » (dans le fait que la nouvelle version soit meilleure que l'ancienne) est au cœur d'une publication réussie.

La première version revêt une importance toute particulière. Si elle intervient trop tard, après que le produit inclut déjà une fonctionnalité assez riche, d'une part, des utilisateurs peuvent avoir trouvé un produit alternatif et sont donc définitivement perdus. D'autre part, le produit étant déjà bien avancé, des changements de direction majeurs, suite aux commentaires des utilisateurs, pourraient être difficiles à mettre en œuvre. Cependant, une publication trop précoce peut résulter en un produit immature avec trop peu de fonctionnalités par rapport aux attentes et à l'utilisation prévue. Les utilisateurs seront déçus et, par conséquent, ne l'essayeront qu'une seule fois. Trouver le bon moment pour publier un produit dépend d'un certain nombre de facteurs dont l'appréciation aura une grande influence sur la réussite ultérieure du projet.

Il est évident que pour satisfaire les utilisateurs il est nécessaire de pouvoir introduire rapidement les changements demandés dans un processus très dynamique. Ainsi une des règles d'or du développement dans la Physique des Hautes Énergies est « publiez tôt, publiez fréquemment ». Le « schéma directeur » à suivre est remplacé par une adaptation dynamique aux exigences des utilisateurs et par un suivi « continu » des modifications apportées grâce à des publications fréquentes de nouvelles versions avec des modifications incrémentales tant que faire se peut.

Une telle stratégie de développement exige une attitude d'écoute et de dialogue constante de la part des développeurs. Cela a pour effet de mettre sous pression les équipes qui doivent préserver un équilibre entre la concentration indispensable au développement et l'ouverture d'esprit et l'attitude d'écoute nécessaires pour rediscuter en permanence les choix architecturaux et parfois la direction même du travail. Le danger qui guette est celui de décider une « pause » dans ce processus dialectique en « gelant » la publication de nouvelles

éditions pour satisfaire « pour un bon moment » les demandes des utilisateurs en « travaillant en paix » pour un certain temps.

Cela doit être évité à tout prix car il risque d'engendrer un cercle vicieux. Plus le temps de publication est long, plus le nombre de changements est élevé. Dans une telle situation, il devient vite impossible d'assurer la compatibilité du nouveau système avec l'ancien et de s'assurer de la pertinence et de la justesse des nombreux changements apportés. Cela peut conduire à la nécessité d'exposer les utilisateurs à la nouvelle version en parallèle avec l'ancienne, rendant l'administration plus lourde et l'interaction avec l'utilisateur plus compliquée. Les ressources humaines nécessaires pour maintenir deux versions seront fatalement soustraites au développement de la nouvelle version, qui donc prendra plus de temps à être terminée. Les utilisateurs, se méfiant à raison de la nouvelle version, ne l'utiliseront que peu ou guère, en réduisant leurs commentaires et donc leur contribution au contrôle de qualité, ralentissant encore davantage le processus de mise au point.

En présence d'une telle situation, la solution est souvent contre-intuitive. Il faut en effet accélérer les processus de publication, en réduisant le nombre de modifications entre deux versions pour ménager les développeurs. Le « contrat » avec les utilisateurs sera préservé, car ils continueront à pouvoir se fier au processus, même si le rythme auquel les problèmes sont résolus ralentit, et les développeurs pourront « souffler » et récupérer [58].

Ce modèle de développement peut apparaître plus approprié pour des petites modifications et rendre les grands changements difficiles. Cependant, d'expérience nous avons appris qu'il était possible d'effectuer des refontes même assez importantes du code dans un cycle de publication rapide. Néanmoins, il est vrai que cela requiert une planification très stricte, et une équipe très performante et parfaitement soudée pour pouvoir gérer l'inévitable instabilité qui suit un changement important du code de production.

L'établissement d'un calendrier de publication, indépendamment des exigences et du temps de développement, a souvent été proposé comme gestion d'un projet de logiciel. En effet, l'expérience du projet ALICE Offline, que nous décrirons dans la prochaine partie, est tout à fait positive à cet égard.

Remarquons ici que plusieurs des caractéristiques du processus de publication décrites plus haut pour la Physique des Hautes Énergies sont les mêmes que celles proposées par les théories modernes du génie logiciel et par la communauté Open Source. Nous sommes confrontés à un cas de « découverte indépendante » de ces principes par la Physique des Hautes Énergies, et nous ne pouvons que regretter le manque d'une étroite interaction avec les partisans des techniques modernes de génie logiciel et les gestionnaires de projets Open Source.

3.1.4 Vérification et documentation

Il est important de noter comment la stratégie de développement décrite plus haut s'accompagne d'une stratégie de vérification du code particulièrement efficace. Dans un cycle de publications assez rapide, la vérification du code effectuée par les développeurs ne peut qu'être limitée. Néanmoins, le fait que les modifications d'une version à l'autre sont peu nombreuses rend une vérification

« légère » suffisante. La vraie vérification vient des utilisateurs mêmes qui, grâce à leur confiance dans le processus, ont tendance à utiliser toujours la dernière version et donc à la confronter à la réalité des traitements de données. Souvent, quand un utilisateur reporte un problème dans une partie critique du code, c'est son application qui, adéquatement modifiée, est incluse dans la suite de vérification validant le code avant publication.

Cela n'implique pas qu'un jeu de tests complet serait inutile, mais dans le contexte de la Physique des Hautes Énergies, les ressources nécessaires pour développer un tel système sont mieux employées à assurer l'adaptation rapide du code aux exigences des utilisateurs, en restant attentif aux exemples de test qui sont naturellement fournis par les utilisateurs eux-mêmes. De plus, il faut remarquer qu'une stratégie comme celle décrite ici ne peut marcher que si le code est maintenu à un niveau de qualité générale, suffisamment élevé pour inciter les utilisateurs à adopter la nouvelle version qu'ils anticiperont être fiable.

Il est clair que la rédaction a priori de la documentation n'est pas une option viable pour un tel projet, car les exigences et les fonctionnalités évoluent constamment. En outre, le problème récurrent des ressources limitées impose des arbitrages difficiles entre la rédaction de la documentation et le développement du code.

Le manque de documentation pour les codes de la Physique des Hautes Énergies est souvent considéré comme un fait acquis, mais un examen plus approfondi des grands programmes montre que ce n'est pas nécessairement le cas. GEANT 3, GEANT 4, FLUKA, CERNLIB, PAW, ZEBRA, HYDRA et Root ont tous des manuels d'utilisation très complets. En outre, le code est généralement assez riche de commentaires. Le problème vient du fait que cette documentation est habituellement rédigée uniquement lorsque le programme atteint une certaine maturité et donc de la stabilité.

Les meilleurs exemples de documentation sont ceux issus d'une collaboration entre un ou plusieurs utilisateurs et les auteurs. Les auteurs ne sont pas toujours les mieux placés pour documenter le système, car leur connaissance du produit ne leur permet pas de prendre le recul nécessaire pour apprécier le niveau d'explications nécessaire aux utilisateurs.

Pour les systèmes modernes comme Root, le code est disponible en-ligne sur des pages Web en format hypertexte avec des liens entre les différentes parties du code.

Cette documentation hypertextuelle en-ligne est produite automatiquement par des outils commerciaux tels que Doxygen [59] ou développés par la communauté de la Physique des Hautes Énergies comme la classe THhtml dans Root. Ces outils sont d'une importance primordiale pour des codes orientés objet comprenant plusieurs millions de lignes, car ils fournissent une documentation beaucoup plus puissante et toujours « à jour » par rapport à la version imprimée, les deux étant néanmoins complémentaires.

Un autre outil intéressant à mentionner à cet égard, est l'outil de rétro-ingénierie développé par l'institut FBK [60] pour ALICE. Cet outil analyse le code et il en extrait les diagrammes UML [61] (Unified Modelling Language) représentant

l'état réel de la structure des classes. Bien que UML se soit avéré plutôt infructueux comme outil de conception dans notre environnement, il s'est montré d'une certaine utilité pour illustrer et discuter la conception du code une fois mis en œuvre.

3.1.5 La question du portage du code

S'impose, à ce point de la discussion, une petite digression sur la question du portage du code, c'est-à-dire sur la possibilité de l'utiliser sur différentes architectures matérielles et systèmes d'exploitation. Les collaborations de Physique des Hautes Énergies ont toujours été distribuées et ont ainsi été amenées à exploiter une grande variété de matériel et de systèmes d'exploitations. Les physiciens ont donc été forcés à écrire du code utilisable sur une variété de systèmes. Cet exercice était particulièrement difficile dans les années 70 et 80, quand le code du CERN devait être utilisé sur des machines à l'architecture aussi différentes que l'UNIVAC 1100 avec six « octets » de 6 bits, le CDC 7600 / 6600 avec six « octets » de dix bits, l'IBM 360 avec quatre octets de 8 bits et le VAX 760 avec deux octets de 8 bits mais à l'adressage inversé en mémoire (petit-boutiste). Outre la portabilité des codes, la portabilité des données posait aussi un défi, car toutes ces machines avaient une représentation différente pour les nombres avec virgule flottante. La portabilité du code était en partie facilitée grâce au langage de programmation utilisé, le FORTRAN, dont le standard était bien défini de sorte que, en évitant l'utilisation d'« extensions » spécifiques des différents vendeurs, la partie algorithmique du code pouvait être compilée avec très peu de changements sur les différentes machines. En revanche, le standard du FORTRAN ne spécifiait ni les interfaces avec les systèmes d'exploitation (entrée et sortie, manipulations des fichiers, services pour la date et l'heure, etc.), ni les opérations de bas niveaux, telles que la manipulations de bits individuels, qui dépendaient étroitement de l'architecture de la machine.

Pour « cacher » les différences entre les architectures, les parties « spécifiques » étaient codées à l'aide de procédures de « service » qui avaient, dans la mesure du possible, la même interface et la même fonctionnalité sur les différents ordinateurs. Ces procédures étaient les plus souvent codées en langage assembleur, qui permet un accès direct aux éléments inaccessibles au standard FORTRAN. Il a ainsi été possible de développer des programmes assez « propres », où toute la dépendance de l'architecture était « confinée » à un certain nombre de procédures. Le portage sur une nouvelle architecture se limitait, en principe, au codage de ces procédures, qui pouvaient être appelées à partir du code FORTRAN.

La situation s'est progressivement « normalisée » dans les années 90 avec l'adoption progressive, par tous les vendeurs d'architectures à quatre octets de 8 bits, de la représentation IEEE [62] pour les nombres avec virgule flottante et du système d'exploitation UNIX, avec néanmoins l'apparition des premières architectures à 64 bits. Pendant une brève période le problème du portage semblait derrière nous, notamment pendant la période qui a vu le développement du code des expériences LHC, réalisé principalement sur des machines Intel Pentium à 32 bits, géré par le système UNIX, en ayant confiance dans la portabilité intrinsèque du langage de programmation utilisé (C++).

L'introduction au milieu de la décennie 2000 des architectures à 64 bits (Intel Xeon, Intel Itanium et AMD Opteron) a été à l'origine de sérieux problèmes, car le code avait été écrit sans attention particulière aux problèmes de portage.

Dans ce contexte, l'expérience ALICE a fait exception. En 1998, au début du développement de la suite de logiciel hors-ligne AliRoot, qui sera décrite dans le prochain chapitre, l'auteur a imposé de pouvoir utiliser le code sur un ordinateur AlphaServer avec une architecture et un système d'exploitation à 64 bits (Tru64 Unix). Bien que cela ait été le seul AlphaServer sur lequel AliRoot ait jamais fonctionné, l'exercice a été très utile, car lorsque les CPUs à 64 bits Intel Itanium et AMD Opteron sont apparues, ALICE a été en mesure de porter son code sur ces machines en quelques jours, avec des adaptations mineures de la version Tru64. Ainsi, pendant plusieurs mois ALICE a été la seule expérience capable de fonctionner sur ces machines avant que les autres expériences puissent porter et valider leur code et donc rattraper leur retard.

L'importance d'écrire le code de façon portable ne peut être surestimée. Tout d'abord il est plus que probable que, pendant la durée d'une expérience, le code devra être porté sur différents matériels et systèmes d'exploitations pour des raisons à la fois économiques et sociologiques, en plus de l'obsolescence naturelle des systèmes de calcul. S'assurer que le code marche sur plusieurs machines à un instant donné est un atout pour que le code marche sur la « prochaine » architecture.

Le portage et la vérification régulière du code sur des architectures différentes sont aussi un très puissant instrument pour le contrôle qualité. Certains bogues peuvent rester cachés pendant des années, produisant des résultats erronés sur une architecture, et provoquer une erreur macroscopique et facilement détectable sur une autre.

En Physique des Hautes Énergies, nous avons maintes fois fait l'expérience de la difficulté de porter un code sur une nouvelle architecture s'il n'avait pas été conçu pour être porté, et, peut-être encore plus important, s'il n'a pas été porté et validé sur différentes architectures dès sa première version. Cependant, même la meilleure des conceptions visant à assurer la portabilité n'aide pas si le code n'est pas réellement porté, exécuté et vérifié sur plus d'un système. En ne travaillant que sur un seul système, les spécificités propres à ce système (système d'exploitation et matériel) tendent à se glisser dans le code à l'insu des développeurs, les obligeant à des changements parfois très lourds lorsque la nécessité de porter le code devient finalement inéluctable.

Nous ne pouvons pas ici éviter une très brève digression sur la question de la virtualisation. On appelle virtualisation le processus avec lequel on fait tourner un ou plusieurs systèmes d'exploitation sur une seule machine réelle comme s'il s'agissait de machines distinctes. L'évolution du matériel impose des mises à jour régulières de la version du système d'exploitation, car souvent les nouveaux composants (disques, cartes réseaux, cartes graphiques, etc.) ne sont pas compatibles avec l'ancien système. Grâce à la technique de la virtualisation, il est possible de créer une machine virtuelle qui exploite une ancienne version du système d'exploitation sur une machine avec une version plus récente du système d'exploitation. Le programme qui permet la virtualisation imite (on dit « émule ») le comportement des anciens composants en « masquant » les

nouveaux, et donc l'ancien système d'exploitation ne se rendra pas compte qu'il tourne sur une machine virtuelle et non pas sur une machine réelle. La technique de la virtualisation permet de créer très rapidement de nouvelles machines et de les supprimer sans avoir à réinstaller la machine de base. Elle permet aussi de changer les paramètres matériels virtuels (taille de mémoire, dispositifs d'entrée et sortie, cartes réseaux) sans devoir éteindre l'ordinateur et opérer des interventions physiques. Cette technique est de plus en plus utilisée par sa flexibilité et versatilité.

Dans notre cas, il est courant d'entendre que le problème du portage du code est dépassé par l'utilisation des machines virtuelles. Il serait suffisant de tourner toujours la même version du système d'exploitation dans une machine virtuelle, indépendamment du matériel ou du système d'exploitation « réel ». C'est notre conviction que bien que tentante, les désavantages d'une telle solution dépassent les avantages. D'abord la virtualisation comporte une pénalité en termes de performance qui peut atteindre 10 %. Avec l'évolution du matériel, la perte de performance peut devenir de plus en plus importante avec les améliorations du matériel qui ne sont pas exploitées par l'ancien système. Cela est particulièrement vrai pour la CPU et l'entrée et sortie pour lesquels les compilateurs et pilotes les plus récents sont nécessaires pour obtenir la performance idéale. De plus, les mises à jour des anciens systèmes d'exploitation sont arrêtées à partir d'un certain moment et donc les bogues ne sont plus corrigés. Si un de ces bogues concerne un problème de sécurité, il n'y a pas d'autre alternative que de retirer l'ancien système de la production. En outre, renoncer à porter son code signifie perdre tous les avantages que nous avons mentionnés par rapport à la robustesse et à la découverte de bogues dans le programme. Nous considérons que renoncer à porter un programme crée une situation d'une grande fragilité pour le projet informatique en question.

Malgré cela, on ne peut nier que des changements technologiques pourront donner une véritable place à la virtualisation dans le futur [63]. Pour l'instant, la virtualisation reste une technique très intéressante pour la gestion des machines de test et des serveurs.

3.1.6 Le facteur humain

Il ressort clairement de ce qui précède que la gestion d'un projet de logiciel d'envergure implique toute une série de décisions, qui sont critiques pour son succès, telles que le calendrier de sortie des nouvelles versions et leur contenu, la stratégie d'évolution et, éventuellement, même de refonte du code et bien d'autres. Tous ces choix n'impliquent pas seulement la compétence technique des développeurs, que ce soit en tant que programmeurs ou scientifiques, mais dépendent d'un grand nombre de facteurs qui peuvent être décrits, mais pas facilement quantifiés.

Bien qu'il soit important d'analyser les circonstances qui ont conduit certains projets au succès et d'autres à la faillite, un certain malaise demeure dans la difficulté à définir un processus « à toute épreuve » pour la gestion d'un projet de développement de logiciel.

Le sort du projet semble reposer en grande partie sur les capacités du chef de projet à constituer et à diriger une équipe de développeurs compétents,

partageant la même vision et travaillant en harmonie, capables d'écouter le message émanant des utilisateurs, et souvent même de l'anticiper.

La personnalité des participants semble donc jouer un rôle central, qui ne peut être réduit à un ensemble de règles à appliquer. Si nous revenons à la métaphore du génie logiciel selon laquelle les « solides méthodes de l'ingénierie » doivent être appliquées à la « fabrication » du logiciel pour en assurer le succès, nous pouvons peut-être l'accepter, au prix cependant d'un raffinement de l'analogie. Les grands projets de logiciel peuvent être comparés au travail d'un architecte dans la conception d'un théâtre ou d'un musée. Les « solides principes » de l'art sont une condition nécessaire mais pas suffisante à l'accomplissement d'une œuvre réussie.

Le succès de l'entreprise dépend donc de la créativité des hommes et des femmes engagés dans le travail. La réflexion sur ces questions n'en est pas moins importante, mais confère à la production des logiciels toute la complexité d'une activité créatrice de haut niveau intellectuel telle que la conception industrielle avancée ou la recherche scientifique.

Une autre caractéristique du processus de développement du code réside dans les relations « sociologiques » entre utilisateurs et développeurs, brièvement évoquées précédemment. Le développement du logiciel est généralement géré par un projet, souvent appelé « Offline » (terme l'anglais pour *hors-ligne*) pour le distinguer du logiciel pour l'acquisition des données qui opère *en-ligne* avec le détecteur (en anglais *On-line*). Les scientifiques qui participent à une expérience de physique viennent de différents laboratoires et universités qui collaborent à la construction et à l'exploitation du détecteur. L'une de ces institutions, souvent un grand laboratoire national ou international, accueille l'expérience. Habituellement (mais pas nécessairement), le laboratoire d'accueil a la responsabilité de coordonner le projet Offline et de développer l'infrastructure informatique cadre. C'est à l'intérieur de cette structure que les groupes responsables pour la construction et l'exploitation des différents sous-détecteurs composant le dispositif expérimental pourront insérer leur code pour la simulation, la reconstruction, le calibrage et l'analyse.

Chaque sous-détecteur est en effet une petite collaboration expérimentale en elle-même, avec son projet Offline distinct qui doit travailler en collaboration avec les autres sous-détecteurs. Le point important ici est la répartition géographique des différents groupes dans différents instituts, pays et continents. Les expériences actuelles rassemblent jusqu'à trois mille physiciens. La relation de « projet » est plutôt informelle, les différents instituts étant liés par un protocole d'entente (en anglais *Memorandum of Understanding*) qui n'a pas force de loi. L'expérience elle-même est une entité abstraite sans statut juridique. Le lien réel entre physiciens est l'objectif commun : la construction et l'exploitation d'un détecteur pour la recherche fondamentale.

Dans cette situation, la gestion du projet Offline doit procéder par consensus, et toutes les décisions et les accords n'ont qu'une valeur morale. Non seulement la plupart des chercheurs et ingénieurs qui participent au projet Offline dans les différents instituts n'ont pas de relation hiérarchique entre eux, mais le projet lui-même est défini et géré par une entité (l'expérience) qui n'a pas d'existence juridique.

Cela introduit une grande hétérogénéité de rapports entre individus. À un extrême, certains des chercheurs appartiennent à la même institution et ils ont une relation hiérarchique précise entre eux. D'autres travaillent dans une des institutions de la même organisation nationale de recherche, et donc leurs carrières et titres sont comparables. D'autres travaillent dans des continents différents, et leur seule relation est l'appartenance à la même expérience, qui parfois se trouve dans un troisième continent, hébergée par une institution avec une gestion des carrières aussi différente. La situation est encore plus complexe pour les projets logiciels destinés à plusieurs expériences, tels que Root ou GEANT.

Cette situation pourrait entraîner le chaos total, sauf que, si bien gérée, elle peut se transformer en un système d'organisation du travail flexible et efficace. La grande liberté d'initiative de chaque groupe favorise l'émergence de solutions différentes pour chaque problème. Cela évite le risque d'une décision erronée prise de manière centralisée entraînant l'ensemble du projet dans la mauvaise direction. Bien sûr, éviter divergences et duplications reste une préoccupation constante. Cependant cela doit être réalisé par l'intermédiaire d'un processus dialectique de recherche de consensus, car il n'existe aucun moyen pratique d'« imposer » des décisions.

L'autre effet de cette organisation non uniforme est la structuration naturelle par la communication. Les chercheurs appartenant à la même institution, au même institut national de recherche ou au même pays se rencontrent en personne plus fréquemment et adoptent souvent une position commune au sujet de l'évolution du code. Il en va de même pour des institutions différentes qui, même par delà les frontières nationales, participent à la construction du même sous-détecteur. Ces communautés, qui souvent se recouvrent partiellement, fournissent une structure de communication naturelle qui favorise l'émergence d'accords partiels, préparant des consensus plus larges. Le risque que tout le monde discute avec tout le monde est ainsi évité, tout comme la nécessité de trouver un compromis entre des centaines de points de vue différents. Bien entendu, cette organisation introduit le danger de la formation de différentes « parties » qui peuvent s'opposer à l'échelle nationale, ou des sous-détecteurs, avec une rupture possible de la communication à différents niveaux, ce qui doit être évité à travers les structures décisionnelles formelles mises en place par le projet.

Finalement, il faut mentionner qu'une organisation décentralisée permet, si elle est efficacement gérée, une optimisation des ressources humaines. Les meilleurs développeurs sont rarement concentrés au même endroit, et une organisation qui permet d'exploiter les talents là où ils sont, sans devoir les déplacer physiquement, offre de grandes opportunités.

3.1.7 Génie logiciel moderne

Comme nous avons cherché à le démontrer dans ce qui précède, le développement du logiciel dans la Physique des Hautes Énergies suit un procédé identifiable et plutôt efficace, qui n'a jamais été formalisé ni, à la connaissance de l'auteur, décrit d'une façon exhaustive. Cela constitue un sérieux problème, car du fait de l'absence de manuels ou d'autres matériels didactiques sur la pratique du développement du logiciel, il faut plus de temps pour former des

programmeurs au style de travail de la Physique des Hautes Énergies. Ce problème devient particulièrement grave dans les grandes collaborations modernes, où le taux de renouvellement des étudiants et des jeunes docteurs de recherche (en anglais *postdocs*) est assez élevé. Une description claire et consensuelle des meilleures pratiques du développement des logiciels dans la Physique des Hautes Énergies dans la langue et le formalisme du génie logiciel pourrait aider les jeunes chercheurs à devenir actifs plus rapidement et avec moins d'encadrement. Cela pourrait également fournir une base sur laquelle construire un dialogue entre les professionnels de l'informatique et les ingénieurs logiciels, conduisant à des échanges plus fructueux.

Aussi logique que puisse paraître cette démarche, elle n'a jamais été entreprise. Nous croyons que c'est surtout la conséquence du problème des ressources et de l'échelle de temps. Les ressources, comme nous l'avons vu, sont toujours très limitées dans les expériences de physique. De plus, une telle démarche serait profitable sur le long terme, c'est-à-dire sur une échelle de temps qui va au-delà des exigences immédiates d'une expérience de physique. Il est vrai aussi que les personnes qui ont développé le logiciel de la Physique des Hautes Énergies ont toujours concentré leurs forces sur le développement proprement dit plutôt que sur l'académisation et la transmission traditionnelle de leurs connaissances.

Néanmoins, le temps est peut-être venu d'entamer cette démarche, vu la taille des expériences et l'augmentation de la dimension et de la complexité des programmes. Cela est d'autant plus vrai dans la mesure où, depuis une décennie, de nouvelles tendances sont apparues au sein de la communauté des experts en génie logiciel. En effet, on a assisté au cours des dernières années à une évolution vertigineuse de la dynamique du marché des logiciels, en particulier avec ce qu'on appelle l'e-commerce. Les conditions dans lesquelles les développeurs de logiciels pour les expériences de physique ont toujours travaillé sont les suivantes :

- Mise « sur le marché » des produits dans un délai bref.
- Évolution rapide des besoins de la communauté des utilisateurs.
- Changement du matériel et du système d'exploitation.
- Renouvellement rapide des équipes de développeurs.

Désormais, ces points ne sont plus spécifiques de la recherche fondamentale, mais ils se sont banalisés à l'ensemble du marché des services et produits informatiques. Ce sont aussi les aspects que le génie logiciel moderne tente de saisir et de satisfaire. Les tendances modernes du génie logiciel sont connues sous le nom de « technologies agiles » ou Procédés de Petite Cérémonie (en anglais *Low Ceremony Processes*, *LCP* [64]). Le but de ces procédés est de répondre à l'insuffisance relative des méthodes classiques et à l'évolution rapide du contexte du développement du logiciel. L'idée de base de ces techniques est de s'adapter au changement plutôt que d'essayer de le contrôler, se concentrant sur l'interaction des intervenants (développeurs, gestionnaires, utilisateurs, clients) plutôt que sur la définition d'un processus. Le processus doit être gardé aussi simple que possible, afin de réagir rapidement au changement, considéré comme inévitable. Le développement est basé sur des cycles d'interaction courts

et hautement interactifs, de l'ordre de quelques semaines, avec un dialogue constant avec les clients et les utilisateurs, dans le but de privilégier :

- Les individus et leurs interactions plutôt que les processus et les outils.
- Un logiciel fonctionnel plutôt qu'une documentation énorme.
- La collaboration avec les clients plutôt que la négociation d'un contrat.
- La réponse au changement plutôt que le suivi d'un plan.

Un des exemples les plus connus est celui nommé eXtreme Programming (XP) [65]. XP a été une des premières méthodologies de génie logiciel qui, en suivant les principes exposés ci-dessous, essaie de réduire le coût du changement grâce à la réduction de l'effort de conception initial et à l'introduction d'un processus centré sur le changement lui-même. XP est construit autour de quatre principes simples :

1. Communication : un projet a besoin d'une communication en continu entre utilisateurs et développeurs ; le plan et le code doivent être constamment à jour et compréhensibles.
2. Simplicité : utiliser constamment la solution la plus simple ; une conception simple sera facilement extensible si besoin est.
3. Rétroaction : profiter des commentaires continus des clients sur les systèmes de production ; le développement doit être accompagné par une validation continue de chaque changement.
4. Courage : la conséquence des trois points précédents est la nécessité d'un développement agressif ; chaque fois que la possibilité d'apporter une amélioration au système est décelée, elle doit être saisie sans hésitation.

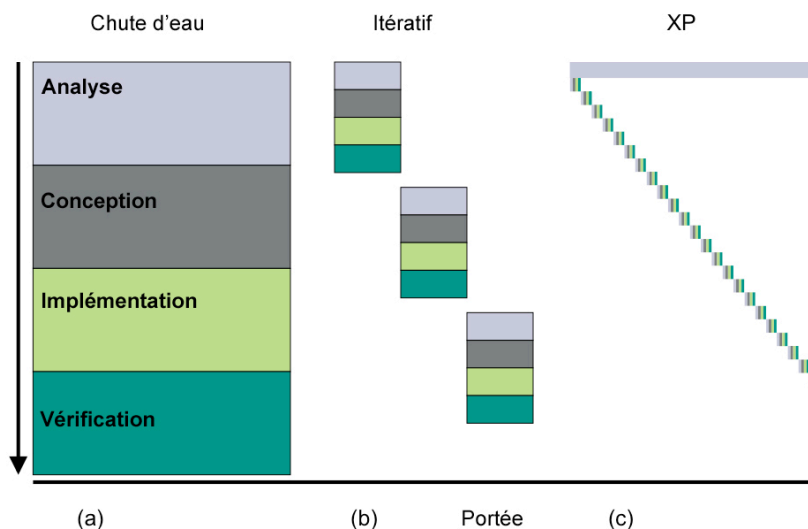


Figure 38 : Synopsis de la programmation XP.

XP reprend l'idée de base de la rétroaction guidée par les utilisateurs que l'on retrouve également dans le génie logiciel classique, et il en fait le centre du processus de développement. Le plan de conception est donc remplacé par un

processus dynamique basé sur des microcycles de rétroaction (voir Figure 38) avec validation et intégration continues des changements.

Les similitudes avec le développement dans la Physique des Hautes Énergies tel que nous l'avons décrit plus haut sont très fortes, même si les conditions dans lesquelles XP a été mis au point sont différentes de l'environnement d'une expérience de physique. Les caractéristiques de la méthode XP et le développement de logiciel dans la Physique des Hautes Énergies ont en commun la conception du système en constante évolution, l'accent mis d'abord sur les fonctionnalités les plus demandées, des temps très courts de « mise sur le marché », la disponibilité du code de développement et l'assurance de qualité effectuée en continu par la communauté des utilisateurs. Il y a cependant une différence importante : les logiciels des expériences de physique sont réalisés par des équipes distribuées et non par des petites équipes de programmeurs travaillant ensemble, comme il est prescrit par XP. Cependant cela ne constitue pas une différence majeure car les méthodes agiles sont des modèles qui doivent être adaptés aux situations spécifiques.

Cette discussion sur les relations et les convergences entre les méthodes de développement du code dans la Physique des Hautes Énergies et les nouvelles tendances de la technologie de l'information ne serait pas complète sans une mention du développement des logiciels « Open Source » [66]. Cet argument mériterait plusieurs pages pour expliquer son importance pratique et philosophique dans l'histoire de la science de l'information, mais par souci de concision nous nous limiterons à une brève digression. La mise à la disposition du public de la toile des logiciels au format source a été l'une des plus grandes révolutions culturelles du siècle dernier. Fondé sur les principes du partage libre et universel de la connaissance, dans le respect de la propriété intellectuelle et sans aucune contrainte idéologique ou politique, le mouvement Open Source a mis à la disposition de quiconque des outils de calcul de qualité égale et souvent supérieure à leurs équivalents commerciaux. C'est le même principe de « démocratie numérique » qui a été à la source du développement de Wikipédia et qui a été à la base du développement de la toile. Le code source développé par le CERN et constituant la librairie de programmes standard des expériences de Physique des Hautes Énergies pendant les quarante dernières années a toujours été distribué librement à toute institution en faisant la demande. La distribution était gratuite pour les institutions académiques et de recherche, et moyennant le paiement d'une modeste somme pour les compagnies commerciales. Nous ne pouvons pas affirmer qu'il s'agissait de « Open Source » avant la lettre, car (encore une fois) l'aspect « philosophique » n'avait pas été développé comme l'a fait la communauté Open Source dans les années suivantes. Néanmoins, la démarche reflétait déjà une volonté de partager librement les développements informatiques dans la communauté scientifique et au-delà.

Il est très encourageant de constater que certaines tendances modernes de la technologie de l'information semblent expliquer et justifier ce que nous faisons dans la Physique des Hautes Énergies depuis de nombreuses années. Cela nous permet enfin de commencer à comprendre pourquoi les choses fonctionnent en dépit des critiques exprimées par des générations d'experts du génie logiciel

HCP. Cette compréhension, si elle est bien exploitée, peut nous aider à améliorer notre processus à travers sa formalisation et sa transmission.

3.2 Le développement de AliRoot

Le logiciel pour le traitement « hors-ligne » des données développé par le projet « Core Offline » de l'expérience ALICE pendant les douze dernières années sous la direction de l'auteur, est identifié par son nom AliRoot [67]. Dans cette section, nous décrirons le processus de développement et la maintenance. Dans le chapitre suivant nous en ferons l'historique et décrirons l'architecture actuelle du système.

Il est important de mentionner que le projet de calcul de l'expérience ALICE est géré par une série de règles qui, proposées par l'auteur, ont été élaborées par le Comité du calcul et approuvées par le Comité de direction de la Collaboration. Ces règles sont publiées sur le site officiel du projet Offline [68], et nous les répetons ici :

- Toutes les données (brutes et traitées ou provenant de la simulation) et tous les codes utilisés pour produire des résultats qui seront diffusés doivent respecter les règles suivantes:
 - Le code et les données doivent être accessibles au public au sein de la collaboration.
 - Le code doit être intégré dans l'environnement AliRoot, sauf dans le cas d'un petit programme autonome.
- Les modalités pour rendre le code ou les données publiquement disponibles et celles concernant l'intégration du code dans le logiciel cadre sont définies en détail par le projet Offline.
 - En cas de désaccord sur l'interprétation ou l'application des règles, le Comité du Calcul est arbitre.
 - Si aucune décision ne peut être atteinte par le Comité du Calcul, la question sera portée à l'attention du Conseil d'Administration de la collaboration.
- La politique informatique de ALICE exige que tout résultat issu de n'importe quelle analyse de données collectées au nom de la collaboration ALICE et qui sont présentées à des conférences, publiées ou diffusées à quiconque hors de la Collaboration, soit produit avec le code officiel ALICE, et que le code et les données soient mis à la disposition de la collaboration entière via des outils de production standard.
- Par résultat, nous entendons un résultat de tout calcul, que ce soit la simulation, la reconstruction ou l'analyse de données réelles ou simulées, effectué par un membre de la collaboration ALICE.

L'organisation du projet Offline et en particulier la procédure de développement du code a été conçue pour permettre l'application de ces règles de façon efficace et rigoureuse.

3.2.1 L'organisation du code

Le code de AliRoot est stocké sur un serveur au CERN et il est disponible en lecture à tout utilisateur d'Internet. Le code est accessible via un logiciel de gestion de versions appelé Subversion ou svn [69]. Le gestionnaire de versions a pour but de gérer l'historique des différentes versions des fichiers sources d'un projet informatique. Le développeur récupère (en anglais *check-out*) une copie du code source sur son ordinateur et il effectue éventuellement des changements. Quand ces changements sont satisfaisants, il envoie la nouvelle version d'un ou plusieurs fichiers sur le serveur (en anglais *check-in*) qui à partir de ce moment est à la disposition de tous.

La particularité du dépôt svn est qu'il garde en mémoire tout l'historique du processus de développement, permettant de récupérer et analyser chacune des plus de 50.000 modifications apportées à AliRoot depuis le début de son développement en 1998. svn offre une grande panoplie de services pour comparer différentes versions, revenir sur une modification, examiner les commentaires qui accompagnent chaque modification et ainsi de suite.

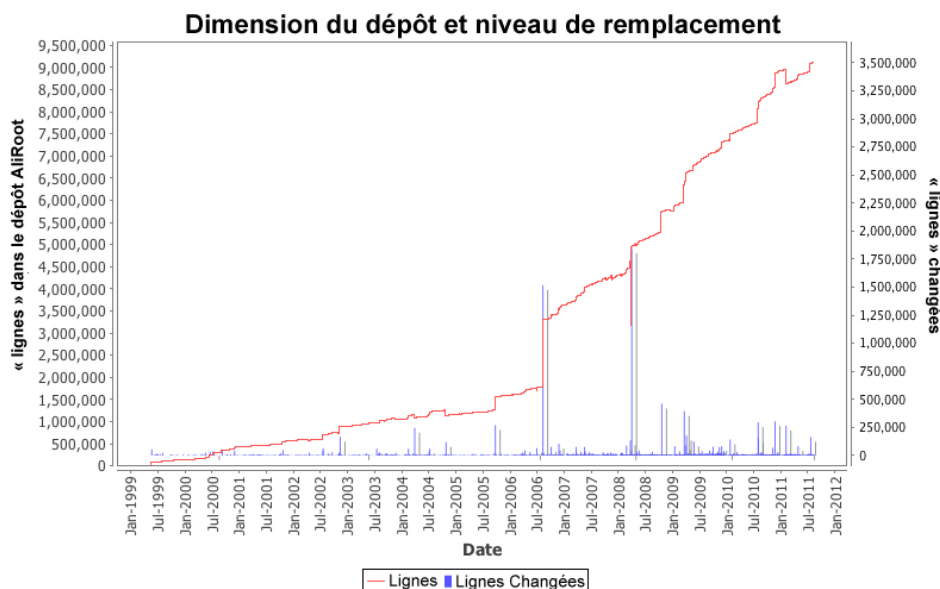


Figure 39: Évolution du dépôt de AliRoot.

Une des fonctionnalités principales d'un serveur svn est de permettre le développement simultané par différents développeurs. Cela est possible grâce à un système sophistiqué de verrouillage. Supposons que deux utilisateurs récupèrent le même fichier pour y apporter des modifications. Si l'un des deux envoie le fichier modifié sur le serveur central, le deuxième ne pourra pas faire de même à moins de récupérer la toute nouvelle version et d'y apporter ses modifications. C'est évidemment là un exemple très simple, et svn est capable de gérer de conditions de « conflit » plus compliquées.

Une autre fonctionnalité très utile offerte par svn est la possibilité de créer des « branches » dans le processus de développement. Supposons qu'un développeur décide de faire d'importantes modifications à un ou plusieurs fichiers source, ce qui peut prendre un certain temps. Pendant ce temps-là, des bogues peuvent être

découverts dans la version originale du code. Il est donc nécessaire que, pendant le développement de la nouvelle version, une copie corrigée de l'ancienne version soit produite et publiée. Dans ce cas, svn offre la possibilité de créer une « bifurcation » dans le code. La nouvelle version des fichiers en cours de modification sera sauvegardée sur une des « branches » tandis que l'ancienne version corrigée sera sauvegardée sur l'autre branche. L'utilisateur a le choix de récupérer une des deux branches selon qu'il veuille essayer le nouveau code ou travailler avec la version corrigée de l'ancien. Évidemment, à un certain moment, il sera nécessaire de « joindre » les différentes branches. svn offre une série très complète d'outils pour comparer et fusionner deux branches. Dans un projet de grande envergure tels qu'AliRoot, l'utilisation des branches est, comme nous le verrons, indispensable, mais seulement si strictement encadrée, car la prolifération de versions parallèles peut rapidement créer une situation chaotique.

		Lignes de code					Années- personne	Coût estimé (M€)
		C++	FORTRAN	C	Total	%		
Ecrit	AliRoot	1.792.586	4.141	1.091	1.804.482	66,3	451,70	42,7
	externe	154.706	753.853	2.063	918.365	33,7	231,38	21,9
Généré	AliRoot	1.413.164			1.413.164		349,51	33,0
	externe	21.477			21.477		4,50	0,4

Table 5 : Tableau récapitulatif du code de AliRoot.

Dans le projet AliRoot, le code est physiquement partagé en plusieurs sous-répertoires, chacun spécifique d'un sous-détecteur, d'un groupe de physique ou d'une application particulière telle qu'un générateur d'événements. Pour chaque sous-répertoire un ou deux administrateurs ont la responsabilité du développement et de la maintenance du code associé, et des développeurs (entre un et cinq) ont le droit de sauvegarder le code sur le serveur. Le nombre de développeurs avec droit « d'écriture » est décidé par les administrateurs. Le nombre total de développeurs qui ont introduit du code dans le dépôt de AliRoot depuis sa création est d'environ 140. Le dépôt de AliRoot contient un peu moins de 9.000.000 de « lignes » (incluant des fichiers de données en format binaire et la documentation). L'évolution de la taille du dépôt svn (Figure 39) indique une progression d'environ 85.000 « lignes » par année de 1999 à 2006, quand les Rapports de Conception Technique des différents détecteurs ont été produits. Ensuite, la progression est passée à 450.000 « lignes » par année. Une analyse plus fine du code a été effectuée avec le programme SLOCcount [70] qui utilise le modèle COCOMO [71] et dont les résultats sont montrés dans la Table 5.

Une partie du code dans le dépôt AliRoot est effectivement rédigée par les membres de la collaboration. Le restant du code sont des programmes externes à la collaboration et « importés » dans le dépôt pour en faciliter la distribution et l'installation. Il s'agit surtout de programmes de simulation utilisés pour le MonteCarlo de transport de particules.

Le programme Root, dont nous parlerons brièvement par la suite, comprend un utilitaire qui, pour chaque classe du code utilisateur, génère des méthodes

auxiliaires pour la visualisation, l'entrée et sortie et le dictionnaire, afin d'ajouter la fonctionnalité de réflexion au code C++¹⁶. Les valeurs reportées dans le tableau donnent une idée de l'économie qu'un tel utilitaire peut apporter, en plus du fait que les méthodes générées automatiquement sont à jour et correctes par définition.

Le chiffre estimé d'environ 452 années-personne pour le développement de AliRoot nous donne une moyenne de 35 développeurs travaillant en continu, ce qui est une estimation raisonnable de l'effort qui a été nécessaire au développement du code. Du point de vue du langage de programmation, on peut remarquer que AliRoot est écrit quasi complètement en langage C++ (99,3 %), alors que les programmes « externes », des codes « hérités » de la génération précédant la transition vers C++ sont majoritairement écrits en langage FORTRAN (82 %).

3.2.2 Le cycle de développement

Le cycle de développement de AliRoot est basé sur une libre modification de la méthode XP adaptée aux exigences de la Physique des Hautes Énergies. Comme nous l'avons déjà expliqué dans le paragraphe précédent, plusieurs développeurs peuvent déposer leur code dans le serveur svn de AliRoot. Dans le langage svn, la version de développement de AliRoot est appelée « tronc » (en anglais *trunk*) parce qu'elle constitue la « branche » principale du dépôt svn.

L'importance d'un processus flexible mais « prévisible » est capitale dans notre environnement du fait de deux facteurs d'instabilité spécifiques des grandes expériences de physique.

Le premier facteur est la décentralisation du développement. Pour le dire de façon simpliste, n'importe quel développeur habilité à le faire peut déposer dans le serveur svn de AliRoot n'importe quel morceau de code à n'importe quel moment. Malgré la présence de conventions de codage (d'ailleurs pas toujours respectées) et d'instructions générales sur la façon d'écrire le code, le Core Offline, responsable ultime du bon fonctionnement du système, n'a aucun contrôle, ni aucune idée par ailleurs, sur la qualité du code qui est développé, ni sur le moment auquel il sera introduit dans le dépôt svn.

La deuxième source d'instabilité est le taux de remplacement des développeurs. La durée de la présence d'un développeur dans l'expérience peut aller d'un stage d'été d'un mois à un contrat à durée indéterminée. Pour ne citer qu'un exemple, au début de l'année 2012 le groupe Core Offline au CERN était composé de 31 membres dont 7 avec un contrat à durée indéterminée, 4 avec un contrat de cinq ans, 5 avec un contrat de trois ans dont 4 étudiants de doctorat, 6 avec un contrat de deux ans et les autres avec des contrats plus courts.

Ces deux facteurs rendent nécessaire la prise de dispositions particulières pour s'assurer que, dans la mesure du possible, le code ait une certaine « uniformité »

¹⁶ En informatique, la réflexion est la possibilité pour un programme d'accéder pendant l'exécution à l'information sur les structures internes de haut niveau (objets ou structure des données). La réflexivité est la propriété d'un langage de programmation d'écrire de tels programmes. On dit alors que le langage est réflexif.

sémantique de façon à ce que le code d'un développeur puisse être compris et maintenu par un autre, collègue ou successeur.

Pour cette raison, en plus des règles syntaxiques du langage (qui dans le cas de AliRoot est, comme nous avons vu, le C++), le projet Offline a développé une série de conventions de codage (en anglais *Coding Conventions*) [72] qui doivent être respectées par les développeurs. La vérification de la conformité du code avec les conventions de codage est effectuée par un programme qui a été développé en collaboration avec l'institut de recherche Fondazione Bruno Kessler en Italie [73].

De plus le projet Offline a décidé d'interdire l'utilisation des méthodes de la librairie STL¹⁷, ainsi que des « templates »¹⁸ (en français *modèles*) pour les variables de type dérivés et des « C++ exceptions ». Cela a donné lieu à un débat très vif dans la communauté des développeurs, car une partie d'entre eux a questionné le droit du projet Offline d'« amputer » un langage de programmation d'une partie du standard. Cette décision était justifiée par le souci de réduire la complexité du code et d'éviter à des programmeurs non expérimentés de développer du code qui, même si raisonnable en soi, ne puisse produire des effets secondaires négatifs dans le contexte plus général de AliRoot. Après dix années de développement, nous pensons que cette décision, généralement bien respectée, n'a pas porté préjudice à la fonctionnalité de AliRoot, même si le débat « idéologique » continue aujourd'hui.

AliRoot builds											
Builds											
Start	Duration	Status	OS	Arch	Release	Prepare	Make	-install	-test	-cache	-autopkg
11.08.2011 07:00	2:03	OK	linux	x86_64	v4-20-Rev-38	5m 58s	20m 55s	2m 5s	1:29	3m 0s	13s
11.08.2011 07:00	3:08	OK	linux	x86_64	v4-20-Rev-38	5m 28s	1:03	2m 45s	1:53	3m 52s	3s
11.08.2011 07:00	3:36	OK	darwin	x86_64	v4-20-Rev-38	6m 6s	1:10	1m 49s	2:13	2m 36s	5s
11.08.2011 00:00	2:04	OK	linux	x86_64	v4-20-Rev-38	7m 14s	20m 50s	2m 8s	1:29	3m 12s	10s
11.08.2011 00:00	3:36	OK	darwin	x86_64	v4-20-Rev-38	6m 6s	1:10	1m 49s	2:13	2m 36s	5s
10.08.2011 07:00	2:05	OK	linux	x86_64	v4-20-Rev-38	8m 21s	21m 14s	3m 4s	1:27	3m 24s	11s
10.08.2011 07:00	2:56	OK	linux	i686	v4-20-Rev-38	6m 1s	1:01	1m 13s	1:42	4m 44s	6s
10.08.2011 07:00	3:36	OK	darwin	x86_64	v4-20-Rev-38	6m 6s	1:10	1m 49s	2:13	2m 36s	5s
10.08.2011 00:00	2:07	OK	linux	x86_64	v4-20-Rev-38	8m 1s	21m 19s	3m 6s	1:29	3m 30s	12s
10.08.2011 00:00	3:08	OK	linux	x86_64	v4-20-Rev-38	5m 28s	1:03	2m 45s	1:53	3m 52s	3s
10.08.2011 00:00	3:36	OK	darwin	x86_64	v4-20-Rev-38	6m 6s	1:10	1m 49s	2:13	2m 36s	5s
09.08.2011 19:48	3:03	OK	linux	x86_64	v4-20-Rev-38	7m 21s	1:19	2m 56s	1:29	3m 32s	14s
Show entire builds history											
AliRoot benchmarks											
Reconstruction PbPb											
Start	End	Duration	State	OS	Arch	Release	CPU time	Wall time	Total memory	RSS memory	File size
11.08.2011 08:06	11.08.2011 09:34	1:28	FAILED	Linux	unknown	v4-20-Rev-38					
11.08.2011 01:09	11.08.2011 02:37	1:28	FAILED	Linux	unknown	v4-20-Rev-38					
10.08.2011 08:12	10.08.2011 09:40	1:27	FAILED	Linux	unknown	v4-20-Rev-38					
10.08.2011 02:04	10.08.2011 03:33	1:28	FAILED	Linux	unknown	v4-20-Rev-38					
09.08.2011 22:01	09.08.2011 23:30	1:28	FAILED	Linux	unknown	v4-20-Rev-38					
09.08.2011 01:47	09.08.2011 03:13	1:25	FAILED	Linux	unknown	v4-21-32-AN					
08.08.2011 20:43	08.08.2011 22:13	1:29	FAILED	Linux	unknown	v4-20-Rev-36-1					
08.08.2011 01:40	08.08.2011 03:00	1:19	FAILED	Linux	unknown	trunk					
07.08.2011 08:39	07.08.2011 09:57	1:18	FAILED	Linux	unknown	trunk					
07.08.2011 01:39	07.08.2011 02:58	1:18	FAILED	Linux	unknown	trunk					
04.05.2007 01:22	04.05.2007 03:31	2:09	FAILED	Linux	x86_64	v4-05-13					
Show all history for this test											

Figure 40: Aperçu du système de compilation et vérification de AliRoot.

¹⁷ La Standard Template Library (STL) est une bibliothèque du C++ qui fournit des conteneurs et différentes opérations logico-mathématiques entre eux. Tout le code est contenu dans des fichiers header, et donc il n'y a pas de librairies à charger.

¹⁸ Les templates (en français modèles ou patrons) sont une façon d'exprimer la généricité en C++. Ils permettent d'écrire du code générique qui est automatiquement « spécialisé » en fonction des paramètres d'appel.

Chaque nuit, une version complète du code est compilée et plusieurs programmes de validation sont exécutés sur différentes plateformes. Les résultats sont automatiquement publiés sur une page web (voir Figure 40). Si la validation révèle des erreurs, elles sont examinées par l'équipe Core Offline du CERN et signalées aux responsables du code en erreur.

Chaque nuit, le code est également analysé par l'outil d'analyse de code Coverity [74] dont le CERN a acquis une licence. Le but de ce programme est de déceler des erreurs de programmation. Ces erreurs, ainsi que les possible erreurs et « warnings » (avertissements) de compilation, sont automatiquement signalées aux auteurs du code via des courriels. Parfois les auteurs du code ne sont pas ceux qui l'ont déposé dans le dépôt svn, et donc c'est à eux de contacter les auteurs pour obtenir les modifications nécessaires, car en introduisant du code dans AliRoot, ils deviennent responsables de sa maintenance. Chaque nuit, l'outil THtml de Root est exécuté sur AliRoot de façon à générer une documentation hypertextuelle du code avec une charte d'héritage des classes.

La vérification de la conformité du code avec les conventions de codage mentionnées plus haut sera bientôt également exécutée automatiquement chaque nuit. Tous les développeurs ne sont pas conscients de la nécessité de maintenir leur code en conformité avec les prescriptions du projet Offline. C'est pourquoi, selon les règles du Core Offline de ALICE, tout auteur laissant un avertissement sans réponse pendant huit jours consécutifs se verra privé de son droit d'écriture dans le dépôt svn de AliRoot, ses droits étant rétablis seulement après avoir fourni les corrections nécessaires.

D'autres vérifications sont effectuées de façon moins régulière, telles que la détection des « fuites de mémoire » (en anglais *memory leaks*) et des allocations de mémoire vive importantes dans le programme. L'automatisation de ces vérifications est à l'étude.

Toute cette procédure permet de garder la version « tronc » de AliRoot dans un état « opérationnel » en conformité avec le principe XP d'intégration et de vérification continues. Comme les vérifications sont faites chaque nuit, si la version « tronc » renvoie des erreurs, les changements responsables du problème sont ceux introduits pendant la journée, c'est-à-dire en moyenne une dizaine de changements erronés par jour. Il est donc relativement aisé d'identifier et de corriger la source d'un dysfonctionnement.

La différence avec les prescriptions XP est que nos vérifications ne sont pas unitaires (en anglais *unit tests*) mais « globales », car nous ne disposons pas des ressources nécessaires pour définir des vérifications spécifiques pour chaque unité du programme.

Bien que très régulièrement vérifiée, la version « tronc », continuellement en développement, ne peut pas être utilisée pour une production, car le taux de changement trop élevé peut créer des instabilités et des différences inacceptables dans les résultats. En effet, les vérifications faites pendant une nuit ne sont pas suffisantes pour valider complètement le code. Nous avons donc élaboré une procédure spécifique pour publier une version du code (en anglais *release*).

La date de publication est annoncée un mois à l'avance à la communauté des utilisateurs, ce qui leur permet de mettre dans le dépôt svn toutes les modifications qu'ils jugeront utiles. Deux semaines avant la date annoncée, une branche « candidate pour la publication » est créée dans le dépôt svn, nommée *vN-M-Release*, où N est l'identifiant numérique principal de la publication principale et M l'identifiant numérique secondaire. À partir de ce moment, seul le responsable de la librairie des programmes AliRoot (en anglais *program librarian*) a le droit d'écrire sur cette branche. Pendant les deux semaines précédant la date de publication, le code dans la branche svn est vérifié pour s'assurer de sa qualité et de sa stabilité. Si des problèmes sont relevés, ils sont corrigés à la fois dans la branche « candidate pour la publication » et dans le tronc. Le contraire est également possible, c'est-à-dire que le portage d'un problème corrigé dans le tronc soit demandé sur la branche candidate pour la publication. Après deux semaines, le jour de la publication, une « étiquette » (tag)¹⁹ est créée dans le dépôt du style *vN-M-Rev-L*. La logique est la même qu'auparavant, avec l'identificateur de révision L en plus. La distribution est ensuite « assemblée » (en anglais *built*) sur des machines virtuelles pour toutes les configurations de matériels et systèmes d'exploitation (serveurs d'assemblage, en anglais *build servers*) qui sont offertes aux utilisateurs de AliRoot, et les fichiers correspondants au format *tar* (en anglais *tar files*) sont préparés pour la distribution sur la Grille (voir Chapitre 5). Le processus d'assemblage des libraires binaires et des fichiers exécutables est géré par l'utilitaire CMake [75].

Pendant ce temps, le développement continue sur le tronc, qui donc diverge graduellement de la branche d'édition. En parallèle, l'utilisation de l'étiquette de la branche d'édition va révéler des bogues ou simplement des insuffisances qui doivent être corrigés. Les développeurs appliquent donc leurs corrections sur le tronc et ils émettent ensuite des « demandes de portage » de ces changements du tronc à la branche publiée. Ces demandes sont examinées chaque semaine par le groupe Core Offline du CERN pendant une visioconférence où les développeurs expliquent la raison et l'importance de la demande de portage, et chaque demande est soit acceptée soit refusée après discussion. La discussion est guidée par le souci de garder l'édition aussi stable que possible, seules les corrections des bogues sont acceptées et non les nouveaux développements. Une fois les demandes de portage acceptées et réalisées, une nouvelle étiquette est créée sur la branche de publication en incrémentant l'identificateur de révision. Cette nouvelle version du code est ensuite vérifiée et un nouvel « assemblage » est préparé et validé pour être distribué sur la Grille informatique.

Ce procédé permet d'avoir une mise à jour hebdomadaire du code de AliRoot. Cependant, il reste des modifications non portées du tronc à la branche de publication. Il s'agit de modifications incluant des développements nouveaux qui sont soit trop importants, soit sources potentielles d'instabilité. L'utilisation d'une version tronc sur la Grille, où les physiciens ont accès à d'importantes ressources de calcul et à la totalité des données, peut donc s'avérer intéressante. Pour satisfaire un tel besoin, une étiquette est créée chaque semaine sur le tronc,

¹⁹ On appelle *tag* (étiquette) une chaîne de caractères qui identifie un instantané du dépôt svn au moment où le tag est créé.

et le code correspondant est assemblé et distribué sur la Grille. On appelle cela une « étiquette d'analyse » (en anglais *analysis tag*) car elle est surtout utilisée pour les activités d'analyse des données qui très souvent nécessitent la toute dernière version du code.

Cette dynamique du code entraîne à partir d'un certain temps des différences entre le tronc et la branche de publication telles qu'il devient impossible de porter les corrections du tronc. Il faut donc préparer une nouvelle publication en partant du tronc, suivant le procédé décrit précédemment. Il y a entre quatre et six publications de AliRoot chaque année et une vingtaine d'étiquettes pour chaque branche de publication. Quand une nouvelle publication est déployée sur la Grille, l'ancienne devient obsolète. Dès lors les demandes de portage ne sont plus acceptées et son utilisation est « déconseillée ». L'édition reste quand même disponible sur la Grille pendant plusieurs mois, et le code et tout l'historique de l'édition sont disponibles dans le dépôt svn.

D'un point de vue pratique, le suivi des bogues et des demandes de portage est effectué à travers le système de suivi de bogue Savannah [76], développé au CERN sur la base du système GNU/Savane [77]. Nous avons en effet remarqué qu'un suivi par courriel n'est pas suffisant pour identifier et suivre clairement les différentes discussions. En ce moment, 151 bogues « ouverts » sont répertoriés dans Savannah, sur un total de 3156 depuis son introduction en 2005.

Le processus de développement de AliRoot est basé sur des microcycles d'une semaine, ce qui est en accord avec la philosophie XP. Une semaine est en effet le temps minimal pour un cycle complet, car les demandes de portage sont acceptées le lundi. Le mardi, le portage est effectué et le mercredi la nouvelle étiquette est vérifiée et assemblée. Le jeudi elle est disponible sur la Grille, ce qui laisse quatre jours en tout pour déceler les problèmes et demander de nouveaux portages. Évidemment c'est là un calendrier « idéal », car très souvent les demandes de portage sont incomplètes ou contiennent des erreurs, ou bien la nouvelle étiquette ne peut pas être vérifiée. Dans ce cas, le code doit être débogué avec l'aide des développeurs au prix de longues séances de programmation en couple (en anglais *pair programming*), souvent par téléphone ou visioconférence. Parfois la nouvelle étiquette peut être créée seulement le vendredi. Il est très important que chaque semaine il y ait une nouvelle étiquette. Si une demande de portage engendre des problèmes ou des conflits qui ne peuvent pas être résolus rapidement, elle est simplement repoussée à la semaine suivante. C'est en effet une des prescriptions les plus importantes de XP : en cas de difficulté, ne jamais repousser la date de publication du code, mais, si nécessaire, en réduire la portée en reportant les changements.

La procédure de développement que nous venons de décrire a été mise en place progressivement par l'auteur à partir du 2008, et elle a été opérationnelle pendant les premières années de prise de données. Comme on pouvait s'y attendre, nous avons eu un grand nombre d'urgences et de très fortes pressions pour « faire des exceptions ». On peut affirmer que Core Offline a répondu à toutes les demandes sans violer les principes de base du processus, qui a finalement montré suffisamment de souplesse pour s'adapter aux changements, souvent imprévus. La simplicité et l'extrême automatisation du système d'assemblage ont permis de répondre aux problèmes urgents en accélérant le

cycle sans l'altérer, et nous avons démontré la capacité à compléter un cycle de développement complet en moins d'une journée, y compris le déploiement sur la Grille, évidemment au prix d'une réduction de portée, car cela a été fait pour une seule modification critique, et d'une vérification très sommaire de la nouvelle étiquette.

Malgré son efficacité, ce système a présenté des faiblesses que nous nous efforcerons de corriger dans le futur.

Le processus de vérification a montré plusieurs fois son incapacité à déceler des problèmes qui apparaissent immédiatement dès que le code est utilisé en production sur la grille. Une vérification plus réaliste est nécessaire, ainsi qu'une meilleure automatisation du système de signalisation des problèmes, qui, s'ils ne se traduisent pas en un « plantage » (en anglais *crash*, condition d'erreur non recouvrable d'un programme), peuvent être noyés dans les messages émis par le programme.

L'intégration d'outils de diagnostic tels que valgrind [78] ou les outils de contrôle de performance de Google [79] dans le processus de vérification pourrait augmenter la fiabilité de chaque étiquette, mais seulement si des mécanismes automatiques de signalisation des problèmes sont mis en place, car l'inspection visuelle de la sortie diagnostique de ces utilitaires est trop lourde, surtout pour un programme de la taille de AliRoot.

Un autre problème sérieux du processus que nous venons de décrire est qu'il « emprisonne » le personnel du Core Offline en général et le responsable de la librairie des programmes en particulier dans une routine répétitive et assez stressante. La construction d'une stratégie suivant les grandes lignes de la méthode XP c'est avérée assez efficace, mais l'usure des hommes et des femmes « à la manœuvre » est perceptible et nous nous interrogeons sur l'issue dans le long terme.

Nous n'avons pas encore trouvé de solution à ce problème. Le Core Offline de ALICE est une équipe resserrée, hautement compétente et triée sur le volet. Y ajouter des ressources, même si c'était faisable du point de vue économique, est possible à la seule condition de trouver des professionnels de profil comparable. Le taux de remplacement a été jusqu'à maintenant tout juste suffisant pour équilibrer les départs « naturels » des personnels avec des contrats temporaires. À l'inverse, le travail est souvent assez « routinier » pour pouvoir le confier en grande partie à des techniciens plutôt qu'à des professionnels avec un doctorat en physique ou en science de l'information. Mais cela est vrai seulement pour 90 % des cas, quand tout va bien. En état d'urgence, toute la compétence d'un « physicien programmeur » chevronné est nécessaire pour comprendre promptement s'il s'agit d'un problème de physique ou du code, et pour le diagnostiquer et le résoudre rapidement. Cela est essentiel, car une accumulation trop importante de problèmes « signalés » vers la prochaine étiquette aurait pour effet de « casser » le processus, car le principe de microcycle fonctionne seulement si, en moyenne, le temps de résolution des problèmes liés à une nouvelle étiquette est d'au plus une semaine.

4 La structure du code AliRoot

Dans le chapitre précédent, nous avons discuté le processus de développement et de déploiement du code de AliRoot. Dans ce chapitre, nous décrivons la structure du code de AliRoot ainsi que ses fonctionnalités les plus importantes. Nous commençons par un bref historique du projet, qui aidera à comprendre le pourquoi des décisions qui ont conduit à la structure et aux fonctionnalités actuelles du code.

4.1 Histoire de AliRoot

Depuis l'installation du premier ordinateur Ferranti-Mercury en 1958, le CERN a développé des bibliothèques de logiciels pour le traitement des données de ses expériences. Ces programmes ont toujours été distribués librement aux physiciens travaillant au CERN, qui ont trouvé naturel de les utiliser dans le cadre de leurs expériences également en dehors du CERN. Il devint tout à fait naturellement d'introduire dans ces bibliothèques des développements réalisés en-dehors du CERN. Ainsi, il fut possible de créer un ensemble cohérent et complet capable de fournir tous les outils informatiques nécessaires au traitement des données. Comme nous avons déjà dit, cela préfigurait l'esprit « Open Source ».

C'est ainsi que vit le jour la Bibliothèque des Programmes du CERN (en anglais *CERN Program Library*) ou CERNLIB. Écrite principalement en langage FORTRAN, elle fut développée et distribuée activement jusqu'à la fin des années 1980, et devint « de facto » le standard pour la Physique des Hautes Énergies. Même si d'autres grands laboratoires (SLAC, FNAL ou DESY) développèrent leurs bibliothèques de programmes, la bibliothèque CERNLIB était considérée comme un élément indispensable pour n'importe quelle expérience de physique des particules et nucléaire.

Au début des années 1990, avec en perspective l'avènement du LHC, la question du renouvellement de CERNLIB se posa. Bien que mondialement utilisée et appréciée, le devenir de CERNLIB fut mis en question par les nouvelles technologies de la science de l'information, en particulier la programmation Orientée Objet (en anglais *Object Oriented*, OO) [80], laissant la communauté devant le dilemme entre évolution et révolution. La discussion parmi les physiciens programmeurs fut très animée, avec d'un côté les tenants d'une évolution s'appuyant sur le nouveau standard FORTRAN (FORTRAN 90) incluant le FORTRAN utilisé pour CERNLIB (FORTRAN 77). D'un autre côté se trouvaient les tenants d'une voie plus radicale, à savoir le développement *ab initio* d'une nouvelle bibliothèque CERNLIB rédigée en langage orienté objet tel que C++, Java ou Eiffel. D'autres encore s'interrogeaient sur la possibilité de trouver des produits commerciaux incluant une partie des fonctionnalités de CERNLIB afin de limiter le développement en interne.

Pour trouver des solutions rationnelles, la communauté des physiciens travaillant aux expériences CERN, et en particulier ceux qui préparaient les expériences du LHC, se réunirent en un comité pour le calcul au LHC, LCB (en anglais *LHC Computing Board*). Dans les années 1993-1994 furent ainsi lancés une série de projets de recherche et de développement, dans le domaine de

l'informatique, semblables à ceux qui avaient été lancés pour le développement des nouvelles technologies de détection de particules. Le but était d'examiner les technologies de programmation orientée objet et l'utilisation de programmes commerciaux dans la Physique des Hautes Énergies. Entre-temps, l'option FORTRAN 90 avait été majoritairement rejetée par la communauté des physiciens.

L'histoire de ces projets est longue et complexe et aujourd'hui encore pas « consensuelle ». Elle est relatée en détail dans un livre édité par l'auteur [81]. En ce qui nous concerne, il suffit de dire qu'à la fin des années 1990, aucun de ces projets n'avait produit un résultat fiable et univoque. Les expériences LHC durent donc se « contenter » de l'ancienne librairie CERNLIB pour leur préparation, en particulier pour les simulations visant à évaluer la conception et la performance des détecteurs. Le développement de CERNLIB, avait auparavant été « gelé » !

En parallèle à cette R&D, une petite équipe au sein de l'expérience NA49 [82], dirigée par René Brun, un des auteurs de CERNLIB, avait développé une librairie de programmes orientés objet en C++ appelée Root [83]. Cette librairie, spécialement conçue pour les besoins des expériences de physique de nouvelle génération (FermiLab Run II, RHIC et LHC), avait été officiellement présentée dans un séminaire au CERN en 1995. Elle souleva un énorme intérêt auprès des physiciens. À ce moment-là Root incluait déjà toutes les fonctionnalités de l'ancienne CERNLIB et avait dépassé beaucoup de ses limitations. Le management du CERN, préoccupé par un détournement potentiel des ressources des projets « officiels », désormais ouvertement en crise, réagit en interdisant formellement aux expériences LHC d'utiliser Root. En dépit de cela, en 1998, à la suite d'une intense campagne d'essais, Root fut adoptée officiellement par les expériences de FermiLab Run II et de RHIC aux États-Unis.

C'est à ce moment que l'auteur est nommé coordinateur du calcul de l'expérience ALICE, avec pour tâche de bâtir l'environnement de simulation, reconstruction, calibrage et analyse et l'infrastructure de production. Le choix était entre bâtir un environnement en FORTRAN en attendant les résultats des projets « officiels » lancés par LCB, qui apparaissaient de moins en moins viables, mais qui jouissaient du soutien « officiel » du CERN, ou « prendre ce qui marche », c'est-à-dire Root et GEANT 3, pour la simulation de la réponse du détecteur, et construire un environnement orienté objet *ab initio*. Les autres expériences LHC avaient opté pour la première des deux solutions.

ALICE devait d'urgence commencer des études poussées sur la conception des détecteurs, pour pouvoir lancer leur construction. Il fallait donc choisir vite. Différents éléments rendaient le choix difficile. Bien que la confiance des physiciens dans les projets « officiels » fût ébranlée, sa renommée étant engagée, il était probable que le CERN aurait « fait ce qu'il fallait » pour rendre ces projets officiels viables. En attendant, chacune des autres expériences avait formé deux équipes informatiques, l'une développant les programmes en FORTRAN avec CERNLIB, et l'autre, majoritairement composée d'informaticiens, suivant l'évolution des projets officiels. ALICE n'avait pas les ressources humaines suffisantes pour faire de même. Cependant, en adoptant Root, ALICE se mettait en position de braver une décision officielle de la direction du CERN, avec des

conséquences potentiellement non négligeables. Malgré l'expérience et les essais convaincants effectués par les expériences outre-Atlantique, malgré la réputation de René Brun qui constituait une garantie en soi, la réaction de la communauté des utilisateurs, qui ne comptait que très peu de programmeurs C++, à une transition rapide vers Root et C++ était imprévisible. Au problème créé par le choix de Root s'ajoutait celui du choix de GEANT 3 codé en FORTRAN. En effet, l'un des projets « officiels » concernait la réalisation de GEANT 4, un programme de simulation de détecteur écrit en C++ et orienté objet, successeur de GEANT 3, mais qui était encore loin d'être opérationnel.

A la suite d'une brève mais vive discussion à l'intérieur de la collaboration, la décision fut prise de baser le développement sur Root et GEANT 3, et de l'appeler AliRoot (logiciel de ALICE basé sur Root). Le problème de la compatibilité entre Root en C++ et GEANT 3 en FORTRAN fut résolu en développant une série d'interfaces qui faisaient apparaître GEANT 3 comme une classe C++ et qui permettaient donc aux développeurs de travailler entièrement en C++.

Le manque de temps et le peu de ressources humaines disponibles ont été à l'origine de deux autres choix qui caractérisent le projet Offline de ALICE. L'équipe Core Offline comprenait aussi bien les développeurs du code que les physiciens qui l'utilisaient pour les études de conception du détecteur, les deux rôles se confondant souvent dans une même personne. Cela s'est révélé un choix très positif, bien que dicté par des considérations purement pratiques, car il a été à l'origine d'une coopération très étroite entre « programmeurs » et « physiciens ». Cette excellente communication a rendu possible l'expérimentation de cycles de développement très courts qui ont été à l'origine du modèle de développement actuel de AliRoot. Le deuxième choix était d'utiliser Root directement, sans une « couche d'isolation » ce qui aurait permis, si nécessaire, de remplacer Root par une autre librairie de classes. C'était un risque technologique majeur en cas de débâcle, politique ou technique, du projet Root.

La plupart des projets « officiels » ont été arrêtés sans produire aucun résultat, avec la notable exception de GEANT 4. Toutes les expériences LHC ont ainsi fini par adopter Root comme base de leurs logiciels. À la différence de ALICE, ils ont développé des « couches d'isolation » plus ou moins sophistiquées, qu'avec la confirmation de Root comme unique choix ils doivent maintenant simplifier, non sans peine.

La simplicité de la structure initiale de AliRoot et le support fourni par le Core Offline ont permis à une grande partie de la communauté des physiciens de ALICE de faire une transition du FORTRAN au C++ sans trop de peine. Le fait que Root soit sémantiquement et logiquement très semblable à l'ancienne CERNLIB a été d'une grande aide. Cela étant dit, nous pensons que le passage de FORTRAN à C++ n'a pas eu que des effets positifs. Les discussions sur les langages de programmation prennent parfois des tons « doctrinaires » et nous n'avons pas ici l'intention d'ajouter à un débat qui a fait couler beaucoup d'encre et de « bits », mais seulement de relater quelques faits que nous avons observés.

La plupart des physiciens « seniors » n'ont pas opéré la transition vers le langage C++. Il en a résulté une rupture de communication avec les nouvelles générations de physiciens. Un langage de programmation est un moyen de communication

non seulement entre homme et machine, mais aussi entre collègues. Avant le passage à C++, un jeune étudiant pouvait en tout cas montrer à son superviseur un algorithme sous la forme du code de programmation (en FORTRAN) utilisé pour l'implémenter. Maintenant cela n'est pas toujours possible. Nous avons perdu des ressources et créé une coupure générationnelle. On pourrait objecter que la situation est temporaire, car une fois toutes les générations de physiciens familières avec le nouveau langage, la communication sera rétablie. Ça serait sans tenir compte de la longévité exceptionnelle du FORTRAN comme langage de programmation et de l'apparition d'autres nouveaux langages dans les logiciels des expériences LHC tels que ruby, python, perl et Java. En ce sens AliRoot est plutôt une exception car il est écrit à plus de 99 % en C++. Cette brisure générationnelle est donc appelée à se répéter dans le futur, à moins qu'un langage d'une longévité telle que celle du FORTRAN n'apparaisse.

Le concept de programmation orientée objet est élégant et, pourrait-on dire, « logiquement nécessaire ». La preuve de son efficacité n'était plus à faire. Le C est un langage puissant qui permet d'accéder directement à l'architecture de la machine (adressage direct de la mémoire, accès aux structures de données et aux services du système d'exploitation, manipulation de bits etc.), au point qu'il peut être considéré comme un « assembleur de haut niveau » : rien (ou presque) n'est impossible en C. Les compilateurs C sont parmi les plus solides et les plus efficaces, et, de plus, ils font du C un langage extrêmement portable. La versatilité et la puissance du C ont comme corollaire qu'il est possible d'écrire du code « obscurci » (en anglais *obfuscated*) très difficile à déchiffrer.

La décision de baser un langage orienté objet sur C fut donc naturelle et pragmatique. Celle d'inclure dans ce langage la totalité ou presque du C le fut aussi. L'adoption dans le standard C++ des *templates* était raisonnable à la vue de leur versatilité, mais au prix d'une deuxième sémantique pour exprimer le polymorphisme qui, d'un certain point de vue, est orthogonal à l'héritage des classes. Le résultat est un langage qui hérite des avantages du C, mais qui est extrêmement complexe et qui permet de mélanger une programmation orientée objet assez élégante avec les « ruses » du C, dans un code qui peut être obscurci à souhait, le tout dans une polysémie totale. Le standard du FORTRAN peut être contenu dans une centaine de pages, la dernière version du standard C++ s'étend sur plus de 1.300 pages.

De plus, le C++ n'introduit pas de mécanismes globaux de gestion de la mémoire, tels que le éboueur de mémoire (en anglais *garbage collector*) de Java. La justification donnée par l'auteur du C++, Bjarne Stroustrup, est que cela aurait eu pour effet d'alourdir l'exécution de tous les programmes en C++, alors qu'un éboueur de mémoire répondant aux exigences du problème en question peut facilement (sic !) être développé au cas par cas. La même remarque pourrait être faite à propos du choix de ne pas imposer de hiérarchie unique de classes, et même de ne pas imposer l'utilisation des classes.

Si maintenant nous imaginons ce langage de programmation utilisé dans le contexte d'une expérience de Physique des Hautes Énergies, avec environ 200 développeurs aux compétences en informatique très variables, on peut facilement imaginer le besoin d'une discipline stricte pour développer le projet. Malheureusement, C++ était à l'époque de son introduction un langage encore

assez nouveau, publié seulement en 1985 par Bell's Laboratory [84], et personne dans les projets Offline des collaborations LHC n'avait ni l'expérience ni les connaissances pour pouvoir élaborer les directives générales pour développer un système de taille importante en C++. L'utilisation de Root a énormément aidé à limiter erreurs et errements, mais pour le reste nous avons appris tout en construisant le système. Et cela avec un langage où n'importe quel utilisateur peut « détourner » les intentions à tout moment et d'un nombre infini de façons, détournement que seule une inspection « visuelle » du code peut déceler. Les conséquences de ces détournements sont potentiellement catastrophiques par exemple sur la taille de mémoire et la performance du programme, sinon sur les résultats eux-mêmes.

De plus, pour ne rien arranger, l'introduction du C++ dans la Physique des Hautes Énergies a été effectuée en mettant l'accent sur la « liberté » que ce langage apporte aux utilisateurs dans leur travail, alors que seule une grande cohérence sémantique peut permettre de développer et entretenir des systèmes de grande taille. Je crois que l'adoption précoce de règles de codage et la mise en place avec AliRoot d'un cadre assez contraignant, avec même l'interdiction d'utiliser certaines parties du standard C++, allait dans la bonne direction. Mais le fait reste que nous avons procédé souvent par tâtonnement et cela a laissé dans AliRoot beaucoup de « scories » qui alourdissent le code et qui ne sont pas faciles à éliminer.

En effet, nous avons été avertis de tout cela quand la collaboration BaBar à SLAC [85] décida la première de passer à C++ en 1995. Dans un discours très remarqué [86] à la conférence Computing in High Energy Physics 1997 [87] à Berlin, le coordinateur du calcul de BaBar, Bob Jacobsen, avait donné une vision très claire des problèmes, en avouant que la seule façon qu'il avait trouvée de faire marcher les choses était de réécrire la nuit ce que les physiciens avaient développé pendant le jour. Clairement une stratégie qui présentait un problème d'échelle avec l'agrandissement des collaborations et du nombre des développeurs. Ces avertissements ont été largement ignorés car contraires à la doxa répandue par les tenants de C++ à l'intérieur des collaborations LHC, et ils ont été « redécouverts » non sans peine dans les années suivantes.

Concluons cette digression sur le C++ avec une remarque sur la performance des programmes. La « distance » entre une opération entre objets et le code machine qui l'implémente est telle qu'on obtient des différences de performances extrêmes et surprenantes avec différentes expressions sémantiquement équivalentes. On peut en comprendre la raison seulement en examinant le langage machine généré par le compilateur, ou en utilisant des outils d'analyse de performance qui sont spécifiques de l'architecture matérielle utilisée, ce qui demande beaucoup de temps et de compétences qui ne sont pas toujours disponibles pendant le processus de développement. Ici aussi nous sommes confrontés à un processus d'apprentissage long et difficile, et l'apparition continue de nouvelles architectures avec de plus en plus de parallélisme ajoute à la complexité de cet exercice.

Pour conclure c'est l'opinion de l'auteur que probablement le passage au C++ était inévitable, pour différentes raisons techniques et historiques, mais qu'il a été beaucoup plus difficile et complexe qu'annoncé pas ses apôtres, qui eux-

mêmes manquaient de l'expérience et de la connaissance nécessaires. Nous avons beaucoup appris, même si la route pour une compréhension satisfaisante de tous les éléments mentionnés ci-dessus est probablement encore longue. Une meilleure collaboration entre expériences et entre physiciens programmeurs et informaticiens sera probablement nécessaire pour progresser plus vite et pour mieux se préparer aux prochains changements qui s'annoncent dans le matériel et les logiciels.

4.2 Le logiciel Root

Avant de décrire en détail la structure de AliRoot, il convient de décrire brièvement le logiciel Root puisqu'il constitue la colonne vertébrale de AliRoot. Plutôt qu'une description technique détaillée de ce logiciel, très bien documenté par ailleurs (voir le site de Root [88]), nous en discuterons les principes de base.

Une des limitations les plus sérieuses de l'ancienne librairie de programmes CERNLIB était la difficulté d'en étendre les fonctionnalités de façon uniforme et cohérente, en particulier en ce qui concernait l'interface utilisateur des programmes exécutables tels PAW, sans passer par des recompilations massives ou la régénération des dictionnaires d'interface utilisateur.

Afin de proposer un système cohérent et extensible, Root a choisi une hiérarchie de classes avec une seule racine (TObject) à la manière de Java. Cela permet la définition d'une série d'interfaces virtuelles et de conteneurs polymorphes et aussi la mise en place de dictionnaires dynamiques d'objets. L'utilisateur peut définir ses propres classes dérivant de TObject et qui peuvent alors utiliser les services de Root, ou encore il peut définir des services qui étendent les fonctionnalités de Root grâce à des interfaces virtuelles.

Le système est physiquement construit comme une série de bibliothèques partagées (en anglais *shared libraries*, voir Figure 41) auxquelles l'utilisateur peut ajouter les siennes qui peuvent être chargées dynamiquement par l'application Root sans qu'une recompilation soit nécessaire.

Le langage d'interaction et d'écriture des procédures interactives est le C++ lui-même. Cela a demandé le développement d'un interprète (appelé cint), c'est-à-dire d'un programme qui est capable de lire du C++ ligne par ligne et de l'exécuter immédiatement sans passer par la compilation et la génération d'un fichier exécutable, mais en créant « au passage » une représentation compacte du code en mémoire appelé *bitecode*.

Ce principe offre un système puissant et flexible, mais un problème très difficile à résoudre demeurerait. Dans le langage C++, face à l'extrême complexité que peuvent prendre les structures des données et leurs relations, les opérations d'entrée et sortie restent relativement simples, se bornant à la possibilité d'écrire et lire des « flots » de bits. Cependant les relations entre structures de données exprimées par des pointeurs sont perdues lorsque ces structures sont écrites sur disque. De plus, n'importe quel petit changement de la classe risque de rendre inutilisables les fichiers déjà écrits.

Pour pallier ces problèmes, Root fournit un utilitaire basé sur l'interprète interactif qui lit et décode les fichiers d'entête « headers » et pour chaque classe, écrit des méthodes auxiliaires pour l'introspection et l'entrée et sortie. Cela

permet d'implémenter un « dictionnaire », c'est-à-dire une description complète des classes et des méthodes, qui peut être consulté par le programme lui-même pendant son exécution. Grâce à ce dictionnaire, les informations sur les données écrites (métadonnées, en anglais *metadata*) peuvent être insérées dans les fichiers au prix d'une taille additionnelle réduite. Cela permet à un programme de relire des anciennes données même si la classe a changé (*évolué*). On parle dans ce cas d'« évolution du schéma » (en anglais *schema evolution*). Grâce à cette fonctionnalité, il est aussi possible de maintenir les relations de référence via pointeurs entre les classes qui sont traitées dans la même opération d'écriture.

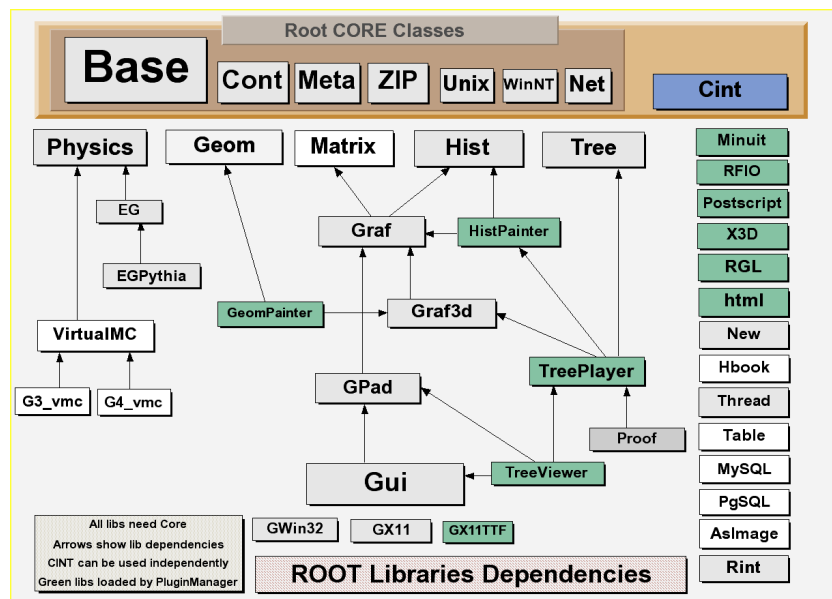


Figure 41 : Aperçu des bibliothèques de Root.

De plus, grâce à ce dictionnaire, le langage interactif a accès à toutes les classes de Root sans compilation, et on peut même en définir interactivement de nouvelles et les utiliser avec les classes compilées. Tout cela offre une grande souplesse et simplicité d'utilisation. Nous avons remarqué que la possibilité d'exécuter interactivement des lignes de C++ a été d'une grande aide pour beaucoup d'utilisateurs lors de leur apprentissage du C++. Cette philosophie a aussi introduit un modèle de développement propre à Root. Le code est développé interactivement en éditant une *macro*, c'est-à-dire un fichier source C++ qui est exécuté interactivement depuis Root. L'interprète est assez robuste pour interpréter plusieurs milliers de lignes de code et il comprend la plupart des instructions du standard C++. De plus, il est possible, toujours à partir du programme Root, de demander que la macro soit compilée avec le compilateur du système et chargée en tant que bibliothèque partagée pour être exécutée. Cela permet de créer et tester très rapidement du code qui, une fois vérifié, peut être transformé en une bibliothèque partagée et être introduite dans le système.

L'application de ce même procédé aux classes définies par l'utilisateur permet d'étendre toutes ces fonctionnalités sans aucune distinction entre les classes « natives » et celles développées par les utilisateurs.

L'interprète cint a été l'un des atouts les plus importants de Root, mais son développement et sa maintenance ont demandé beaucoup de ressources. On peut dire qu'il a été victime de son succès, car, né afin d'interpréter des macros des quelques centaines de lignes pour éviter les tâches répétitives et pour implémenter de logiques simples, il a été de plus en plus utilisé pour exécuter de vrais programmes, parfois des milliers de lignes avec toutes les possibilités du C++. Cela a demandé des extensions continues du prototype initial. C'est seulement depuis deux ans qu'une alternative semble se présenter avec le compilateur LLVM [89] développé par Apple et disponible en code source libre. Le remplacement de cint par LLVM est prévu pour cette année.

Root est donc à la fois un programme d'analyse de données et une infrastructure (en anglais *framework*) qui permet de développer des applications complexes en s'appuyant sur un ensemble de services et d'algorithmes.

	Lignes de code					Années- personne	Coût estimé (M€)
	C++	C	Sh	Total	%		
Écrit	1.455.708	411.191	68.233	1.976.036	66,3	577,58	78,0
Généré	154.706	753.853	2.063	918.365	33,7	231,38	21,9

Table 6 : Composition de la source de Root.

Le langage de programmation C++ offre une grande liberté de choix dans le style de programmation, aussi bien au niveau syntaxique que sémantique. L'utilisation d'une infrastructure prédéfinie restreint considérablement cette liberté. Dans le cas de Root par exemple, le fait d'hériter de l'objet racine TObject est une contrainte et ajoute 16 octets à chaque classe, mais donne accès à toute une série de services pour l'introspection, l'entrée et sortie et la possibilité d'utiliser les conteneurs polymorphes de Root. Tout au long de l'évolution de Root, les avantages et les désavantages d'une infrastructure plus ou moins contraignante, comparés à une ou plusieurs librairies de classes indépendantes, ont été discutés. Une caractéristique assez intéressante de ce débat est le fait que la majorité des utilisateurs et des « théoriciens » est en faveur d'une infrastructure aussi légère et peu contraignante que possible, voire d'une simple collection de librairies de classes, et qu'en même temps ces mêmes utilisateurs ont toujours tendance à demander plus de services et plus d'« intégration ».

Malgré cette contradiction apparente, nous pensons qu'une telle attitude des utilisateurs est très utile aux développeurs, car elle les encourage à éviter toutes contraintes inutiles et à limiter au minimum la pénalité de l'infrastructure en termes de performance, taille de mémoire et contraintes d'utilisation.

Pour ce qui nous concerne, nous pensons que la constitution d'un système intégré est nécessaire dans la construction d'un programme de la dimension de AliRoot. La question est simplement de savoir qui va définir et implémenter l'infrastructure. Si nous n'avons à notre disposition que des librairies de méthodes, l'expérience est en charge de l'infrastructure. Si, en revanche, nous choisissons un système tel que Root qui offre, voire impose un cadre assez précis, l'expérience peut s'affranchir de ce travail. Dans le cas de Root, l'avantage est encore plus grand dans la mesure où Root est capable de générer une bonne

partie des méthodes pour pouvoir utiliser les services offerts par l'infrastructure. Nous avons vu dans la Table 5 la taille du code généré, proche de celle du code effectivement écrit. De plus, avec Root, nous disposons d'une infrastructure qui évolue grâce aux contributions d'une communauté considérablement supérieure en nombre à celle d'une seule expérience.

Comme nous pouvons le voir dans la Table 6, du point de vue quantitatif Root est composé de presque deux millions de lignes de code, en grande partie en C++.

4.3 Structure de AliRoot

La philosophie du développement de AliRoot a été de fournir une infrastructure informatique à laquelle les différents sous-détecteurs ajoutent leur code, en limitant leur contribution à la partie « algorithmique » spécifique de chaque détecteur. L'idée est de minimiser la quantité de code que chaque détecteur doit développer et de lui permettre de faire appel à des services communs pour toutes les opérations d'accès aux données, de graphique, de description de la géométrie du détecteur, etc. Comme nous le verrons, cela a été réalisé par l'introduction de classes d'interface virtuelles pour pouvoir faire abstraction de la fonctionnalité de l'implémentation et permettre le maximum de standardisation. Cette approche, qui a été rendue facile grâce à l'utilisation d'un langage orienté objet, présente des avantages en termes de standardisation et de clarté du code. De plus, chaque nouvelle fonctionnalité introduite dans l'infrastructure est facilement utilisable par tous les détecteurs grâce à l'emploi d'interfaces virtuelles communes.

La fonctionnalité de l'infrastructure AliRoot est schématisée dans la Figure 42. Les données simulées sont générées par les « générateurs », c'est-à-dire des programmes MonteCarlo qui reproduisent l'état final de la collision, qui devient l'état initial du transport dans le détecteur. Dans cette phase, les particules ou « traces » générées sont transportées à travers le détecteur par l'intermédiaire de logiciels de simulation de détecteurs tels que GEANT 3 [52], FLUKA [90] et GEANT 4 [56], dont nous parlerons dans la Section 4.5. Ces programmes calculent dans le détail le dépôt d'énergie des particules dans le détecteur, qui est généralement appelé un « hit » (anglais pour *coup*), selon la terminologie héritée de GEANT 3.

Les résultats sont ensuite transformés en une réponse *idéale* du détecteur (signaux bruts), qui est ensuite transformée en réponse réelle du détecteur, en tenant compte de la mise en forme électronique des signaux jusqu'à la numérisation (signaux numérisés, en anglais *digits*). La nécessité de « superposer » différents événements simulés nous a amenés à développer un format de sortie intermédiaire consistant en des signaux numérisés sommables. Ce sont des signaux à haute résolution, avec le seuil à zéro, qui peuvent être additionnés quand plusieurs événements simulés sont superposés. Les signaux numérisés sont ensuite transformés dans le format qui sera produit par l'électronique des détecteurs (RAW ou données brutes). À partir de là, le traitement des données réelles est identique au traitement des données simulées.

Les données produites par les générateurs d'événement contiennent des informations complètes sur les particules générées : l'identité des particules (PID) et leur impulsion. Comme ces événements sont traités par la chaîne de simulation, l'information est dégradée et réduite en un format identique à celui utilisé pour l'information générée par les particules lorsqu'elles traversent le détecteur. Les algorithmes de reconstruction ont pour but de reconstruire la trajectoire et de déterminer l'identité des particules à partir des informations contenues dans les données brutes.

Afin d'évaluer le logiciel de reconstruction et les performances du détecteur, les événements simulés sont traités par l'ensemble du cycle et ensuite l'information reconstruite est comparée avec les informations prises directement à partir de l'événement généré par le MonteCarlo.

Les simulations rapides introduisent des raccourcis dans la chaîne, comme indiqué par les flèches sur la figure. Leur but est d'augmenter la vitesse de la simulation au détriment des détails et elles sont utilisées pour des études spécifiques. L'infrastructure AliRoot contient plusieurs algorithmes de simulation rapide.

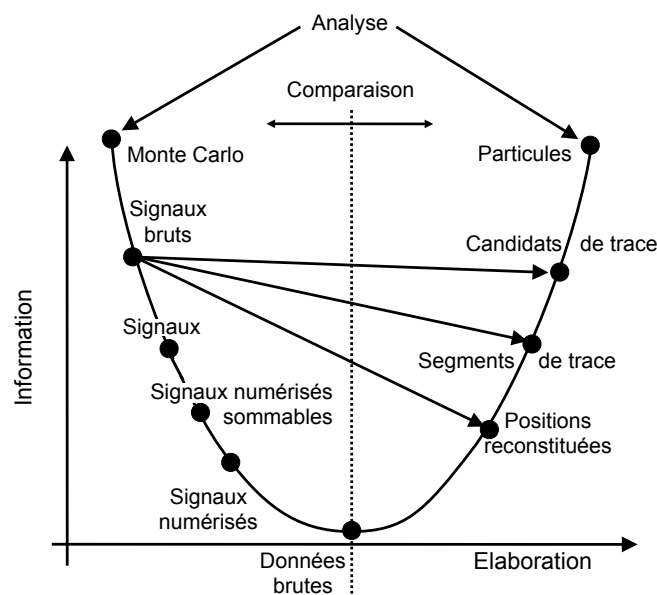


Figure 42 : Représentation fonctionnelle de AliRoot.

L'utilisateur peut intervenir dans ce cycle à condition de respecter les interfaces exposées par AliRoot. Les interfaces d'entrée et sortie et les interfaces utilisateurs font partie du système, ainsi que les outils de visualisation et d'analyse des données et toutes les procédures et les services d'intérêt général. Le champ d'application de l'infrastructure évolue avec le temps suivant les besoins des utilisateurs.

4.4 Principes de conception de AliRoot

Les principes de base qui nous ont guidés dans la conception de l'infrastructure AliRoot sont la réutilisabilité et la modularité, avec l'objectif de minimiser la

quantité de code d'utilisateur à écrire ou réécrire et de maximiser la participation des physiciens dans le développement du code.

La réutilisabilité est la protection de l'investissement réalisé par les physiciens programmeurs de ALICE. Le code contient une grande quantité de connaissances scientifiques et il est donc une ressource précieuse. Nous préservons cet investissement par la conception d'un système modulaire dans le sens ci-dessus et en faisant en sorte de maintenir le montant maximum de compatibilité ascendante, tout en évoluant dans notre système.

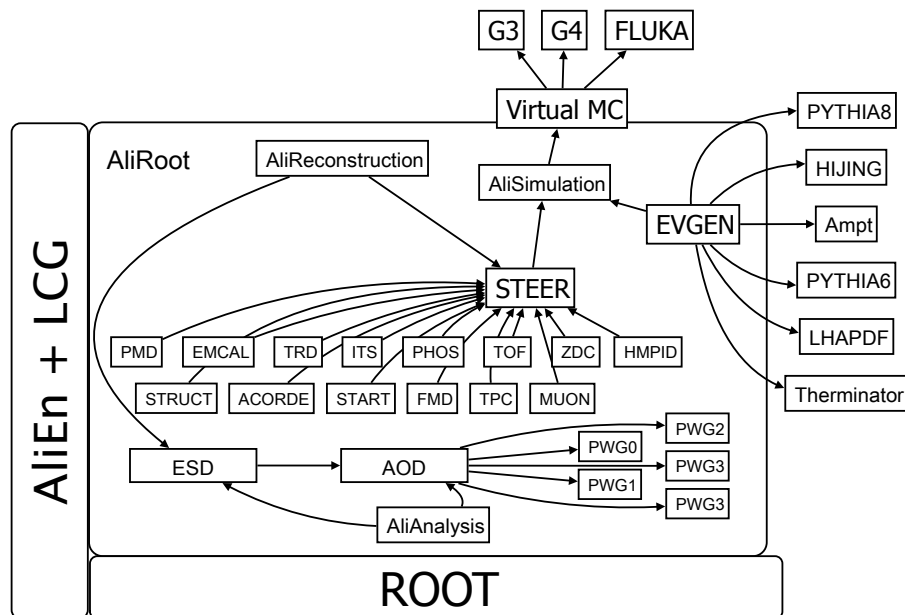


Figure 43 : La structure de AliRoot.

La modularité permet le remplacement des parties du système avec un impact minimal ou nul sur le reste. Toutes les parties du système ne sont pas destinées à être remplacées et la modularité est ciblée sur les éléments susceptibles d'être modifiés. Il y a des éléments que nous n'avons pas l'intention de changer, mais plutôt de développer en collaboration avec leurs auteurs. Chaque fois qu'un élément doit être modulaire dans le sens ci-dessus, nous définissons une interface abstraite. Quelques exemples :

- Différents programmes MonteCarlo peuvent être utilisés pour le transport sans avoir besoin de changer la génération des événements ou la description de la géométrie des différents sous-détecteurs. Cela est réalisé comme décrit plus en détail dans la section 4.5 par l'intermédiaire d'un ensemble d'interfaces virtuelles qui font partie du système Root, appelées Virtual MonteCarlo.
- Des générateurs d'événements différents peuvent être utilisés et même combinés via une seule interface abstraite.
- Différents algorithmes de reconstruction pour chaque sous-détecteur peuvent être utilisés sans changements dans le reste du code. Cela inclut

également les algorithmes du déclencheur de haut niveau, afin de permettre leur comparaison avec ceux hors-ligne. Encore une fois, cette possibilité est mise en œuvre via des interfaces abstraites.

- Les codes des différents détecteurs sont indépendants, de sorte que les groupes de détecteurs différents peuvent travailler simultanément sur le système tout en minimisant les interférences.

La structure générale de AliRoot est représentée dans son ensemble dans la Figure 43. Dans la suite, nous décrivons dans les grandes lignes les différents composants. AliRoot s'appuie sur Root pour les services de base, et sur l'infrastructure distribuée de la Grille informatique de WLCG et le logiciel de Grille AliEn pour l'infrastructure de production. Nous décrivons la Grille et l'infrastructure de production de ALICE dans le Chapitre 5.

L'infrastructure de l'Offline de ALICE a été conçue et réalisée comme un ensemble modulaire de bibliothèques en C++. La Figure 43 donne un aperçu de la structure du code où une large utilisation est faite des classes abstraites qui définissent l'architecture et la fonctionnalité du code des différents composants.

Les bibliothèques centrales (groupe STEER) contiennent les classes de base (prototypes) pour la simulation, le calibrage, la reconstruction et l'analyse. Ces bibliothèques offrent plusieurs fonctions telles que :

- Contrôle de l'exécution du programme pour la simulation, le calibrage, la reconstruction et l'analyse.
- Gestion de l'exécution et de l'entrée et sortie. Création et destruction des structures de données, initialisation et terminaison des phases du programme, gestion des fichiers.

Les sous-détecteurs sont des modules indépendants qui contiennent du code spécifique pour la simulation et la reconstruction. La simulation de la réponse du détecteur peut être effectuée à l'aide de différents codes de transport via le mécanisme du Virtual MonteCarlo (voir Section 4.5).

La même technique est utilisée pour accéder à des générateurs d'événements différents. Les générateurs d'événement sont accessibles via une interface virtuelle qui permet le chargement de différents générateurs au moment de l'exécution. La plupart des générateurs sont écrits en FORTRAN, mais la combinaison de bibliothèques à chargement dynamique et des classes « d'enveloppe » en C++ rend cela complètement transparent pour les utilisateurs.

Chaque sous-détecteur implémente une spécialisation de ces classes de base pour les détecteurs selon ses propres caractéristiques. Cela a permis une grande flexibilité et uniformité des fonctions de base. Il a été par exemple très facile d'introduire de nouveaux sous-détecteurs à l'occasion des études pour l'amélioration du détecteur prévue pendant les prochains arrêts de l'accélérateur (2013-2014 et 2017-2018), sans modifier la structure du code.

Nous donnerons dans la suite une brève description des fonctionnalités de simulation, calibrage, reconstruction, analyse et visualisation.

4.5 La simulation

Dans le système AliRoot, les classes de simulation contiennent le code pour la description géométrique de AliRoot et pour la génération des signaux émis par le détecteur au passage des particules. L'entrée du programme de simulation est constituée des fichiers de calibrage, des conditions générales du run qui doivent être simulées et des caractéristiques des événements à simuler selon les conventions du générateur d'événements choisi. La sortie d'une simulation contient les signaux émis par le détecteur sous la forme de « valeurs numérisées » (en anglais *digits*) dans un Tree de Root. La simulation peut aussi formater les signaux dans le format produit par le système en-ligne, c'est-à-dire en format « données bruts ». La sortie d'une simulation peut aussi être accompagnée de différentes distributions statistiques et des informations pour le contrôle de qualité de la reconstruction des trajectoires.

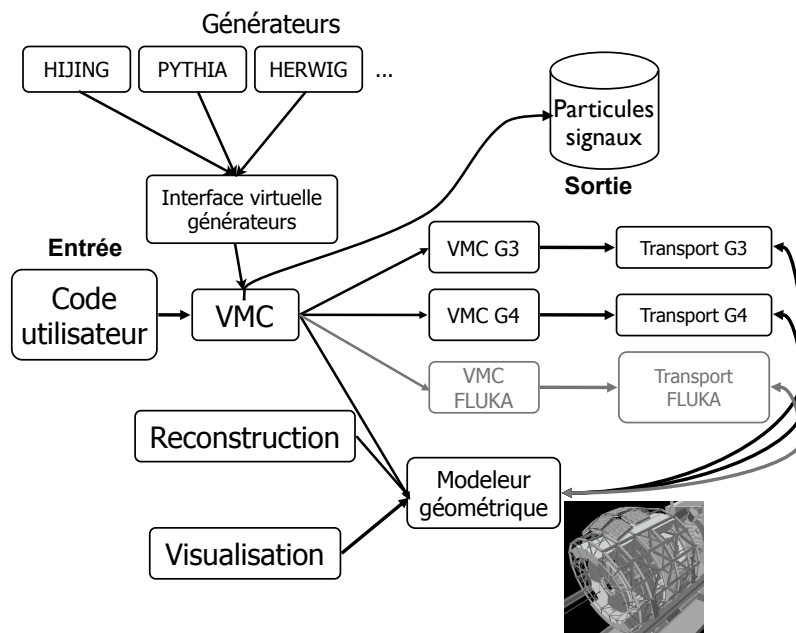


Figure 44 : Conception et schéma du MonteCarlo Virtuel.

Un des problèmes auxquels l'auteur a été confronté au début de AliRoot en 2000 fut la question de la simulation. L'expérience avait besoin d'un programme de simulation prédictif et complet pour la conception du détecteur. En 2000, il y avait trois possibilités.

GEANT 3 [52] avait été écrit, en FORTRAN, pour les expériences du LEP au début des années 1990. Il a été déclaré hors maintenance par le CERN. La partie électromagnétique de ce MonteCarlo avait été bien validée, tandis que la partie hadronique était assez approximative car basée sur des codes vieux de presque dix ans, et malgré des mises à jour plus récentes. L'utilisation de ce MonteCarlo aurait posé à court terme le problème de la transition de l'environnement FORTRAN au C++.

FLUKA [90] était et reste un des meilleurs programmes de transport de radiation dans la matière. Malheureusement son utilisation pour la simulation de la réponse des détecteurs est assez difficile à cause de sa structure informatique en

général, et en particulier de son programme de description de la géométrie du détecteur, qui est assez efficace mais d'utilisation peu aisée, et à cause aussi de son système d'accumulation des quantités à mesurer (en anglais *scoring*) très puissant mais peu flexible. De plus le code est en FORTRAN et l'accès au code source ainsi que l'utilisation avec d'autres programmes sont fortement limités par une licence d'utilisation plutôt restrictive. FLUKA a été utilisé par toutes les expériences LHC pour évaluer la quantité de radiation absorbée par les composants du détecteur, pour des calculs de radioprotection et pour des études poussées des performances des détecteurs en faisceau test (en anglais *test beams*). Par contre, aucune expérience (sauf ALICE, mais par le biais du Virtual MonteCarlo) n'a utilisé FLUKA pour la simulation de la réponse du détecteur complet (en anglais *full detector simulation*).

GEANT 4 [56] est le successeur de GEANT 3 et il a été publié au début de l'année 2000. Il est écrit en C++ et il est conçu pour la simulation des détecteurs de la *génération LHC*. Malgré la longue période de développement, il était encore en phase de validation, et donc il n'était pas envisageable de l'utiliser pour des études de conception du détecteur.

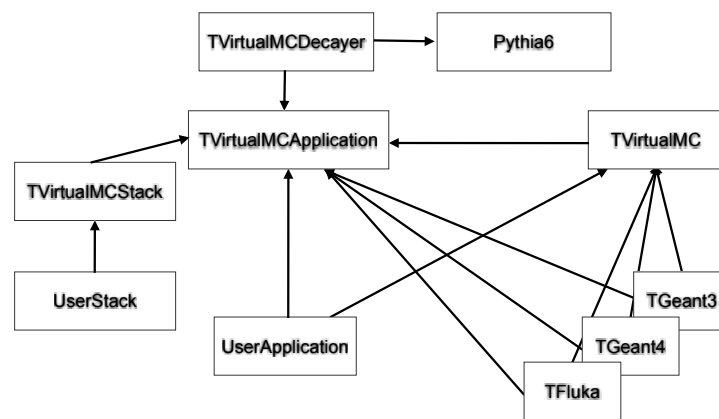


Figure 45 : Description de la structure des classes du MonteCarlo Virtuel.

Si d'un côté il était urgent de fournir une simulation prédictive et fiable, de l'autre aucun des choix disponibles ne paraissait satisfaisant. GEANT 3 était obsolète et sa maintenance n'était plus assurée. Même si une simulation avait été développée dans le cadre de GEANT 3, la migration à GEANT 4 était inévitable, avec un travail de changement de langage (de FORTRAN à C++) et de description géométrique (celle de GEANT 3 étant totalement incompatible avec celle de GEANT 4). L'utilisation de FLUKA pour une simulation de la réponse du détecteur était très difficile et la migration vers GEANT 4 encore plus ardue, car la géométrie de FLUKA est une forme de géométrie constructive des solides (CSG, en anglais *Constructive Solid Geometry*) [91] sans hiérarchisation des volumes comme dans le cas de GEANT 3 et GEANT 4. L'utilisation de GEANT 4 était prématurée car le programme n'était pas encore validé.

De plus, il y avait un problème « philosophique » qui rendait le choix encore plus difficile. Il était clair que le logiciel de ALICE serait en C++. Si l'adoption d'un programme de simulation en FORTRAN avait nécessité deux infrastructures,

l'« ancienne » en FORTRAN et la « nouvelle » en C++, avec une pénible transition en prime, l'adoption d'un système complexe comme GEANT 4 en C++ aurait eu pour effet d'« influencer » toute la future infrastructure de ALICE. De plus, le manque d'intégration entre GEANT 4 et Root et les incompatibilités au niveau de la conception entre les deux systèmes rendaient le problème encore plus aigu.

Pour résoudre cette situation, l'auteur de cette thèse a conçu et réalisé avec son équipe, et en étroite collaboration avec l'équipe Root, l'interface MonteCarlo Virtuel [92] (VMC, en anglais *Virtual MonteCarlo*).

Le VMC (voir Figure 44) définit une couche d'abstraction entre le code d'utilisateur pour la simulation du détecteur et le code de transport. De cette façon, le code utilisateur devient indépendant du MonteCarlo spécifique et il peut être utilisé avec les différents codes de transport GEANT 3, GEANT 4 et FLUKA avec le même logiciel de simulation.

Le VMC, inclut d'un côté l'interface avec le MonteCarlo de transport lui-même, `TVirtualMC`, et de l'autre côté l'interface avec l'application de l'utilisateur, `TVirtualMCApplication` (voir Figure 45). De cette façon, nous arrivons à découpler la dépendance entre le code utilisateur et le MonteCarlo de transport. En outre, on définit `TVirtualMCStack`, qui est l'interface à la liste de particules (en anglais *stack*) définie par l'utilisateur. Toutes ces classes sont disponibles avec quelques classes utilitaires de plus dans le paquet VMC du système Root.

Le grand avantage de l'introduction du VMC a été la possibilité de concevoir et développer une infrastructure en C++, tout en utilisant initialement le programme GEANT 3 en FORTRAN, « emballé » (en anglais *wrapped*) dans une interface C++. Grâce aux interfaces virtuelles, il a été facile d'effectuer ensuite la transition vers GEANT 4 et d'utiliser FLUKA pour des études particulières telles que l'estimation de la dose de radiation déposée par les neutrons de basse énergie ou la réponse du détecteur aux antiprotons. Tout cela a été fait sans changer ni la description géométrique du détecteur, ni le code pour la collecte et l'analyse des signaux. Le seul changement nécessaire est une ligne dans un fichier de configuration. Cela est à comparer avec la situation d'autres expériences qui ont dû développer une description du détecteur en FORTRAN pour GEANT 3, une nouvelle description en C++ pour GEANT 4, et, en parallèle, une description en « langage FLUKA » pour les études de radiation. Le fait de pouvoir changer seulement le MonteCarlo de transport en gardant le reste du code offre ainsi l'avantage de réduire considérablement le temps de développement et la probabilité d'introduire des erreurs et des inconsistances dans les différentes suites de simulations.

Encore plus important, cette technique offre la possibilité d'évaluer l'effet systématique des différents modèles de physique utilisés dans les MonteCarlo de transport, car tout le reste de l'environnement reste le même. Il est regrettable que l'interface avec FLUKA soit disponible seulement pour l'expérience ALICE à cause de la licence très restrictive de ce programme, car elle a permis d'obtenir des résultats assez intéressants.

L'implémentation du VMC a été le résultat d'une collaboration assez vaste à l'intérieur de ALICE et d'une remarquable quantité de travail. La première

interface avec GEANT 3 a été développée en grande partie par l'auteur lui-même, grâce à sa connaissance du programme GEANT 3, dont il avait été le chef de projet pendant trois années. L'interface avec GEANT 4 a demandé beaucoup plus de travail, car la structure du programme se prêtait très mal à être interfacé avec des éléments extérieurs. L'interface avec FLUKA a été aussi assez longue à mettre au point, surtout à cause, entre autres, des différences dans le maniement de la liste des particules entre FLUKA et GEANT.

L'élément fondamental du VMC est un modelleur géométrique, c'est-à-dire un logiciel de CAO (Conception Assistée par Ordinateur) capable de décrire un assemblage assez complexe de volumes, allant jusqu'à plusieurs millions d'objets, de matériaux différents. Pour une position et une direction donnée, ce logiciel doit pouvoir répondre très rapidement à trois questions :

1. Où (dans quel objet) suis-je ?
2. Quelle est la distance à la prochaine frontière en suivant la direction donnée ?
3. Quelle est la distance minimale à une frontière ?

Ces interrogations sont effectuées des milliards de fois pendant le transport des particules dans le détecteur, et donc elles doivent être implémentées d'une façon extrêmement efficiente. Il est donc évident que la description du détecteur doit être contenue dans la mémoire vive de l'ordinateur (RAM), vu que des accès au disque auraient pour effet de ralentir de façon inacceptable le calcul. Cela introduit la contrainte additionnelle que la description des plusieurs millions de volumes composant le détecteur doit être réalisée en utilisant une taille de mémoire « raisonnable », ce qui est souvent en contradiction avec la demande de rapidité d'exécution.

Il faut ici éclaircir deux aspects qui ne sont pas toujours évidents pour les non-experts. Bien que la CAO ait été un des premiers et plus actifs domaines de développement de logiciel, il n'existe pas sur le marché des logiciels, soit commerciaux soit en Open Source, capables de satisfaire les demandes que nous avons détaillées ci-dessous. Les seuls logiciels disponibles sont ceux qui ont été développés dans le cadre des MonteCarlo de transport, GEANT 3, FLUKA, GEANT 4, mais qui sont intimement liés au logiciel et donc quasiment impossibles à « extraire » pour être utilisés de façon indépendante. De plus, il est faux de croire qu'un programme de transport de particules n'a pas besoin d'une description très détaillée du détecteur. Malheureusement cela dépend de la nature du détecteur. S'il est vrai que (dans la plupart des cas) il n'est pas nécessaire de décrire chaque écrou ou boulon, il y a des détails parfois minimes qui ont un effet majeur sur la réponse d'un détecteur. Une couche de ruban adhésif peut altérer le spectre des neutrons thermiques et épi-thermiques car il contient de l'hydrogène, et une couche même très mince de matériel autour d'un détecteur gazeux peut changer la quantité d'électrons et photons qui entrent dans la partie sensible.

En tenant compte de tout cela, l'auteur a décidé de démarrer un projet en collaboration avec l'équipe Root pour la réalisation d'un modelleur géométrique indépendant, accessible depuis le VMC, et de l'interfacer avec les différents MonteCarlo. Ce développement a donné naissance à la librairie TGeo [93] de

Root. TGeo est une « boîte à outils » d'utilisation très générale pour la description d'assemblages complexes de volumes allant jusqu'à plusieurs millions dans un format très compact. Il a été optimisé comme modelleur géométrique pour le transport des particules. Il est partie intégrante de Root et il est utilisé par ALICE et par d'autres expériences de Physique des Hautes Énergies telles que PANDA, CBM, OPERA, STAR.

Avec TGeo nous avons réalisé une description complète de la géométrie du détecteur ALICE. TGeo est également capable de stocker et appliquer à la géométrie du détecteur les différences entre la position « idéale » des éléments et leur position réelle (désalignement, voir Section 4.6). Il intègre des outils de vérification, un constructeur interactif de géométrie et de puissants outils de visualisation. Il dispose aussi d'outils de diagnostic pour détecter les géométries « illégales » comportant des superpositions de volumes solides (en anglais *overlaps*).

TGeo a été interfacé à GEANT 3, GEANT 4 et FLUKA et validé pour les trois programmes. De plus, étant un paquet de Root, TGeo peut être utilisé indépendamment de la présence d'un programme MonteCarlo, et donc il est utilisé aussi pendant la reconstruction et pour la visualisation du détecteur et des événements.

Cela permet à ALICE de maintenir une seule et unique description de la géométrie et de l'utiliser pour la simulation et aussi pour la reconstruction des événements, ce qui garantit une très grande cohérence des résultats. À la connaissance de l'auteur, ALICE est la seule grande expérience en fonction qui, grâce à TGeo et au VMC, est capable d'utiliser exactement la même géométrie²⁰ pour la reconstruction et pour la simulation. Cette particularité a aussi permis à ALICE de développer un système d'alignement très performant que nous décrirons dans la prochaine section.

4.6 Système d'alignement et de calibrage

Le système d'alignement et de calibrage fournit l'infrastructure pour la production, le stockage et la récupération des données dites « conditions expérimentales » (en anglais *condition data*) nécessaires pour l'alignement et le calibrage hors-ligne des données brutes. Il s'agit essentiellement de données qui ne contiennent pas d'événements et qui sont utilisées pendant la reconstruction et l'analyse. La conception a été principalement dictée par le besoin d'avoir un accès transparent à un ensemble cohérent d'informations de calibrage et d'alignement dans une situation où les données et les ressources de calcul sont distribuées dans le monde entier dans de nombreux centres de calcul indépendants.

²⁰ Il faut préciser cette remarque. D'autres expériences ont choisi de décrire la géométrie avec un métalangage qui est réinterprété par les programmes de simulation et de reconstruction. C'est l'opinion de l'auteur que, malgré les avantages de cette approche, le danger reste d'une différence d'interprétation entre les divers logiciels qui lisent et interprètent l'entrée commune, tandis que dans le cas de ALICE ce danger est écarté parce qu'un seul et unique logiciel est utilisé.

Les informations sont organisées en unités qui sont sélectionnées pour être chargées dans le programme en exécution selon un ensemble de critères tels que le numéro de run, la « période » de prise de données, le détecteur, la date, le type de faisceau, la version des données requise et ainsi de suite. Dans une situation où les programmes sont exécutés « à proximité » des données et où il n'y a qu'une copie de ces données, l'utilisation d'une base de données relationnelle fournit une assez bonne solution.

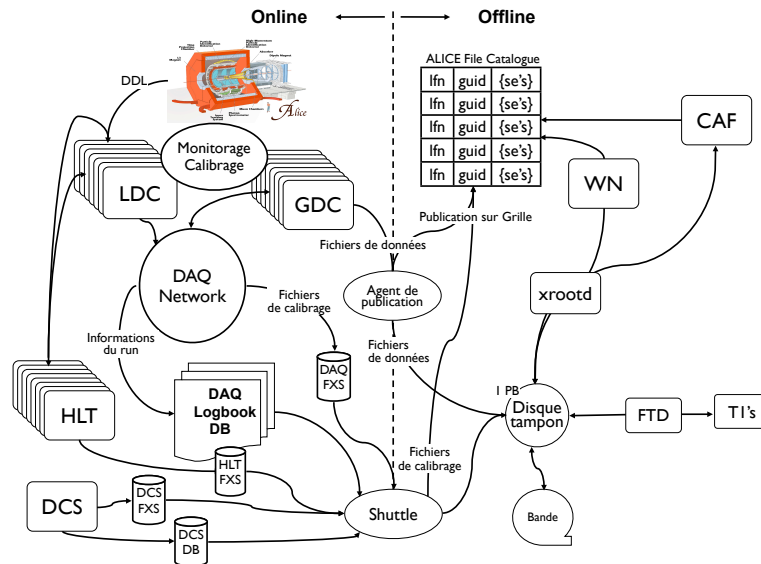


Figure 46: Vue schématique des relations entre les procédures d'alignement et de calibrage et les sources de données dans les systèmes AliRoot et AliEn.

Dans le cas où les données sont distribuées dans plusieurs dizaines de centres de calcul, la synchronisation des différentes bases de données relationnelles à travers des réseaux à grande distance (en anglais WAN, *wide area networks*) peut s'avérer problématique. De plus, ces données sont intrinsèquement à lecture seule, car normalement il y a qu'un programme d'écriture et plusieurs programmes de lecture. La plus grande partie de la fonctionnalité d'une base de données relationnelle, qui offre la possibilité de gérer plusieurs lectures et écritures des mêmes données à la fois, reste donc inutilisée. En outre, les mises à jours sont effectuées sur des collections « atomiques » de données qui vont de quelques kilooctet à plusieurs mégaoctets et non sur chaque donnée numérique indépendamment.

En tenant compte de tout cela, l'auteur a décidé de se départir de la solution classique et de stocker les unités d'information dans des fichiers au format Root en lecture seule. Les informations sur ces fichiers (métadonnées), qui permettent leur sélection, sont stockées dans le catalogue de fichiers de la Grille (voir Chapitre 5), dans une structure qui s'appelle base de données de calibrage hors-ligne (OCDB, en anglais *Offline Calibration Data Base*). Ce catalogue de fichiers associe les métadonnées décrivant les informations de calibrage et alignement avec les noms logiques²¹ des fichiers où l'information est stockée. Ces

²¹ Pour la définition de ces concepts voir Section 5.1.1.

métadonnées sont la principale méthode d'accès aux fichiers lors de la phase de production et d'analyse.

L'évolution des données de calibrage et d'alignement est gérée par le biais d'un schéma de versions des fichiers. Quand une nouvelle version d'une unité d'information est disponible, un nouveau fichier est introduit dans le catalogue avec une version plus élevée que la précédente.

L'utilisation de la possibilité qu'offre Root de stocker et relire les classes C++ avec évolution du schéma, associée aux métadonnées stockées dans le catalogue GRID des fichiers, évite le développement d'une structure plus complexe, mais fonctionnellement équivalente basée sur un système de bases de données relationnelles distribuées.

L'infrastructure fournit un ensemble de classes pour accéder et manipuler les objets d'alignement et de calibrage. Ceux-ci peuvent être stockés dans le système distribué de la Grille ou copiés sur le disque d'une machine déconnectée du réseau. Dans ce cas, les métadonnées sont encodées dans le nom du fichier. Par sécurité, les métadonnées sont aussi stockées dans le fichier même et elles sont vérifiées à l'ouverture du fichier.

La granularité minimale de validité des objets d'alignement et de calibrage est d'un run unique. Les objets, cependant, peuvent contenir des informations plus fines qui sont fonction du numéro de l'événement, du temps, ou d'autres variables, par exemple sous la forme d'un histogramme. Chaque sous-détecteur a sa propre partition de stockage d'objets, éventuellement avec une granularité spécifique. Il y a aussi des partitions contenant des informations communes pour tous les sous-détecteurs, par exemple les cartes de champ magnétique et les informations provenant de l'accélérateur.

Les données source pour le calcul des informations d'alignement et de calibrage ont des degrés de validité différents. Certaines sont stables pour plusieurs années, voire indéfiniment, d'autres varient pendant le même run. Les principales sources de ces données sont (voir Figure 46) :

- La base de données de construction du détecteur (DCDB) : utilisée par les différents sous-détecteurs au cours de la phase de production et d'intégration et contenant des informations statiques sur le détecteur et la performance des éléments, leur position exacte et leur identification.
- La base de données du système de contrôle de l'expérience (ECS, en anglais *Experiment Control System*) : contient des informations sur la configuration globale du détecteur et des différents sous-détecteurs qui participent à la prise des données dans chaque partition.
- La base de données du système d'acquisition des données et le cahier de bord électronique (en anglais *electronic logbook*) : contient les paramètres connexes avec l'acquisition de données et pour l'affectation des ressources aux tâches d'acquisition de données.
- La base de données du système de déclenchement : contient les classes de déclenchement (y compris la configuration du processeur central de déclenchement), et la définition des masques de déclenchement.

- La base de données du système de contrôle du détecteur (DCS en anglais *Detector Control System*) : contient les paramètres de configuration de l'électronique des différents détecteurs et plusieurs paramètres « environnementaux » contrôlés et archivés tout au long de la vie de l'expérience.
- Les informations provenant du système de déclenchement de haut niveau (HLT), comprenant les ESD produits en-ligne par le HLT pour les études de physique et pour la sélection des événements hors-ligne.
- La base de données de l'accélérateur LHC : contient l'état de la machine et les paramètres des faisceaux.
- Les données elles-mêmes, échantillonnées en-ligne par le LDCs et qui fournissent des informations pour le calibrage dès la fin du run pour une reconstruction rapide des événements.

L'infrastructure d'alignement et de calibrage permet de recueillir les informations provenant des diverses sources et de les rendre accessibles sur la grille pour le traitement distribué des données. Le flux des données est pour l'essentiel unidirectionnel vers l'environnement hors-ligne, à l'exception du système HLT, qui nécessite un accès aux données d'alignement et de calibrage, mais peut aussi en produire.

Les données produites par les différentes sources sont déposées sur des serveurs de stockage (*File eXchange Server*, FXS) pendant la prise de données. Un programme est actif pendant la prise de données pour recueillir ces informations, les formater sous la forme de fichiers Root et ensuite, à la fin du run, les stocker dans l'élément de stockage du CERN et les enregistrer dans le catalogue de fichiers de la Grille. Ce programme est appelé *Shuttle* (anglais pour *navette*) et le principe de son fonctionnement a été conçu par l'auteur et réalisé par son équipe. Les informations spécifiques de chacune prise de données sont enregistrées dans un fichier spécial qui contient les paramètres généraux du run communs à tous les détecteurs (GRP, en anglais *Global Run Parameters*).

Une fois les fichiers de calibrage générés en-ligne stockés dans l'OCDB, ils deviennent accessibles partout sur la Grille. Le calibrage qui peut être effectué en-ligne est très approximatif, car il est basé au mieux sur un échantillonnage des événements. Le calibrage qui peut être effectué hors-ligne est de meilleure qualité, et évidemment beaucoup plus coûteux en termes de ressources de calcul. Dans le modèle original de calcul de ALICE une première reconstruction devait suivre immédiatement la prise des données basée sur les données de calibrage produites en-ligne. Pendant cette reconstruction, une nouvelle version des données de calibrage devait être produite et stockée dans l'OCDB pour permettre une deuxième reconstruction plus précise, qui, à son tour, aurait produit des données de calibrage améliorées pour la troisième reconstruction, censée être la dernière.

Dans la pratique, ce modèle a dû être adapté à la réalité. Des problèmes de matériel et de maturité des algorithmes n'ont pas permis de produire en-ligne des informations de calibrage suffisamment précises pour une première reconstruction « dans la foulée » ayant une utilité quelconque pour la physique. Le modèle a donc été modifié. Après la prise des données, les algorithmes de

calibrages sont appliqués aux données brutes pour produire des fichiers de calibrage qui permettent une reconstruction utilisable pour la physique. De plus, cette procédure est itérative : deux passages, effectués sur une sélection de données correspondant plus ou moins à 10 % du total, sont requis pour obtenir le calibrage. Pendant ce traitement, des informations sur la qualité du calibrage sont produites pour vérifier qu'elle est correcte.

Les fichiers de calibrage sont utilisés non seulement pendant la reconstruction, mais aussi pendant la simulation. Pour obtenir une simulation la plus réaliste et prédictive possible, qui permette de calculer, par exemple, l'efficacité de détection d'une particule en fonction de son impulsion, il est important d'utiliser le même programme de reconstruction pour les données brutes provenant de la simulation. Pour cela, il faut que les données simulées soient « décalibrées », c'est-à-dire qu'elles reproduisent avec exactitude les variations de la réponse des différents éléments de détections, du même type au même signal d'entrée. L'infrastructure de calibrage permet au programme de simulation de lire les fichiers de calibrage et de modifier la réponse simulée de façon à reproduire les imperfections et les caractéristiques réelles de chaque élément de détection. Cette fonctionnalité a été conçue par l'auteur et implémentée par son équipe. On peut donc « ancrer » chaque simulation à une prise de données spécifique en reproduisant exactement les conditions expérimentales. Cette fonctionnalité est indépendante du programme de transport utilisé, car la fonctionnalité est abstraite dans le paquet VMC. Cela a aussi permis de mettre au point les procédures de calibrage des données avant même la construction du détecteur.

L'infrastructure d'alignement [94] permet d'appliquer à chaque élément géométrique du détecteur une translation et une rotation pour le déplacer de sa position « idéale » vers sa position « réelle », différente de la position idéale à cause à la fois des erreurs de construction et des déformations du détecteur résultant de son poids ou des champs magnétiques. Cette transformation est appelée *delta*.

Le détecteur une fois assemblé, la position de ses éléments principaux est mesurée par des équipes de géomètres par des moyens géodésiques. Certains détecteurs ont aussi des appareillages intégrés pour la détermination par interférométrie optique des positions relatives des éléments. Les résultats de ce processus sont enregistrés dans des fichiers de positions réelles, traitées afin d'obtenir les transformations *delta*. Ces transformations sont appliquées à la géométrie du détecteur décrite par TGeo, qui est ensuite écrite dans un fichier utilisé par la simulation et la reconstruction.

Même si les mesures géodésiques sont très précises, elles peuvent seulement déterminer la position d'un nombre limité d'éléments. Par exemple elles peuvent établir la position de l'ITS, mais pas celle des *échelles* et encore moins celle des plaquettes de silicium. Ces positions sont obtenues en utilisant les traces des particules à l'aide d'une procédure itérative qui minimise les écarts entre la position des signaux générés par les particules et les trajectoires reconstruites.

Cette opération est très complexe à cause du grand nombre de degrés de liberté, six pour chaque élément de détection : sa position et son orientation dans l'espace. Des logiciels spécifiques ont été développés pour résoudre ce problème et ils sont intégrés dans AliRoot.

Ces corrections génèrent de nouveaux ajustements (alignements résiduels) qui sont appliqués à la géométrie. Il serait possible, en principe, de générer une nouvelle géométrie avec ces alignements intégrés, mais comme ces corrections évoluent rapidement en suivant l'amélioration des algorithmes de reconstruction, qui dépendent du calibrage et de l'alignement eux-mêmes, nous préférons garder une géométrie « stable » qui contient seulement les corrections géodésiques et appliquer dynamiquement les corrections calculées.

L'infrastructure d'alignement conçue par l'auteur et implémentée par son équipe a permis de mettre au point et de vérifier ces logiciels de façon très précise selon la procédure suivante. Chaque élément sensible d'un détecteur est déplacé de sa position idéale de façon aléatoire dans la limite des imprécisions de localisation. Un deuxième déplacement est généré, toujours de façon aléatoire, de grandeur inférieure à la précision des mesures géodésiques. Une simulation est effectuée après avoir appliqué à la géométrie du détecteur les deux transformations. La reconstruction des trajectoires est effectuée avec une géométrie où seule la correction « géodésique » est appliquée, mais pas la deuxième qui est plus petite. Cela reproduit la situation réelle où les trajectoires laissent des traces dans le détecteur réel (somme des deux déplacements), mais sont initialement reconstruites dans un détecteur qui est corrigé seulement pour les mesures géodésiques. Les algorithmes d'alignement censés déterminer l'alignement résiduel sont ensuite appliqués et les résultats confrontés avec la deuxième transformation.

Cette stratégie a permis une mise au point de procédures d'alignement très pointues avant même que le détecteur soit opérationnel. L'infrastructure d'alignement permet de reproduire avec une grande exactitude l'alignement des éléments sensibles du détecteur, qui dans d'autres expériences doit être inclus dans les erreurs systématiques.

La possibilité d'appliquer des « déplacements » à une géométrie CSG a demandé un développement assez important de TGeo auquel l'auteur a pris part dans la phase de conception. En effet, une des caractéristiques principales des géométries CSG est de permettre la description de structures répétées (volumes physiques) à partir d'un prototype (volume logique) et d'un algorithme de répétition (positionnement répété dans l'espace ou division). Cela permet une grande économie d'espace mémoire et aussi une optimisation de la navigation, mais seulement si les structures sont régulières. L'introduction de modifications dans chaque élément d'une structure répétée non seulement peut causer une inflation de la mémoire, mais aussi une dégradation sérieuse de la performance de navigation. Pour pallier ces dangers, il a été nécessaire de mettre au point des algorithmes assez sophistiqués qui sont décrits dans le manuel de TGeo [93].

4.7 La reconstruction

L'objectif principal de la reconstruction est de fournir les données nécessaires à l'analyse physique à partir des données brutes enregistrées par le détecteur. Le système de reconstruction est la phase centrale du traitement de données. La qualité de la reconstruction est déterminante pour la qualité des résultats de physique. Il est essentiel que la résolution spatio-temporelle des paramètres des trajectoires et la précision avec laquelle les particules sont identifiées soient

maximisées. La reconstruction doit aussi être efficace, car c'est l'une des activités les plus lourdes en termes de ressources de calcul. Le code de reconstruction évolue rapidement, car il doit s'adapter aux conditions expérimentales et refléter le progrès dans la compréhension du détecteur. Pour cette raison il doit être facile à utiliser et à maintenir.

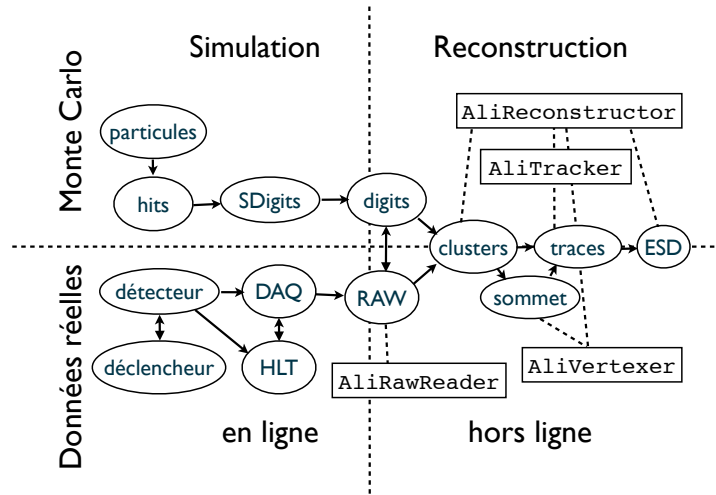


Figure 47: Aperçu du flux général des données pendant la reconstruction.

Le système de reconstruction de ALICE fait partie du système AliRoot. Sa conception modulaire permet à son code d'être regroupé dans différentes bibliothèques partagées et exécutées indépendamment des autres parties de AliRoot. Par défaut, la reconstruction utilise au format brut les données produites par le détecteur ou générées à partir des simulations. La reconstruction peut aussi être faite à partir des « signaux numérisés » au format Root produits par la simulation, ce qui est plus pratique pour le développement et le débogage. La sortie de la reconstruction sont les fichiers ESD contenant les trajectoires reconstruites des particules chargées et l'information sur leur identité, les topologies dites V^0 (désintégrations des particules neutres, telles que $\Lambda \rightarrow p\pi$), coudes (en anglais *kinks*, comme $K^\pm \rightarrow \mu\nu$) et cascades (désintégrations du type $\Xi \rightarrow \Lambda\pi \rightarrow p\pi\pi$) et les particules neutres reconstruites dans les calorimètres.

Pour garantir la flexibilité nécessaire, AliRoot permet de reconstruire les données d'un détecteur, même si d'autres détecteurs ne sont pas opérationnels. Par exemple chacun des détecteurs de trajectographie du « baril » est capable de trouver les traces à partir d'informations génériques (en anglais *seeds*) fournies par un autre détecteur, ou de les trouver seulement à partir de l'information interne du détecteur. De plus, chaque détecteur est capable de propager dans les deux directions (vers le vertex et à partir du vertex, lorsque cela est applicable) les trajectoires provenant de l'extérieur en ajoutant ses propres clusters à la trajectoire.

Nous avons fait un très large usage des classes de base communes qui sont spécialisées pour chaque détecteur selon ses caractéristiques. Nous avons aussi minimisé les interdépendances entre les modules de reconstruction en utilisant

une classe commune pour décrire l'évolution de l'information sur les particules reconstruites.

Le système de reconstruction (voir Figure 47) utilise comme données d'entrée les signaux numérisés ou les signaux bruts rangés dans un `TTree` de `Root`. D'autres informations sont les fichiers de calibrage, les cartes du champ magnétique et la géométrie du détecteur. Ces données sont lues au début de la reconstruction et sont valables pendant toute la reconstruction d'une période de prise de données.

La classe principale de reconstruction qui contrôle la boucle des événements et le chargement des `TTree` est `AliReconstruction`. Elle fournit une interface simple pour la reconstruction et permet de configurer la procédure, d'inclure ou d'exclure un détecteur, et d'assurer le bon ordre des étapes de la reconstruction :

- les algorithmes qui sont exécutés pour chaque détecteur séparément (reconstruction locale, un exemple typique est la recherche de clusters) ;
- la localisation du point d'interaction;
- la reconstruction des trajectoires et l'identification des particules (PID) ;
- la reconstruction des vertex secondaires (topologies de désintégration du type V^0 , coude et cascade).

La classe `AliReconstruction` gère le flux du programme de reconstruction par le biais d'une série d'interfaces virtuelles. Au lieu d'exécuter la reconstruction locale directement sur les données brutes, il est possible de les convertir d'abord en données numérisées dans un `Tree Root`.

Cette classe est également responsable de l'interaction avec le sous-système d'entrée sortie de `AliRoot`. L'interface entre la classe de contrôle `AliReconstruction` et le code spécifique à chaque détecteur est définie par la classe de base `AliReconstructor`. Pour chaque détecteur il y a une classe de reconstruction dérivée. L'utilisateur peut définir des options pour chaque reconstruteur utilisant des chaînes de caractères (strings). Grâce à l'utilisation de classes virtuelles, l'utilisateur peut aussi changer les algorithmes de reconstruction sans modifier le code de contrôle, en définissant une « réalisation » différente de la classe virtuelle de reconstruction pour un détecteur spécifique.

Par défaut, le nom du reconstruteur de détecteur `DET` est `AliDETReconstructor` et il se trouve dans la bibliothèque `libDETrec.so`. Les utilisateurs peuvent définir les bibliothèques qui contiennent le code et le nom de la classe de reconstruction via le gestionnaire d'extension de `Root` (en anglais *plugin*).

Le premier pas de la reconstruction est la *reconstruction locale* qui a lieu dans chaque détecteur de façon indépendante. Il s'agit de déterminer dans chaque détecteur les positions et parfois les temps de passage des particules à partir d'ensembles de dépôts d'énergie, laissées par les particules, groupées dans l'espace et dans le temps (en anglais *cluster*).

Les algorithmes de définition des clusters sont spécifiques des différents détecteurs. Par ailleurs, au sein d'un détecteur donné, en raison du chevauchement des clusters dû au taux d'occupation élevé, la position et le

nombre des clusters et les paramètres des traces ne sont pas indépendants. Le nombre et la position des clusters sont fixés seulement quand les paramètres des trajectoires sont définis, dans une procédure itérative.

Pour chaque point de l'espace-temps, défini comme le centroïde du cluster, il faut aussi calculer l'incertitude de l'estimation. Tous les détecteurs de trajectographie centraux (ITS, TPC, TRD) ont leur propre paramétrisation détaillée des incertitudes de la position et du temps des points, cependant certains paramètres peuvent être fixés uniquement à l'étape de la recherche de la trajectoire (voir ci-dessous). Les points de l'espace-temps, ainsi que leurs incertitudes, sont ensuite transmis à l'algorithme de reconstruction des trajectoires.

Si, en plus de la position du point dans l'espace-temps, le détecteur est également capable de mesurer la quantité d'énergie déposée par la particule (ionisation), cette information peut être utilisée pour l'identification des particules.

Les implémentations spécifiques de l'algorithme de reconstruction des trajectoires utilisent un ensemble de classes de base communes, ce qui rend facile le passage des trajectoires d'un détecteur à l'autre et de tester différentes parties de la chaîne de reconstruction. Par exemple, on peut facilement basculer entre les différentes implémentations de l'algorithme pour trouver les clusters, ou de la procédure d'ensemencement des trajectoires.

La reconstruction des événements simulés peut se faire à partir des positions exactes des signaux, éventuellement étalées de façon aléatoire, plutôt qu'à partir des positions reconstruites de la réponse simulée du détecteur, ce qui est très utile pour effectuer des tests. Toujours dans le cas de la simulation, chaque signal contient les informations sur la particule qui l'a généré. Bien que cela implique un stockage d'informations supplémentaire, il a été prouvé être très utile pour le débogage du code de reconstruction.

La reconstruction globale commence par la détermination de la position des vertex primaires²². Ceux-ci peuvent être trouvés avant d'identifier les trajectoires par une simple corrélation des points reconstruits dans les deux couches de pixels de l'ITS. Comme cela a été démontré dans [95], une précision de 25 μm dans la direction du faisceau et d'environ 5 μm dans le plan transversal est facilement atteinte pour des événements de multiplicité suffisamment élevée. L'information sur la position des vertex primaires, avec son incertitude, est ensuite utilisée lors de la recherche des trajectoires (ensemencement et application de la contrainte de vertex), et pour la reconstruction des vertex secondaires.

Les vertex principaux sont reconstruits par un objet dérivé de `AliVertexer`. Après la reconstruction locale, la méthode `FindVertexForCurrentEvent` est appelée. Elle retourne un pointeur vers une liste d'objets de type `AliESDVertex`.

Après reconstruction des trajectoires, une meilleure détermination de la position des vertex dans le plan transversale aux faisceaux peut être effectuée. Cela est

²² Dans les collisions proton-proton par exemple nous pouvons avoir plus d'une collision par événement.

nécessaire à la reconstruction des mésons B et D dans les événements proton-proton.

La sortie de la reconstruction est un ensemble des fichiers ESD contenant des événements reconstruits.

Le détecteur de l'expérience ALICE est composé de plusieurs sous-ensembles qui utilisent des techniques de détection très variées. Ce n'est pas notre intention de fournir une description complète de ces techniques (voir le chapitre 5 du ALICE Physics Performance Report. [98]). Néanmoins nous donnerons une brève description de la procédure pour trouver les trajectoires des particules et les identifier dans le «baril» et dans le bras MUON. Cela nous fournira un aperçu de la complexité du processus de reconstruction.

La mise au point d'une telle procédure de reconstruction a nécessité beaucoup de temps et d'essais. Cela a commencé bien avant que le détecteur soit construit, avec la reconstruction des signaux générés par les logiciels de simulation. Pour contrôler la qualité de la reconstruction, pendant la simulation, les paramètres de chaque particule sont stockés à différents endroits de sa trajectoire par le biais de la classe `AliTrackReference`. Pendant la reconstruction, les mêmes paramètres sont stockés aux mêmes endroits pour les trajectoires reconstruites par le biais de la même classe. Ainsi, les trajectoires reconstruites peuvent être facilement comparées avec les particules simulées. Cela permet d'étudier et de contrôler dans le détail les performances de la reconstruction trajectographique.

4.7.1 Reconstruction dans les détecteurs centraux

Selon l'utilisation de l'information fournie par les clusters, les méthodes de suivi des traces peuvent être divisées en deux grands groupes: les méthodes globales et les méthodes locales. Chaque groupe a ses avantages et ses inconvénients.

Avec les méthodes globales, toutes les mesures de la trace sont traitées simultanément et la décision d'inclure ou d'exclure une mesure est prise lorsque toutes les informations sur la trace sont connues. Les algorithmes typiques appartenant à cette classe sont les méthodes combinatoires, la transformation de Hough, les modèles de trajectoire et les mappings conformes. Les avantages sont leur stabilité par rapport au bruit et aux erreurs de mesure et la possibilité d'agir directement sur les données brutes. Ces méthodes nécessitent un modèle de trajectoire assez précis. Un tel modèle peut parfois être inconnu ou même ne pas exister en raison des processus stochastiques (pertes d'énergie, diffusion multiple), de la non-uniformité du champ magnétique, etc. Les méthodes globales sont largement utilisées dans le déclencheur de haut niveau (HLT) optimisé pour la reconstruction des traces de grande impulsion: la précision requise n'est pas cruciale, mais la vitesse des calculs est d'une grande importance.

Les méthodes locales n'ont pas besoin de connaître le modèle global de trajectoire. Les paramètres de la trace sont toujours estimés « localement » en un point donné dans l'espace. La décision d'accepter ou de rejeter une mesure (cluster) comme appartenant à une trajectoire donnée est prise en utilisant les informations locales et celles provenant de l'« historique » de cette trajectoire. Avec ces méthodes, toutes les particularités locales d'une trajectoire (processus physiques stochastiques, champ magnétique, géométrie du détecteur) peuvent

naturellement être prises en compte. Les méthodes locales reposent sur des algorithmes sophistiqués de reconstruction des points spatiaux (y compris la déconvolution des clusters qui se chevauchent). Ils sont sensibles au bruit, aux mesures fausses ou déplacées et à la précision du paramétrage des erreurs sur la position du signal. Nous utilisons l'une des méthodes les plus avancées de reconstruction trajectographique locale, appelée le filtre de Kalman, qui a été introduite par P. Billoir en 1983 [96].

Le filtre de Kalman présente plusieurs propriétés intéressantes. Il permet d'effectuer simultanément la reconnaissance d'une trajectoire et son ajustement (fit) aux données. Pendant le suivi de la trace, il est possible d'accepter ou de rejeter chaque point sur la base des points déjà acquis, tandis que pour les autres méthodes il est nécessaire d'effectuer un (ou plusieurs) passage(s) additionnel(s) pour se débarrasser des points attribués à tort. A chaque point accepté, les paramètres de la trajectoire ainsi que la matrice de covariance entre ces paramètres sont mis à jour.

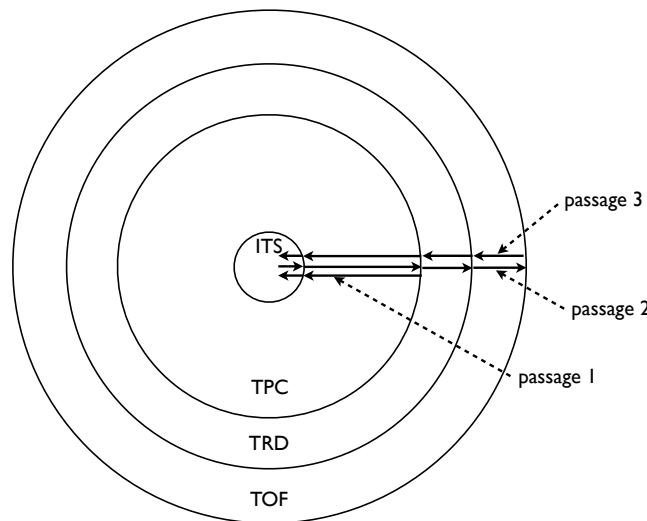


Figure 48: Vue schématique des trois passages pour la reconstruction des trajectoires dans le « baril ».

Pour les méthodes globales, dans le cas d'une diffusion multiple importante, les mesures de la trajectoire sont corrélées, et donc des matrices de grande taille (de l'ordre du nombre de points de mesure) doivent être inversées lors d'un ajustement global. Dans le filtre de Kalman, il n'est nécessaire de manipuler que des matrices 5×5 , ce qui est beaucoup plus rapide, même si l'on doit le faire pour chaque point mesuré. En outre, nous pouvons traiter la diffusion multiple et les pertes d'énergie beaucoup plus facilement qu'avec les méthodes globales. Le filtre de Kalman nous permet aussi d'extrapoler une trace d'un détecteur à un autre de façon tout à fait naturelle (par exemple de la TPC à l'ITS ou au TRD).

En ALICE, nous avons besoin d'une bonne efficacité de détection et précision de reconstruction pour mesurer des particules à partir d'une impulsion transversale de 100 MeV/c. Certains des détecteurs ont un budget matériel important (ITS $\sim 7,3$ % de longueur de radiation, TRD ~ 24 % de longueur de radiation). Dans ces conditions, on ne peut négliger les pertes d'énergie ou la

diffusion multiple dans la reconstruction. Il y a également d'assez grandes zones mortes entre les détecteurs trajectographiques, ce qui complique la recherche de la continuation d'une trace d'un détecteur à l'autre. Pour toutes ces raisons, nous avons choisi le filtre de Kalman pour la reconstruction hors-ligne depuis 1994.

Dans le « baril », c'est-à-dire la partie centrale du détecteur située dans l'aimant solénoïdal, la reconstruction des traces essaie d'accumuler l'information de différents détecteurs afin d'optimiser les performances de la reconstruction (reconstruction combinée). Le résultat est stocké dans des objets contenant l'information combinée des trajectoires (`AliESTrack`). Cette classe offre la possibilité d'échanger des informations entre les détecteurs sans introduire des dépendances entre les modules de reconstruction. La liste des traces combinées est conservée en mémoire et transmise d'un module de reconstruction à l'autre comme argument d'une fonction C++. Les classes chargées de la reconstruction des trajectoires sont dérivées de `AliTracker`.

La reconstruction des traces se fait en trois passages (voir Figure 48). Nous partons de notre meilleur dispositif de traçage, à savoir la TPC, dans les couches extérieures où la densité des signaux est minimale. Tout d'abord nous identifions les *graines* (en anglais *seed*) pour chaque trace, c'est-à-dire trois points (clusters) qui sont alignés avec et sans la contrainte de pointer vers le vertex principal. C'est une procédure itérative où les contraintes relatives aux vertex sont relâchées progressivement. En raison du nombre restreint de clusters assignés à une trajectoire-graine, la précision de ses paramètres ne suffit pas à l'extrapoler vers l'extérieur aux autres détecteurs. Au lieu de cela, le suivi se dirige vers l'intérieur de la TPC. Chaque fois que possible, de nouveaux clusters sont associés à une trajectoire candidate en utilisant le filtre de Kalman et les paramètres de cette trajectoire sont de plus en plus raffinés. Lorsque toutes les graines sont extrapolées à la limite intérieure de la TPC, le processus continue avec les clusters de l'ITS. L'algorithme de traçage de l'ITS cherche à prolonger la trajectoire de la TPC en s'approchant le plus possible du vertex principal. En faisant cela, les clusters très précis de l'ITS sont assignés à la trace, ce qui améliore l'estimation de ses paramètres. Si une prolongation n'est pas trouvée, la contrainte relative au vertex principal est relâchée et la prolongation est réessayée. Cela permet de trouver des vertex secondaires à l'intérieur de l'ITS et de définir avec précision les trajectoires qui sont candidates à être identifiées comme « primaires », c'est-à-dire issues directement d'un vertex primaire, et celles qui sont candidates à être identifiées comme « secondaires ».

Après que toutes les trajectoires candidates provenant de la TPC ont été prolongées jusqu'au vertex principal et leurs clusters dans l'ITS identifiés, un algorithme de traçage spécifique est appliqué au reste des clusters de l'ITS. Cette procédure tente d'identifier les particules qui n'ont pas été trouvées dans la TPC en raison de leur p_t trop faible ou parce qu'elles se sont perdues dans les « zones mortes » entre les secteurs de la TPC, ou encore parce qu'elles se sont désintégrées.

Après cela, la reconstruction est reprise à partir du vertex vers l'extérieur. Les paramètres de chaque trajectoire contiennent maintenant toute l'information de la TPC et de l'ITS, et donc ils permettent un « filtrage » Kalman assez précis des clusters. Aux particules qui ont été identifiées comme issues du vertex principal,

sont attribuées plusieurs hypothèses de masse, chacune avec sa probabilité. Ces hypothèses sont ensuite utilisées pour l'identification des particules (PID) dans le détecteur TOF. Une fois le rayon extérieur de la TPC atteint, la précision des paramètres estimés des trajectoires est suffisante pour les extrapoler aux détecteurs extérieurs (TRD, TOF, HMPID et PHOS). Les points trouvés dans ces détecteurs vont améliorer ultérieurement la précision des paramètres de trace, surtout pour les particules à haut p_t , grâce au long bras de levier. Les informations fournies par ces détecteurs contribuent aussi à améliorer l'identification des particules.

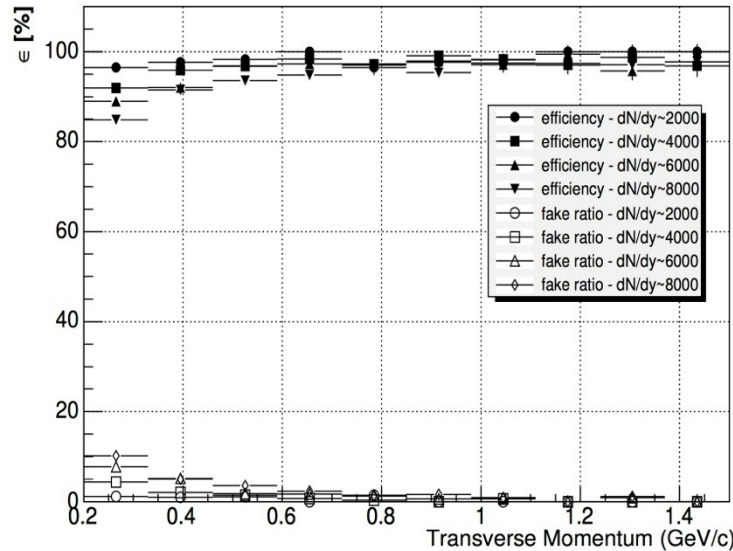


Figure 49: Efficacité combinée des reconstruction (symboles pleins) et de la probabilité d'obtenir une piste fautive (symboles vides) en fonction de l'impulsion transverse pour des différentes densités de particules chargées.

Nous sommes maintenant dans la situation où l'information sur les particules est optimale à la fin de leur trajectoire dans le détecteur, tandis que nous souhaiterions optimiser l'information au niveau de leur vertex d'origine, c'est-à-dire de leur point de génération. Pour cela, un troisième pas de reconstruction est nécessaire jusqu'au vertex d'origine.

Les trajectoires qui n'ont pas passé le filtrage final vers le vertex principal sont utilisées pour trouver les vertex secondaires des V^0 , coudes et cascades. Il y a aussi une option pour reconnaître les vertex secondaires « dans la foulée » pendant le suivi des trajectoires, qui toutefois n'est pas encore utilisée.

Le schéma décrit ci-dessus n'est pas encore complètement appliqué. Des problèmes de calibrage et d'alignement des détecteurs extérieurs par rapport à la TPC sont apparus et donc le troisième passage démarre de la paroi extérieure de la TPC sans tenir compte des paramètres obtenus avec l'extrapolation vers les détecteurs extérieurs. Cela est considéré comme une mesure temporaire.

Les traces reconstruites, avec leur identification, et les vertex secondaires, sont stockées dans le ESDs. Un ESD typique pour une collision centrale Plomb-Plomb contient environ 10^4 particules reconstruites, quelques centaines de candidats V^0 et coudes, et quelques dizaines de candidats de cascades.

La Figure 49 montre l'efficacité idéale de la reconstruction des traces en fonction de l'impulsion des particules pour des événements de différentes multiplicités. L'efficacité est définie comme le rapport entre le nombre de traces reconstruites et le nombre de traces qui peuvent être reconstruites [97]. Certaines trajectoires reconstituées peuvent être associées à des clusters qui n'ont pas été générés par cette particule. La probabilité d'obtenir de telles « fausses » pistes est représentée dans la même figure. On peut voir que, même dans le cas des événements de multiplicité la plus élevée, l'efficacité est toujours supérieure à 85 % et le nombre de « fausses traces » ne dépasse jamais quelques pour cent.

Plus de détails sur les algorithmes de reconstruction peuvent être trouvés dans le chapitre 5 du rapport ALICE Physics Performance Report [98].

4.7.2 Reconstruction dans le spectromètre MUON

Une procédure de reconstruction distincte existe pour le spectromètre MUON. La logique de déclenchement est basée sur l'exigence d'avoir au moins trois signaux (sur un total de quatre chambres) dans les stations de déclenchement, à la fois dans la direction de la déflexion due au champ dipolaire et dans la direction perpendiculaire à la déflexion. De cette façon, on peut définir une « trajectoire de déclenchement », et calculer son écart par rapport à une particule d'impulsion infinie. En exigeant que cette déviation soit inférieure à une certaine valeur, on peut sélectionner des trajectoires candidates ayant une impulsion transversale supérieure à une valeur prédéfinie. Une telle sélection est utilisée pour rejeter les muons « mous », provenant majoritairement de la désintégration des π et K, et est capable de limiter le taux de déclenchement du bras di-muon lorsque la luminosité du LHC est élevée.

La reconstruction des trajectoires est basée ici aussi sur le filtre de Kalman. La procédure démarre à partir des chambres de trajectographie les plus en aval (4 et 5), qui sont moins affectées par le bruit de fond des particules molles qui échappent à l'absorbeur frontal. Des segments droits sont formés en joignant les clusters sur les deux plans de chaque station et une première estimation des paramètres de trajectoire (position, pente et inverse de l'impulsion) et des erreurs correspondantes est effectuée. L'impulsion est estimée en supposant la trajectoire comme provenant du vertex principal et courbée par le champ magnétique constant dans le dipôle. Dans une seconde étape, les trajectoires candidates de la station 4 sont extrapolées à la station 5 (ou vice versa) et couplées à au moins un cluster, selon une sélection basée sur le χ^2 . Si plusieurs clusters sont trouvés, à chaque combinaison possible est associé un candidat de trajectoire. Après cette association, les paramètres et les erreurs pour la trajectoire sont recalculés en utilisant le filtre de Kalman. La même procédure est répétée pour les stations situées en amont, en rejetant, à chaque étape, les candidats pour lesquels aucun cluster n'est trouvé, ou ceux dont les paramètres indiquent qu'ils quitteraient l'acceptance géométrique du spectromètre dans les étapes suivantes.

À la fin de la procédure, des algorithmes supplémentaires sont appliqués pour améliorer la qualité de la trajectoire en ajoutant ou supprimant des clusters suivant une coupe en χ^2 , et en supprimant les fausses trajectoires, qui sont celles composées de clusters qui ont été assignés à d'autres trajectoires. Enfin, les

trajectoires restantes sont extrapolées à la position du vertex principal déterminé par les deux couches internes de l'ITS, et leurs paramètres sont recalculés en tenant compte de la perte d'énergie et de la diffusion coulombienne multiple dans l'absorbeur. Elles sont également recoupées avec les « trajectoires de déclenchement » pour identifier les muons responsables du signal de déclenchement.

4.7.3 Identification des particules chargées

L'identification des particules sur une grande gamme d'impulsions et pour de nombreuses espèces de particules est souvent l'une des principales exigences des expériences de Physique des Hautes Énergies. L'expérience ALICE est capable d'identifier des particules d'impulsion allant de 0,1 GeV/c jusqu'à, dans certains cas, au-dessus de 10 GeV/c.

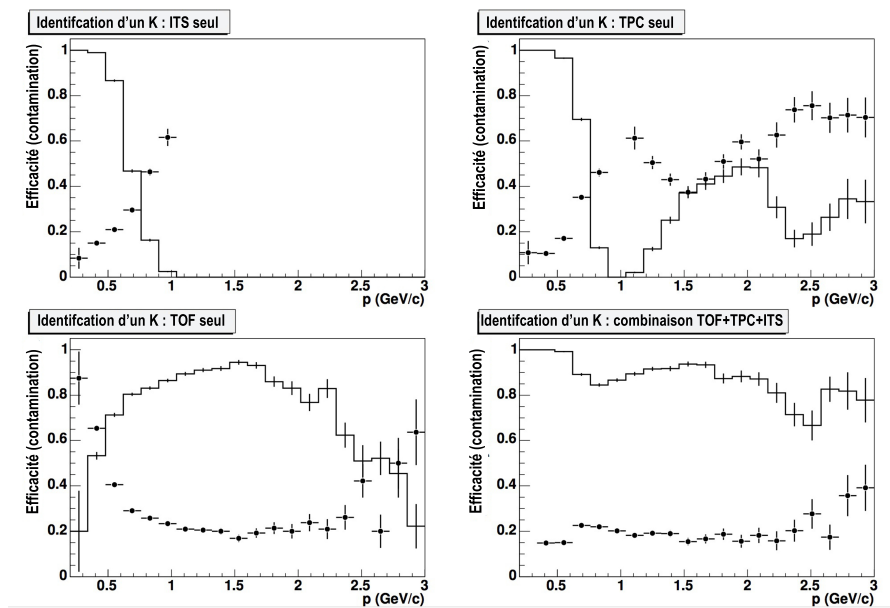


Figure 50 : Efficacité et contamination d'identification d'un Kaon pour ITS, TPC et TOF seuls (trait plein) et pour leur combinaison (points avec barres d'erreur).

Cela peut être obtenu en combinant plusieurs systèmes de détection qui sont efficaces dans des sous-intervalles d'impulsion, et donc complémentaires. Cette combinaison se fait suivant une approche bayésienne [99] qui a plusieurs avantages. Elle peut combiner les signaux PID de nature différente (par exemple, dE/dx et temps de vol) provenant de différents détecteurs. Quand seule une partie des détecteurs a identifié une particule, les signaux provenant des autres détecteurs ne dégradent pas l'identification. En outre, cette méthode prend en compte le fait que l'identification d'une particule dépend de la sélection des particules et des événements

La performance d'identification des kaons chargés dans des événements Plomb-Plomb simulés avec HIJING à $\sqrt{s} = 5,5$ TeV à l'aide de ITS, TPC et le TOF seuls et les résultats de leur combinaison selon l'algorithme de AliRoot sont présentés dans la Figure 50. L'efficacité d'identification est définie comme le rapport du nombre de particules correctement identifiées au nombre réel de particules de

ce type entrant dans la procédure. La contamination est le rapport du nombre de particules mal identifiées à la somme de celles correctement et incorrectement identifiées. L'efficacité et la contamination d'identifications combinées sont nettement meilleures et moins dépendantes de l'impulsion que dans le cas d'une identification des particules par un seul détecteur. L'efficacité du résultat cumulé est toujours supérieure (ou égale) à celle obtenue à partir des détecteurs individuels, et la contamination combinée est toujours inférieure (ou égale). L'approche peut facilement prendre en compte l'information d'identification fournie par le TRD. Cela permet d'améliorer la qualité de l'identification pour tous les types de particules (à l'aide des mesures de dE/dx) et en particulier pour les électrons (en ajoutant le signal de rayonnement de transition).

L'identification des hadrons chargés à haute impulsion dans la région centrale de rapidité est assurée par le système HMPID, qui est constitué d'un ensemble de sept détecteurs à rayonnement Tcherenkov. Une identification combinée des pions, kaons et protons entre HMPID et TOF est possible dans des intervalles d'impulsion partiellement superposés. L'information provenant de ces détecteurs permet d'améliorer l'efficacité de l'identification et de diminuer de façon significative la contamination. Le logiciel pour obtenir le PID combinés avec HMPID et TOF est en cours de développement.

4.7.4 La reconstruction dans les autres détecteurs

Les photons sont détectés par le spectromètre de photons, PHOS, constitué d'un calorimètre électromagnétique pour la mesure des impulsions de photons et un détecteur de veto pour les particules chargées (CPV) pour discriminer particules neutres et chargées (le premier CPV sera installé en 2013). Toutes les particules qui frappent PHOS interagissent avec la matière du calorimètre et produisent des gerbes électromagnétiques en déposant de l'énergie dans les cristaux. Les cristaux avec des signaux sont regroupés en clusters qui permettent une évaluation de l'énergie totale déposée par les particules dans le calorimètre et des coordonnées spatio-temporelles du point d'impact sur la surface du détecteur. Une procédure spéciale est appliquée pour déconvoluer les clusters des gerbes qui se chevauchent. L'identification des particules dans le PHOS est basée sur trois critères [98] : le temps de vol mesuré, la forme des gerbes produites par les différentes particules dans le calorimètre, et, quand le CPV sera installé, la correspondance entre le signal dans le calorimètre et celui dans le détecteur CPV. Le temps de vol entre le moment de l'interaction des faisceaux et la détection dans PHOS est mesuré par l'électronique frontale du détecteur et permet la suppression des particules lentes, en particulier les nucléons à basse énergie ($E < 2$ GeV). La forme de la gerbe produite par des photons et des électrons est différente de celle de la gerbe produite par les hadrons. Cette différence est utilisée pour discriminer les particules qui interagissent avec le calorimètre de façon électromagnétique ou hadronique. La correspondance entre les positions des particules détectées par le calorimètre et les signaux du CPV permettra de distinguer entre particules neutres et chargées.

La tâche principale pour PHOS est la mesure des photons directs, c'est-à-dire ceux qui proviennent directement de l'interaction initiale et qui ne sont pas les résultats de la désintégration d'autres particules. Cette mesure est effectuée pendant l'analyse. Les photons directs sont mesurés comme un excès dans le

spectre des photons de désintégration. Pour mesurer ce spectre on reconstruit le spectre des mésons neutres se désintégrant en photons (π^0 , η , ω , etc.). A leur tour, ces spectres sont évalués statistiquement à partir de la distribution de masse invariante de toutes les combinaisons des photons à bas p_T . Ayant les spectres de tous les mésons neutres, on calcule le spectre de photons de désintégration. Le spectre des photons directs est obtenu en soustrayant la distribution des photons de désintégration ainsi calculée de la distribution totale des photons.

La reconstruction dans le calorimètre électromagnétique (EMCAL) suit une procédure semblable à celle du PHOS, avec la différence qu'ici il n'y a pas de calorimètre de veto pour les particules chargées et que les unités de détection ne sont pas les cristaux, mais plutôt les regroupements des fibres optiques qui sont lues ensemble, et qui prennent le nom de tours (en anglais *towers*). Nous rappelons ici que chaque tour est composée d'un sandwich scintillateur-Pb de 70 couches, chacune traversée par des fibres optiques qui récoltent la lumière produite dans les couches du scintillateur. L'énergie déposée par les particules dans les tours produit une lumière de scintillation qui se propage aux APDs situés à la base des tours. L'impulsion électrique générée par les APDs est échantillonnée, numérisée et stockée dans les données brutes.

Le premier pas de la reconstruction est l'ajustement d'une fonction paramétrée à l'impulsion mesurée, pour extraire l'amplitude du signal et son temps d'arrivée. L'étape suivante est la formation de groupes de cellules (clusters) dont le signal est généré par la même particule. Pour les collisions pp, l'algorithme est une simple agrégation des tours voisines jusqu'à ce qu'il n'y ait aucune tour adjacente au-dessus d'un seuil d'énergie donné. Dans le cas des collisions Plomb-Plomb, où les gerbes des particules différentes se chevauchent souvent, nous joignons une tour à un cluster si elle a plus d'énergie que la précédente. Selon le type d'analyse, on peut utiliser l'une ou l'autre méthode de création de clusters.

Une fois que le cluster est défini, on calcule les paramètres de gerbe qui aideront au niveau d'analyse pour identifier le type de particule. Ensuite, on compare la position du cluster avec les trajectoires détectées dans le « baril » pour identifier les clusters générés par les particules chargées.

Toutes les informations sur les clusters et sur les signaux dans les tours sont écrites dans les ESDs pour permettre de recalculer les clusters, si nécessaire, au niveau de l'analyse.

Il faut mentionner ici qu'il y a trois éléments extrêmement importants pour la reconstruction dans l'EMCAL (et aussi dans PHOS). Pour une reconstruction correcte des clusters et ensuite des jets, il est essentiel que les tours soient calibrées avec la plus grande précision possible. La calibration initiale est effectuée avant d'installer le détecteur. Le degré de précision de la calibration est obtenu avec une procédure itérative basée sur la mesure de la masse du π^0 pour obtenir une précision de 1-2 % en analysant 100~200 millions d'événements. Pour cela, des prises de données dédiées (en anglais *calibration run*) sont organisées. Comme l'EMCAL est le détecteur qui fournit aussi un signal de déclenchement, les tensions des APDs sont ajustées avant chaque prise de données en fonction des résultats de la calibration. Le calibrage doit aussi tenir

compte de la dégradation de la réponse du détecteur avec le temps, qui est constamment contrôlée par les procédures de contrôle de qualité.

L'alignement a aussi une grande importance, et il est constamment contrôlé et mis à jour en comparant les signaux de l'EMCAL avec les trajectoires du « baril » et en mesurant la masse du π^0 , car ces valeurs changent en fonction des variations des positions des tours.

Tous ces facteurs de correction sont stockés dans la OCDB, mais aussi dans la base de données hors-ligne pour l'analyse (OADB en anglais *Offline Analysis Data Base*) afin de pouvoir faire des corrections au niveau de l'analyse.

Le rôle principal du EMCAL dans ALICE est la détection des jets, qui est faite pendant l'analyse avec des bibliothèques spécifiques qui font partie d'AliRoot. Pour cette mesure, ALICE a adopté la même approche que l'expérience STAR, en utilisant la calorimétrie électromagnétique (EMCAL) combinée avec un suivi précis des particules chargées dans un champ magnétique modéré (0,5 T). Cette technique diffère de l'approche la plus commune dans les expériences de collisionneur, dans lequel une mesure purement calorimétrique des jets est faite via deux calorimètres, l'un hadronique et l'autre électromagnétique.

Le choix d'utiliser la trajectographie plutôt que la calorimétrie hadronique a été fait afin d'optimiser la reconstruction des jets dans l'environnement complexe des collisions d'ions lourds, où la caractérisation détaillée de l'événement sous-jacent et de ses fluctuations est nécessaire pour des mesures de jet bien contrôlées. Une telle approche présente des effets systématiques qui ne sont pas présents dans une mesure purement calorimétrique, et qui nécessitent une trajectographie très précise et des corrections spécifiques de l'énergie du jet dans la phase d'analyse. Une preuve de principe de cette approche a été fournie par l'expérience STAR (voir par exemple [100]).

D'autres approches pour la reconstruction des jets dans l'environnement des ions lourds utilisent des coupures cinématiques sur les particules ou sur les tours du calorimètre utilisées dans la reconstruction afin d'éliminer les fluctuations du fond, ou encore des coupures cinématiques imposées par la seule existence d'un fort champ magnétique. Une étude de la reconstruction des jets dans les collisions d'ions lourds par l'expérience STAR montre que ces coupures cinématiques introduisent de forts biais pour les jets étouffés²³ (en anglais *quenched*), et que des mesures non biaisées des jets étouffés nécessitent l'élimination ou la minimisation de ces coupures. Cela peut être accompli avec ALICE avec une reconstruction trajectographique de haute précision et une calorimétrie électromagnétique à haute granularité comme celle du EMCAL, qui impose de très faibles coupures instrumentales sur les particules dans les tours ($\sim 100 - 200$ MeV), contrairement à d'autres expériences du LHC.

4.8 L'analyse

Les procédures d'analyse des données expérimentales sont difficiles à formaliser et à généraliser lorsque les observables s'étendent sur un très grand nombre de

²³ Nous disons qu'un jet est *étouffé* (en anglais *quenched*) quand son énergie est plus ou moins fortement réduite à cause de l'interaction avec le milieu chaud et dense qui résulte de la collision de deux ions lourds.

paramètres. Comparée aux autres expériences au LHC, ALICE a un système de détection particulièrement complexe qui doit prendre en compte plusieurs observables dans le cas des interactions proton-proton, proton-A et A-A. En outre, ALICE est un producteur de données sans précédent dans la Physique des Hautes Énergies, avec des taux d'acquisition de données jusqu'à 4 Go/s et une quantité de stockage de données brutes avoisinant le 3 Po par an.

Un autre facteur important est que, pour ainsi dire, il y a autant d'analyses que de physiciens, et donc nous nous sommes posé la question s'il était raisonnable d'investir dans le développement d'une infrastructure générale d'analyse car, elle risquait d'être ou minimale, et donc presque inutile, ou au contraire trop importante et donc contraignante. L'auteur, après discussion avec son équipe et les physiciens, a retenu l'option d'entreprendre ce développement et il a activement participé à la phase de la conception. Le résultat a été un système commun qui a permis aux physiciens d'effectuer toutes sortes d'analyses et qui a introduit une uniformité dans les formats et dans les interfaces, ce qui rend la communication entre différents groupes et la vérification du code très faciles. Cela a aussi évité une inflation des formats d'analyse, qui est un problème rencontré dans d'autres expériences.

L'infrastructure élaborée par l'équipe Core Offline de ALICE virtualise les modèles d'accès aux données et introduit un langage commun d'analyse pour tous les utilisateurs de l'expérience. Cela doit être considéré comme une situation assez satisfaisante, car dans une collaboration d'environ mille personnes, la coordination des efforts d'analyse des données est obligatoire pour assurer l'accès de tous aux données et la reproductibilité des résultats. L'ensemble des outils mis au point à cet effet constitue ce que nous appelons aujourd'hui l'infrastructure d'analyse de l'expérience et est actuellement utilisé dans le contexte de quasiment toutes les analyses des données.

L'infrastructure d'analyse tient compte de deux dimensions de flexibilité. D'un côté elle satisfait les exigences des dizaines d'analyses différentes qui sont effectuées par les différents groupes de travail de physique et par les physiciens. D'un autre côté elle est aussi utilisable dans des contextes très différents, à savoir:

- L'analyse effectuée par un physicien sur son ordinateur personnel sur un échantillon limité de données stockées localement ;
- L'analyse effectuée sur la Grille par des tâches par lot (en anglais *batch jobs*), soit pour les besoins d'un utilisateur isolé, soit dans le cadre des *trains d'analyse* officiels des groupes de travail de physique. C'est le mode de fonctionnement principal pour accumuler de la statistique et il utilise une fraction importante des ressources de calcul disponibles ;
- L'analyse effectuée de façon interactive et parallèle sur des Fermes d'Analyse (AF, en anglais *Analysis Facility*) que nous décrirons dans la suite, et qui sont disponibles dans plusieurs centres de calcul. Cette analyse est largement utilisée pour le débogage et la mise au point des algorithmes et pour obtenir un retour rapide sur la qualité des données.

En particulier, la Ferme d'Analyse du CERN (CAF, en anglais *CERN Analysis Facility*) est utilisée pour le traitement rapide des données pour la calibration,

l'alignement, la reconstruction et l'analyse. Ce mode de traitement permet d'avoir un premier regard sur les données et d'obtenir un retour très rapide sur la qualité de la prise de données.

Dans un contexte où l'ensemble des données est réparti sur un grand nombre de centres de calcul, l'accès simultané aux données par un grand nombre d'utilisateurs est un exercice exigeant. Pour essayer de réduire la charge sur la structure informatique, et aussi par souci de cohérence des résultats obtenus, les algorithmes d'analyse *officiels* de chaque groupe de travail sont exécutés de façon centralisée. Un autre aspect à considérer est le temps nécessaire à un algorithme d'analyse pour traiter un événement en général comparable au temps nécessaire pour le lire. Cela a pour effet que l'unité de calcul de l'ordinateur est utilisée de façon peu efficace car elle doit attendre que l'événement soit chargé en mémoire avant de pouvoir le traiter²⁴. C'est l'un des objectifs de l'infrastructure d'analyse d'optimiser l'équilibre entre le temps de traitement et le temps d'entrée et sortie des données.

La standardisation des classes et des structures de mémoire qui constituent un événement a une importance fondamentale, et peut-être encore plus importante a été la formalisation des modèles communs d'accès aux données et du flux de l'analyse.

L'effet de l'introduction d'un cadre commun d'analyse est donc de permettre un meilleur usage des ressources existantes, et également de contribuer à développer un cadre commun et robuste et un langage commun pour l'analyse. En effet, du point de vue sémantique, l'infrastructure d'analyse met en œuvre une base de connaissances et une terminologie commune et permet de documenter la procédure d'analyse et de rendre reproductibles les résultats. Par exemple, les premiers mois de l'analyse des données ont vu un développement très actif des critères de sélection des événements. Grâce à l'infrastructure d'analyse ces critères de sélection ont pu être standardisés et mis à la disposition de tous les utilisateurs d'une façon cohérente et sans erreur.

Du point de vue de la fonctionnalité, l'infrastructure d'analyse permet un accès transparent à toutes les ressources de calcul en utilisant le même code utilisateur, qui peut être exécuté dans les environnements Proof et Grid de la même manière que localement. Grâce à l'utilisation d'interfaces virtuelles d'entrée et sortie, l'accès aux données d'entrée est également transparent pour les différents types de données (ESD, AOD ou données MonteCarlo).

Pour minimiser l'entrée et sortie par rapport au temps de calcul, on utilise des « trains d'analyse ». L'idée est qu'une fois les données d'un événement lues, des tâches d'analyse sont exécutées l'une après l'autre, chaque tâche étant considérée comme un « wagon » du train d'analyse. L'infrastructure d'analyse fournit un formalisme pour définir des tâches génériques d'analyse

²⁴ On peut objecter qu'une entrée et sortie asynchrone pourrait réduire le problème. Si cela est vrai en principe, il y a aussi beaucoup d'autres considérations techniques relatives au système d'entrée et sortie qui rendent cette solution moins évidente qu'elle ne l'est en apparence. En particulier le système d'entrée et sortie de Root ne peut pas pour le moment être utilisé en parallèle avec le calcul dans une filière (en anglais *thread*).

interconnectables qui partagent une boucle d'événement unique. L'utilisateur doit seulement mettre en œuvre son algorithme d'analyse dans une classe dérivée de la classe de tâche.

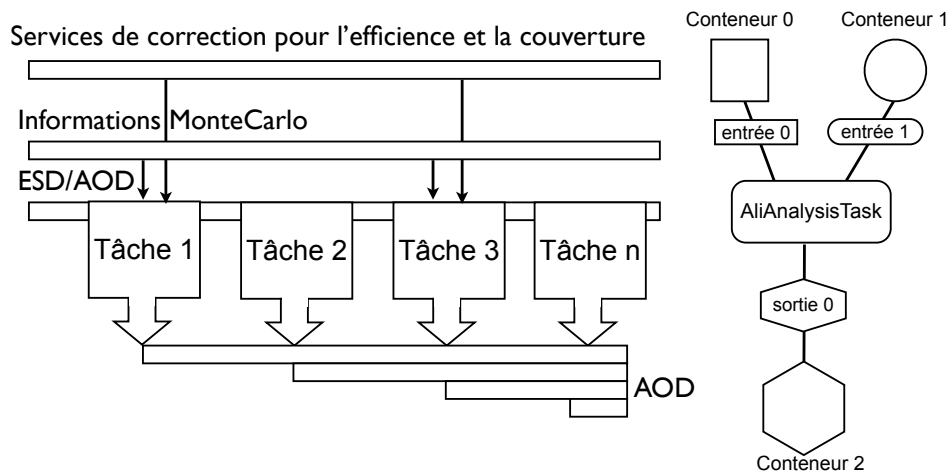


Figure 51 : Train d'analyse construit à partir des tâches.

Comme présenté dans la Figure 51, une tâche peut avoir un nombre arbitraire d'entrées et sorties de différents types connectées à des conteneurs de données. La conception est très générale et n'est pas liée à l'expérience ALICE. Dans ce schéma, les algorithmes dans une tâche ne sont exécutés que lorsque les données réelles sont disponibles dans toutes les entrées. A son tour, la tâche est tenue de publier les résultats dans les sorties définies, en fournissant de cette manière la notification que les données sont prêtes pour d'éventuels clients. Les tâches peuvent être connectées via des conteneurs décrivant un graphe et elles sont exécutées événement par événement par un gestionnaire d'analyse qui gère les dépendances sans que l'utilisateur doive intervenir.

Chaque tâche d'analyse est exécutée en trois étapes:

1. L'initialisation des structures de données de classe (paramètres d'entrée, histogrammes et TTree de Root de sortie). Une partie de cette initialisation peut être faite par la tâche de l'utilisateur lors de sa création sur le client, alors que l'initialisation des objets de sortie doit être faite sur les CPU des esclaves²⁵ (qui peuvent être des machines distinctes des machines clientes)
2. L'exécution de l'algorithme d'analyse. Ceci est fait pour chaque événement en appelant la méthode générique `void Exec()`. Cette méthode doit publier les résultats du calcul dans les sorties déclarées pour cette tâche.
3. Finalisation de l'analyse. Exécutée après l'assemblage de tous les résultats de sortie via une méthode virtuelle, cette étape permet les opérations de

²⁵ Dans une application parallèle hiérarchique, la tâche qui distribue le travail s'appelle « maître » et les tâches qui exécutent le travail en parallèle s'appellent « esclaves ». Ce type d'architecture parallèle d'application s'appelle « maître – esclave » (en anglais *master-worker*).

post-traitements tels que le dessin ou l'ajustement de fonctions aux histogrammes.

Les données d'événements (événements simulés par MonteCarlo, ESD d'événements reconstruits et AODs) dérivent d'une classe de base qui virtualise une interface d'événements commune, ce qui rend possible la manipulation de l'entrée de l'analyse d'une manière transparente. Ces données sont accessibles à partir du catalogue de fichiers de ALICE [101] via le protocole d'accès xRootd (tout cela fait partie du intergiciel de Grille de ALICE dont nous parlerons dans le chapitre prochain). Pour le traitement interactif parallèle sur la CAF, les fichiers de données sont copiés localement sur les disques de la ferme et regroupés en ensembles de données (en anglais *data sets*). Dans les deux cas, le gestionnaire peut exécuter la boucle principale d'analyse sur une série de fichiers d'événements (une chaîne de fichiers dans la terminologie de Root) de façon transparente. Selon le type d'événement, des fichiers supplémentaires peuvent être nécessaires. Pour rendre les opérations d'entrée et sortie totalement transparentes vis-à-vis du type d'événement, le système utilise des gestionnaires génériques d'événements pour l'accès et l'écriture des données, comme indiqué dans la Figure 52.

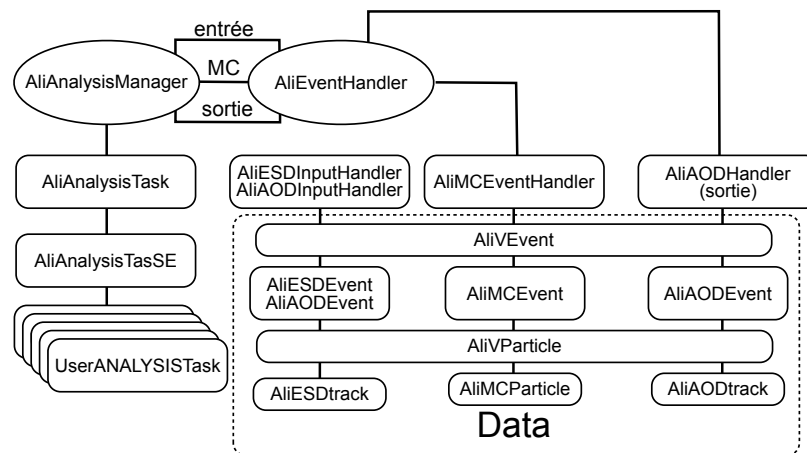


Figure 52 : Vue d'ensemble du flux de données dans l'analyse.

Le système PROOF [102] utilise un simple parallélisme au niveau des événements en divisant les données d'entrée en un nombre de morceaux égal au nombre d'esclaves dans la ferme d'analyse. Le temps d'élaboration est proportionnel au nombre d'esclaves si l'application n'est pas limitée par l'entrée et sortie et si le nombre d'utilisateurs est constant. Dans le cas de la ferme CAF, le temps pour obtenir les résultats dans le cas de l'analyse de 100k événements proton-proton est typiquement de l'ordre de quelques minutes. En plus, PROOF étant un système interactif, on peut récupérer les résultats intermédiaires ou annuler le traitement des données à tout moment.

Le système d'analyse utilise PROOF de la façon suivante : le train des tâches d'analyse est assemblé et initialisé sur la machine de l'utilisateur, le « client ». Le code utilisateur peut être envoyé à la ferme CAF distante de différentes manières – via les bibliothèques d'un AliRoot personnalisé et installé par l'utilisateur, ou via les « paquets » PROOF (des archives en format « tar » regroupant plusieurs

fichiers avec le code source en un seul) qui sont téléchargés et compilés au moment de l'exécution. La deuxième option rend chaque cycle un peu plus lent à démarrer, mais elle est très utile dans la phase de développement des algorithmes d'analyse quand le code doit être changé fréquemment, ce qui peut être fait sans une réinstallation complète des bibliothèques qui prendrait beaucoup plus de temps. L'objet du gestionnaire d'analyse créé sur le client est envoyé au « maître » PROOF qui contrôle la ferme, d'où il est distribué en tant que charge de travail aux esclaves, avec l'information sur les blocs de données à traiter. Une fois sur les esclaves, chaque tâche crée ses propres conteneurs de sortie. Dans le cas où des gestionnaires d'événements de sortie sont utilisés (comme lors de la production des fichiers AOD), le `TTree` de sortie commun est également réservé avant de commencer la boucle d'événements. Après avoir traité tous les événements, les fichiers de sortie sont transmis à un service d'assemblage qui va envoyer le résultat final au client.

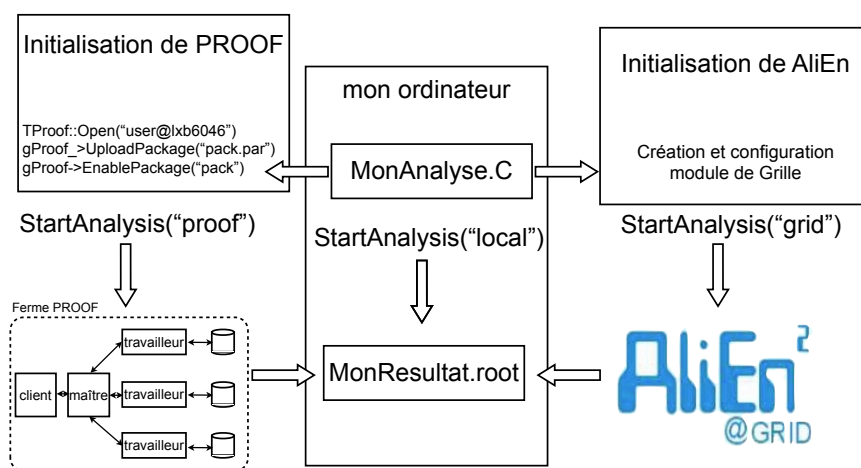


Figure 53 : Aperçu du flux de travail dans l'analyse montrant que le même code peut être utilisé pour des modes d'exécution différents de façon transparente.

L'exécution d'une analyse développée localement sur l'infrastructure de la Grille implique quelques conditions préalables: création des ensembles de données d'entrée à l'aide des procédures AliEn et écriture d'un fichier décrivant le flux du travail dans le langage spécifique de la Grille (JDL, en anglais *Job Description Language*). Ce fichier doit contenir les instructions pour la validation et l'assemblage des résultats, ainsi que pour copier les fichiers depuis et vers le catalogue de fichiers d'AliEn. Une fois que tout cela est préparé, l'analyse peut être exécutée en utilisant le même code que localement.

Un module qui peut être inséré (en anglais *plug-in*) pour travailler sur la Grille a été développé afin d'éliminer (ou du moins simplifier) ce travail préalable. Cela permet à l'utilisateur de travailler à partir de la ligne de commande de Root, en définissant les paramètres d'analyse à travers l'interface de travail de Root (API en anglais *Application Programming Interface*). Ce module automatise toutes les interactions avec la Grille, en réduisant au minimum les informations requises à l'utilisateur : définition des fichiers et des périodes de prise de données requises, versions des logiciels requises, répertoires de travail et de sortie sur la Grille. De

cette façon, comme il est montré dans la Figure 53, on peut exécuter le code d'analyse dans les différents environnements et récupérer les résultats sans avoir besoin de beaucoup de connaissances spécifiques des systèmes AliEn ou PROOF.

Le cadre d'analyse ALICE a été développé et utilisé depuis plus de quatre ans. Tout au long de cette période, il y a eu une augmentation constante de l'utilisation de l'infrastructure et de l'expérience accumulée. Il y a des dizaines de tâches d'analyse dans le format standard qui peuvent être intégrées dans les trains d'analyse ou exécutées de façon isolée. Un train d'analyse officiel est testé régulièrement dans tous les modes de fonctionnement sur des données de test. L'expérience a montré jusqu'ici que l'Installation d'Analyse du CERN (CAF) a été extrêmement utile pour obtenir une compréhension rapide des données par rapport au mode par lots.

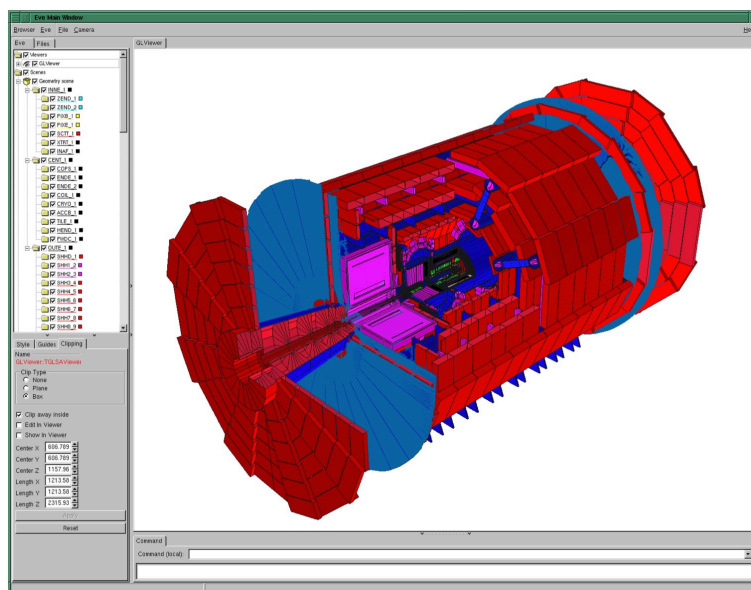


Figure 54 : Géométrie du détecteur ATLAS dans EVE. L'Explorateur d'objets affiche la hiérarchie de nœuds. Une boîte de découpage est utilisée pour montrer la région centrale du détecteur.

La plupart des efforts ont été faits ces derniers temps pour améliorer la stabilité du cadre en mode Grille et pour masquer sa complexité. L'expérience utilisateur a montré que l'infrastructure d'analyse rend assez facile le traitement d'échantillons volumineux de données. Des nombreux tests ont été effectués montrant que le traitement des données via les « trains d'analyse » est beaucoup plus efficace que l'exécution séparée des différentes tâches.

4.9 La visualisation

La visualisation de la géométrie du détecteur et des données des événements joue un rôle important pour un large spectre d'activités d'une collaboration de physique des hautes énergies. L'œil humain a une capacité de saisir synthétiquement les caractéristiques d'un objet complexe, tel qu'un événement, qui constitue un complément indispensable à l'analyse quantitative et statistique des données. Dans ce sens, la visualisation fournit un outil essentiel pour la

conception des détecteurs, le débogage du code de simulation, de reconstruction et de calibration, l'analyse physique et la surveillance en ligne. Cet éventail d'usages implique que l'outil de visualisation doit être capable de gérer tous les types de données d'une l'expérience et de les présenter ensemble dans un environnement unifié. Cette exigence est le principal problème de l'infrastructure de visualisation, absent dans d'autres parties du code hors ligne qui normalement ne fonctionnent que sur une gamme limitée de données d'entrée et sortie. Cela est d'autant plus vrai pour une expérience de la complexité d'ALICE, tant en termes du nombre de sous-détecteurs qu'en termes du nombre de particules à représenter et manipuler dans les collisions d'ions lourds

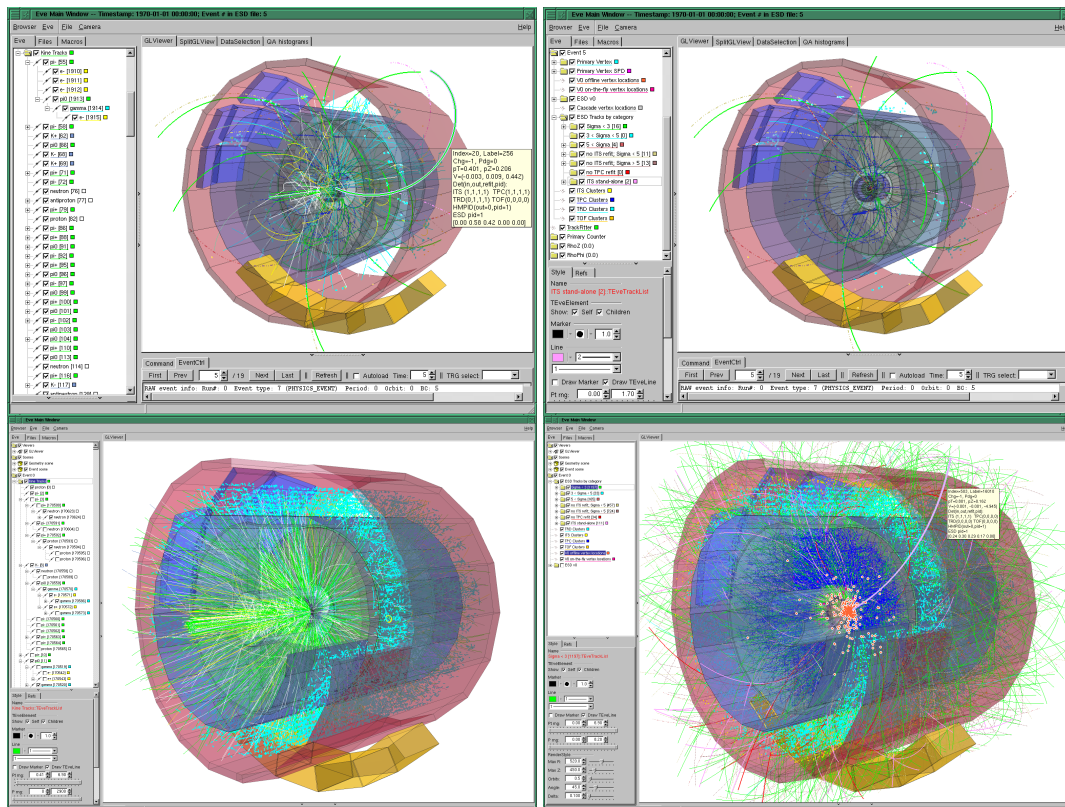


Figure 55 : En haut : collision proton-proton à 14 TeV. En bas : collision Plomb-Plomb a 5,5 TeV par nucléon. A gauche les trajectoires simulées et à droite les trajectoires reconstruites en partant des données simulées.

Une autre complication est due au fait que la visualisation devrait, pour des raisons de modularité, être séparée du cadre général hors ligne, mais en même temps être en mesure de représenter et de « naviguer » tous les objets issus de la physique, des plus complexes comme les jets ou les trajectoires, aux plus élémentaires, comme les clusters, et même les données brutes ou les résultats de la simulation. La première exigence nécessite l'accès à une couche de logiciel la plus mince possible pour la lecture des données, alors que la deuxième implique une intégration complète de l'outil de visualisation dans l'infrastructure logicielle de l'expérience et aussi l'accès interactif aux algorithmes de traitement des données.

Le développement d'un tel outil nécessite un programmeur analyste expert dans plusieurs domaines, de la physique des particules à la programmation graphique, non dépourvu d'un certain gout « esthétique ». Au vu de la situation de l'équipe Core Offline de ALICE et de l'absence ou la non-disponibilité d'une telle personne dans la collaboration, nous décidâmes simplement d'attendre de trouver la bonne personne. Cela fut une stratégie assez risquée car il n'était pas pensable de démarrer l'expérience sans un système de visualisation performant et nous n'avions pas d'idée précise du temps qu'il aurait fallu pour son développement.

La bonne occasion se présenta en 2005 avec deux boursiers, tous les deux experts dans les domaines nécessaires à la réalisation d'un système de visualisation. Leur travail a conduit au système de visualisation de ALICE (AliEVE) [103, 104, 105] et à l'infrastructure de visualisation EVE [106], qui fait partie de ROOT et sur laquelle AliEVE se base pour tous les services qui ne sont pas spécifiques à l'expérience ALICE.

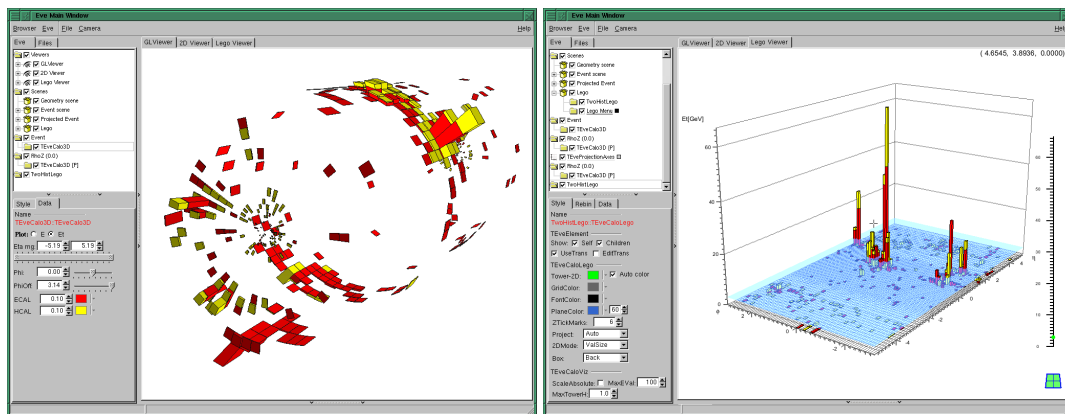


Figure 56 : Représentation des données calorimétriques dans EVE avec TEveCalo3D (à gauche) et TEveCaloLego (à droite).

EVE utilise les modules d'interface graphiques de ROOT pour la couche d'interaction et le système OpenGL [107] pour la représentation des données. L'intégration entre ROOT et OpenGL réalisée par EVE joint la capacité de mise à jour rapide des fenêtres graphiques avec une charge minimale pour le processeur avec la possibilité d'accéder aux objets (dans le sens du C++) affichés à un niveau arbitraire de granularité dans une structure graphique hiérarchique.

Les fonctionnalités de haut niveau fournies par EVE (par exemple la gestion des scènes et de la camera 3D) sont mises en œuvre sous forme de classes, permettant de les manipuler comme objets. Par exemple, pour ajouter un élément dans une scène 3D, il suffit de le glisser et le déposer dans l'objet représentant la scène.

Toutes les classes de visualisation EVE ont des références en amont aux données de l'expérience, grâce à un pointeur (un élément TRef de ROOT) présent dans chaque entité visualisée. De cette façon, les classes de visualisation peuvent être dissociées des classes spécifiques à une expérience, tout en offrant la fonctionnalité de la navigation vers l'objet représenté grâce aux mécanismes de virtualité et polymorphisme.

Le premier prototype basé sur ROOT avec graphique 3D a été mise en œuvre en 2006, basé en partie sur le logiciel GLED [108], dans le cadre d'AliRoot. En 2007 tout le code indépendant d'AliRoot a été déplacé dans le module EVE de ROOT pour être mis à disposition de la communauté d'utilisateurs de ROOT. Depuis, plusieurs expériences en dehors de ALICE (CMS [109], FAIR, NA-62 et T2K) ont adopté EVE comme base de leur système de visualisation.

Du point de vue architectural, les composants d'EVE peuvent être classifiés en trois catégories, en fonction du rôle qu'ils jouent dans le système :

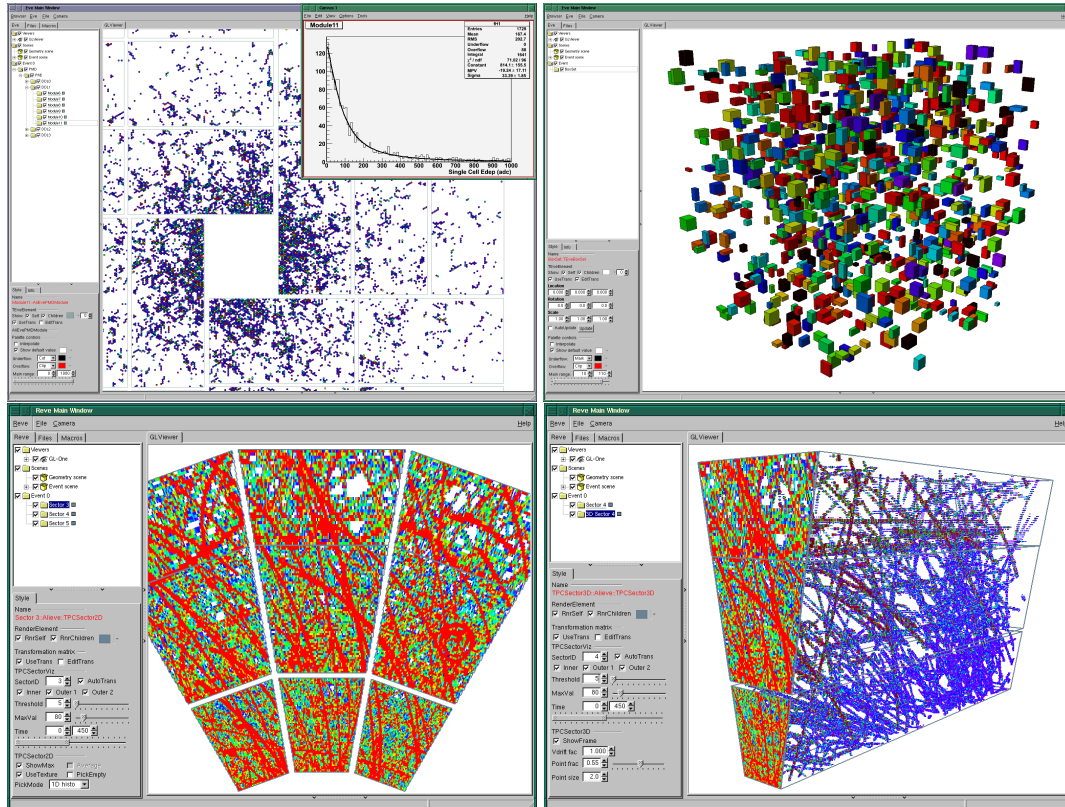


Figure 57 : En haut : exemples de visualisation de données brutes. En bas visualisation de données TPC de ALICE pour des événements Plomb-Plomb. Des sous-classes spécifiques ont été développées pour l'affichage des données en 2D (gauche et droite) et 3D (à droite).

Application de base. Les services généraux exposés aux utilisateurs via une instance de la classe TEveManager et les fonctions statiques de la classe TEveUtil pour la gestion des objets des navigateurs, des scènes 3D et leurs spectateurs et de l'interface graphique des fenêtres de ROOT (onglets, configuration, sélection, navigation).

Classes de base. Ils implémentent la fonctionnalité de bas niveau de la visualisation et des objets graphiques qui sont utilisées par le noyau de l'application pour la gestion des objets et pour la rétroaction à l'utilisateur (nom, titre et couleur de l'objet, mise en évidence d'objets sélectionnés, inspection d'objet via l'interface de la ligne de commande et ainsi de suite).

Classes de visualisation. Les blocs de base pour les tâches de visualisation (visualisation de la géométrie, des points, des trajectoires et de la calorimétrie)

La plupart de ces classes peuvent être automatiquement projetées dans les vues 2D (r - ϕ et p - z). Toutes les classes dans cette catégorie sont équipées d'un éditeur graphique et des interfaces pour le rendu 3D en OpenGL [110].

La classe de base de visualisation (TEveElement) fournit les interfaces entre les objets et le noyau de l'application, le système de rendu 3D et l'interface graphique. Chaque élément rendu peut avoir un nombre arbitraire de « descendants » et il a la liste de ses « ascendants », de sorte que les mises à jour peuvent se propager correctement.

La visualisation des événements implique la représentation de la géométrie du détecteur. EVE offre deux méthodes de visualisation de la géométrie : la première utilise TGeo directement (voir Figure 54) et la seconde présente les volumes faisant appel à des mécanismes de tassellation²⁶ pour une représentation plus approximative de la géométrie (voir Figure 55).

Les dépôts d'énergie et les clusters sont visualisés à l'aide de la classe TEvePointSet, qui gère un tableau de points 3D qui peuvent être rendus avec des couleurs et styles de marqueurs différents. Les trajectoires des particules ont aussi une classe de base pour leur représentation (TEveTrack), qui offre des fonctionnalités d'extrapolation et d'interpolation en champ magnétique en tenant compte des caractéristiques cinématiques. Les particules et leurs trajectoires peuvent être mises dans une structure hiérarchique (nécessaire pour l'affichage de la cinématique) ou combinés pour représenter des objets composites reconstruits comme les V0s ou les résonances, comme nous pouvons voir dans la Figure 55.

La visualisation des calorimètres a une classe dédiée (TEveCaloDataHist) qui représente les données comme histogrammes d'énergie en fonction de η - ϕ . Cette représentation est pratique pour les vues globales de la réponse du calorimètre ainsi que pour les applications d'analyse de physique. Plusieurs classes de visualisation peuvent représenter les données de TEveCaloDataHist comme un ensemble de tours en 3D (TEveCalo3D), une charte « lego » en 2D (TEveCaloLego) ou comme projections des sommations des contenus en η ou ϕ (TEveCalo2D).

La visualisation des données brutes est compliquée par la nécessité d'interpréter les données avant de les visualiser. La valeur de signal est représentée par la couleur dans une palette (TEveRGBAPalette) ou par la dimension de certaines images (TEveFrameBox) autour d'un ensemble de modules de même type. Les données brutes sont représentées par un ensemble de rectangles, de lignes ou d'hexagones (voir Figure 57) en 2D (TEveQuadSet) ou des objets en forme de boîte ou de cônes en 3D de sorte que la variation de la taille s'ajoute aux informations de couleur.

EVE permet aussi des projections 2D et grand-angle, important pour une inspection détaillée de la région du vertex, ainsi que pour une bonne utilisation de la surface de l'écran [111]. Dans EVE, les projections peuvent être effectuées automatiquement pour la géométrie, les points, les trajectoires et les données du calorimètre. Actuellement EVE fournit des projections r - ϕ et p - z , mais l'interface n'est pas limitée et peut facilement prendre en charge de nouvelles projections.

²⁶ Découpe d'une surface en éléments réguliers de base.

AliEVE est l'outil de visualisation standard de l'expérience ALICE. Il est utilisé non seulement pour les tâches standard de visualisation hors ligne, mais aussi pour l'analyse de la physique, la visualisation du fonctionnement du déclencheur de haut niveau et la surveillance en ligne. Pendant la prise des données, un processus échantillonne régulièrement les événements, les reconstruit avec le code hors ligne et les visualise avec AliEVE. Les événements sont projetés sur écrans dans la salle de contrôle et sur plusieurs écrans dans les locaux ALICE du CERN. Au-delà de l'effet visuel, cela fournit une surveillance constante du fonctionnement du logiciel de reconstruction (voir Figure 58).

Tous les efforts sont faits pour maintenir AliEVE aussi petit et aussi simple que possible. Une autre simplification provient de l'utilisation de macros CINT pour l'extraction des données de pilotage du système.

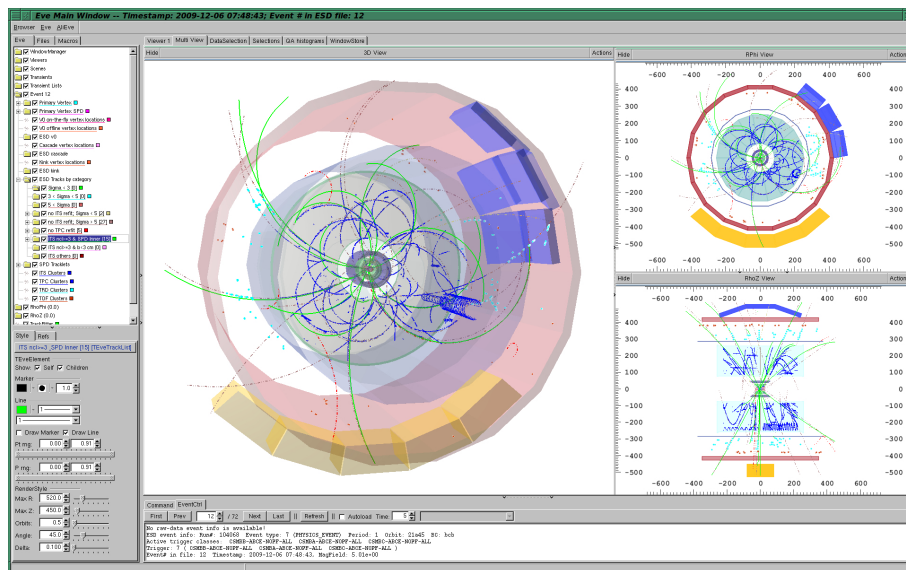


Figure 58 : Visualisation en ligne de la première collision proton-proton à 900 GeV enregistrée par le détecteur ALICE en 2010.

Le choix de faire travailler du personnel payé par ALICE pour développer une partie de ROOT mise à la disposition des autres expériences n'a pas rencontré, au début, un support unanime à l'intérieur de ALICE. Néanmoins ce choix, qui s'était déjà révélé efficace dans le cas du modelleur géométrique, a présenté plusieurs avantages. Le code de EVE, assez important, est maintenant entretenu par l'équipe ROOT, avec une économie considérable de personnel pour ALICE. Le fait que le code soit distribué avec ROOT permet une utilisation encore plus pratique car il n'est pas nécessaire de se préoccuper d'un paquet logiciel supplémentaire qui pourrait entraver le déploiement et la portabilité de AliRoot. L'utilisation de EVE par d'autres expériences a permis de bénéficier d'un débogage bien plus complet du code et de profiter des améliorations de EVE développées dans d'autres collaborations.

Grâce au développement de EVE et AliEVE, ALICE dispose d'un outil de visualisation complet, répondant à la majorité des exigences d'une collaboration HEP. La rapide diffusion de EVE montre que son développement a effectivement fourni une pièce manquante dans le paysage des logiciels de visualisation dans la communauté de la physique des hautes énergies.

5 La Grille informatique de ALICE

5.1 Définition et historique de la Grille du point de vue de la Physique des Hautes Énergies

5.1.1 La naissance de la Grille et de la boîte à outils Globus

En s'appuyant sur le paradigme du méta-calcul [112, 113] et suite à une expérience informatique réussie qui, en 1995, reliait les réseaux de recherche à grand débit afin d'établir un prototype d'une Grille nationale aux États-Unis, la première version de Globus Toolkit [114] (boîte à outils Globus) fut publiée en 1998. Ian Foster et Carl Kesselman publièrent un document [115] décrivant le concept à l'origine de leur « boîte à outils » (en anglais *toolkit*) et ils commencèrent à « évangéliser » l'idée de la Grille comme infrastructure partagée en publiant des articles et un livre très référencé [116] qui a eu un impact profond sur la direction suivie par une grande partie de l'informatique dans le milieu académique les années suivantes.

La base du concept de la Grille repose sur une « vision » assez simple et attrayante. Avec la Toile nous partageons des informations fournies par des serveurs. La localisation et la nature de ces serveurs sont totalement transparentes. La Grille propose quant à elle de partager les ressources de calcul telles que le stockage et les processeurs. Notre ordinateur personnel, qui à présent se compose d'un clavier, d'un écran, d'une unité centrale et d'un système de stockage, pourrait alors se réduire à un écran et un clavier, la Grille fournissant de façon transparente et ubiquitaire la puissance de calcul et le stockage selon nos besoins. Le terme Grille a été introduit par analogie avec le réseau de distribution électrique (en anglais *Power Grid*) auquel on se branche de n'importe où sans se soucier de savoir où et comment l'électricité est produite. De plus, les ressources distribuées par le réseau électrique sont partagées, et chaque utilisateur peut en consommer selon ses besoins.

Dans la dizaine d'années suivant sa publication, le livre de Kesselman et Foster atteint un statut « biblique » dans la communauté grandissante des partisans qui soutenaient avec enthousiasme l'idée d'« une infrastructure matérielle et logicielle fournissant un accès fiable, cohérent, ubiquitaire et bon marché à des ressources de calcul haut de gamme »²⁷, en la considérant comme la solution idéale pour un large éventail de problèmes de calcul intensif scientifiques et sociétaux. Cet enthousiasme a été d'autant plus renforcé qu'il était admis que la technologie pour construire des réseaux de Grille efficaces existait sous la forme de la boîte à outils Globus et que seul un effort marginal d'« assemblage » était nécessaire pour l'adapter aux besoins spécifiques des différents domaines de la science.

Les mots clés définissant les objectifs de la Grille furent continuellement répétés et cités par les membres des communautés (Physique des Hautes Énergies,

²⁷ Définition classique de la Grille selon Kesselman et Foster.

Astronomie) dans les nouvelles demandes de financement pour les projets de recherche et développement, visant à développer une infrastructure logicielle de Grille pour des applications exigeant des hautes performances de calcul tout en promettant d'introduire la technologie de Grille dans d'autres domaines de la science (biologie, médecine, sciences de la Terre), voire dans notre vie quotidienne. Grâce à ces promesses, les budgets pour les projets de Grille affluaient et tout projet dans lequel l'informatique était impliquée sous une forme quelconque devait inclure « un parfum de Grille » pour avoir une chance de financement.

L'idée d'une nouvelle plateforme informatique, ubiquitaire, peu onéreuse et à la disposition de tout le monde était un sujet de recherche attrayant pour les informaticiens, mais aussi pour les chercheurs dans les domaines de la science traditionnellement avides de ressources informatiques, ainsi que pour les organismes de financement qui y voyaient une occasion de fournir massivement des ressources de calcul à un coût raisonnable. C'est la combinaison de tous ces éléments qui a créé le phénomène, nous sommes tentés de dire « la bulle », de la Grille. Surfant sur une vague de financement apparemment illimité et une communauté croissante, les projets de Grille commencèrent à se multiplier à l'aube du nouveau millénaire.

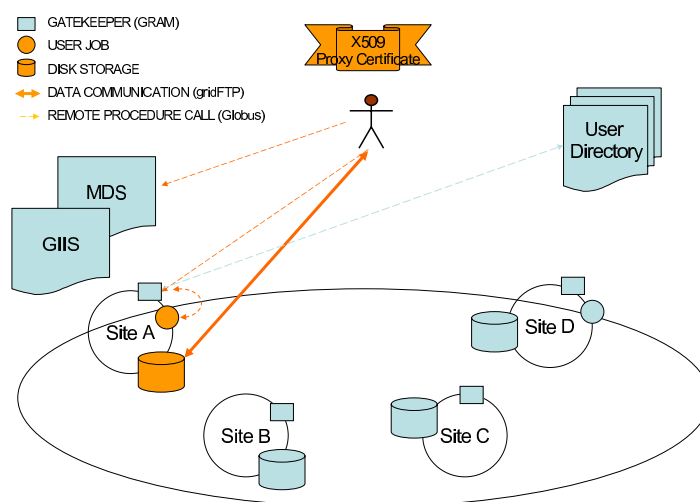


Figure 59 : Schéma simplifié d'une Grille basée sur la boîte à outils Globus.

Au fur et à mesure que la Grille devenait, pour une grande partie des différentes communautés scientifiques, un concept « parapluie » visant à résoudre les problèmes de calcul, une confusion s'installa sur la signification même du terme Grille. Ainsi, presque chaque projet devant résoudre un problème en exécutant de manière distribuée une application particulière pour une communauté d'utilisateurs pouvait prétendre avoir mis en œuvre un système de Grille, sinon « La Grille ».

Afin de clarifier la définition, les auteurs du livre sur la Grille publièrent un article [117] où ils essayèrent de redéfinir la Grille comme une plateforme permettant un accès direct et partagé à « des ordinateurs, des logiciels, des

données et d'autres ressources, selon les besoins d'une gamme de stratégies pour le courtage de ressources et la solution des problèmes qui sont en train d'émerger dans l'industrie, la science et l'ingénierie », mais à la condition que ce partage reste « nécessairement très contrôlé, avec les fournisseurs de ressources et les consommateurs définissant clairement et soigneusement tout ce qui est partagé, qui est autorisé à partager, ainsi que les conditions dans lesquelles le partage se fait ». En outre, ils introduisirent le concept d'« Organisation Virtuelle » (VO, en anglais *Virtual Organisation*) pour désigner « un ensemble d'individus et institutions définis par les règles de ce partage ».

Il s'agissait d'envoyer le message que n'importe quel système informatique distribué ne pouvait prétendre être nécessairement une Grille et que, pour être considéré comme une Grille, l'accès à ses ressources devait être strictement contrôlé. Dans le même temps, la version 2.0 du Globus Toolkit fut promu comme la solution offrant tous les protocoles et composants nécessaires pour la construction d'une Grille et cela de manière plus sûre grâce à l'infrastructure de sécurité Globus (GSI, *Globus Security Infrastructure*) [118].

Pour intégrer des ressources informatiques distribuées et hétérogènes en un « ordinateur virtuel » capable de résoudre un seul problème à un moment donné, une Grille doit disposer d'un intergiciel (en anglais *middleware*). Un intergiciel est un « logiciel interface » entre l'utilisateur ou son application et les ressources de calcul hétérogènes et distribuées. Son rôle est de masquer la complexité, la diversité et la non-localité des ressources et de permettre un accès sûr, transparent et efficace. Pour atteindre cet objectif, l'exécution des applications doit être complètement découplée des composants physiques, c'est-à-dire qu'une application, au lieu d'accéder directement à une composante physique de la Grille, doit en faire la demande par l'intermédiaire de l'intergiciel. Si l'on considère la Grille comme un grand réservoir de ressources, l'intergiciel de Grille doit faire face à une telle complexité, et être extrêmement évolutif, tout en restant sécuritaire. La boîte à outils Globus est constituée d'un ensemble de composants (schématisé dans la Figure 59) destinés à être utilisés pour développer l'intergiciel de Grille :

Infrastructure de sécurité (GSI, en anglais *Grid Security Infrastructure*) : fournit les services d'authentification avec lesquels un utilisateur présente à l'intergiciel son identité, et donc son droit à accéder aux ressources. L'identification se fait sur la base d'un « certificat numérique » crypté qui identifie l'utilisateur comme ayant droit à accéder à la Grille²⁸. Il permet de créer et déléguer des « certificats de procuration » (en anglais *Proxy*) de courte durée sur la base de la clé privée de l'utilisateur, qui peuvent être utilisés pour authentifier l'utilisateur dans le système et éventuellement

²⁸ Un certificat est un bloc de données cryptées qui peut-être vu comme une carte d'identité numérique. Le système de cryptage le plus utilisé est à clé asymétrique. Chaque utilisateur possède une clé privée et une clé publique. Tout message crypté avec la clé privée peut être lu seulement avec la clé publique correspondante, et cela garantit que l'utilisateur en question est à l'origine du message, c'est donc sa signature électronique. Le certificat de Grille est crypté (« signé ») avec la clé privée de l'utilisateur et avec la clé privée d'une Autorité de Certification (en anglais *Certification Authority*), qui est garante des droits de l'utilisateur.

autoriser certaines actions sur la base des informations contenues dans un simple fichier appelé Gridmap.

Gestionnaire Globus d'allocation des ressources (GRAM, en anglais *Globus Resource Allocation Manager*) : transforme des requêtes de services informatiques exprimées dans un langage de spécification de ressource (RSL, en anglais *Resource Specification Language*) compréhensible par les ordinateurs locaux. Ce composant fournit le point d'entrée dans un site de la Grille et il est en charge de la vérification des informations d'identification de l'utilisateur, de leur translation dans le domaine local et de l'abstraction des détails du système local de travail par lots.

Système de surveillance et de découverte (MDS, en anglais *Monitoring and Discovery Service*) : combine des mécanismes de découverte de données avec le protocole léger d'accès de répertoire (LDAP, en anglais *Lightweight Directory Access Protocol*) afin de découvrir et stocker des informations sur l'état du système et de ses composants.

Service d'information des ressources de la Grille (GRIS en anglais *Grid Resource Information System*) : fournit des informations sur les ressources, par exemple, leur configuration, capacité et état.

Annuaire des services d'information de la Grille (GIIS, en anglais *Grid Index Information System*) : coordonne un nombre arbitraire de services GRIS.

Système global d'accès au stockage (GASS, en anglais *Global Access Storage System*) : implémente une variété de stratégies d'accès aux données, permettant aux programmes en cours d'exécution sur des sites distants de lire et écrire des données locales.

GridFTP : fournit un mécanisme de haute performance, sécurisé et robuste pour transfert de données.

Le mérite de la boîte à outils Globus a été d'identifier et codifier les composantes fondamentales de la Grille. Malheureusement, l'implémentation de ces composants n'avait pas la robustesse nécessaire aux communautés plus nombreuses qui ont adopté la Grille en espérant y trouver la solution promise à leurs problèmes. La Physique des Hautes Énergies a été et reste la plus importante en nombre de ces communautés, et elle a donc dû se confronter aux faiblesses du système et y trouver des remèdes.

Le problème fondamental avec le système Globus était la fragilité de ses composants. D'un côté, elle résultait de défauts d'implémentation qui auraient été plus ou moins faciles à corriger. Mais cette fragilité est aussi due à des raisons plus profondes résultant de la nature même de la Grille, et qui se sont avérées beaucoup plus difficiles à maîtriser. La Grille est un environnement très complexe où chacun des composants peut dysfonctionner de façon différente. La construction d'un système « résilient » devient donc très complexe et nécessite plus que la simple combinaison d'éléments modulaires, fussent-ils d'excellente qualité et « résilients » en soi. Il s'agit en effet de bâtir des stratégies où les différents éléments collaborent, et donc le simple assemblage « style lego » des composants de Globus n'est pas aussi simple qu'il le paraît.

De plus, un système de Grille reliant plusieurs centaines d'éléments a souvent un comportement non linéaire. Cela pose de gros problèmes, car il devient

pratiquement impossible de prévoir le comportement à grande échelle d'un prototype qui a démontré bien marcher à petite échelle. Il y a en effet des effets d'« horizon » qui causent des pannes parfois très sérieuses quand un certain seuil est dépassé sans qu'on puisse déterminer d'avance où se situe ce seuil et quels en seront les effets. Il faut d'abord atteindre le seuil pour observer l'effet et y remédier, ce qui pose pas mal des problèmes dans le cas d'une infrastructure « de production » qui est supposée être très disponible.

Nous manquons en effet d'un modèle prédictif du comportement d'une Grille informatique, que ce soit un modèle analytique ou statistique. Cela est assez étonnant, car nous vivons une époque où la simulation prédictive est considérée comme « la troisième voie vers la connaissance », parallèle à la théorie et à l'expérimentation. De plus, la simulation d'une Grille devrait être un problème très attractif, du moins pour une thèse de doctorat dans une faculté de sciences de l'information. Force est de constater que, malgré les efforts de l'auteur pour trouver des étudiants et des professeurs intéressés, rien ou presque n'a été fait dans ce domaine. Cela est d'autant plus étrange que nous disposons d'une « mine d'or » de résultats expérimentaux avec les bases de données des programmes de surveillance, en particulier MonALISA, dont nous parlerons plus loin. Une partie du problème peut venir du fait que, d'après les observations de l'auteur, un certain nombre d'informaticiens considèrent la Grille comme un « problème résolu » du point de vue théorique, et que seul un effort « quantitatif » d'implémentation technique reste à faire, alors que, toujours selon l'auteur, il reste de vastes espaces de compréhension « théorique » à défricher. C'est un autre exemple des améliorations à apporter dans la communication entre physiciens et informaticiens.

5.1.2 Le modèle de calcul MONARC

Dans le budget initial du LHC, approuvé par le Conseil du CERN en 1994 [119], le coût du calcul n'était pas inclus. Ce fait a longtemps été interprété, dans un décryptage « machiavélique », comme une volonté de « cacher » une partie du coût du projet et ainsi faciliter son approbation. Selon l'auteur, la décision de reporter l'estimation des coûts du calcul était raisonnable pour un projet censé entrer en fonction dix ans plus tard (en effet, le LHC ne démarrera qu'en 2008). L'expérience du LEP avait montré qu'une évaluation trop précoce des besoins de calcul [120] pouvait être assez loin des besoins réels lorsque l'on tient compte de l'évolution de l'informatique et de l'évolution tout aussi rapide des demandes des physiciens confrontés à des conditions expérimentales inédites. En 2000, le directeur du calcul du CERN, H. Hoffman, lança une revue dans l'objectif de faire un état des lieux du calcul pour les expériences du LHC et d'évaluer les besoins matériels en termes de puissance de calcul, espace de disque, capacité de réseaux et personnel. Le « rapport Hoffmann » [121] qui s'ensuivit situa l'investissement nécessaire pendant la période 2001-2007 à près de 200 millions de francs suisses. Ce chiffre fit un effet considérable auprès du management du CERN, dont certains représentants accusèrent le comité de vouloir « couler le LHC ». Très surprenante, vu l'importance de la somme, fut la réaction très constructive des organismes de financement dans les pays impliqués dans la construction et l'exploitation du LHC. Ils se déclarèrent prêts à un tel investissement, à condition que les ressources de calcul soient installées localement, dans des centres de

calcul nationaux. En d'autres termes, oui à l'achat et à la mise en œuvre des ressources de calcul, mais à la condition que cette opération puisse générer des emplois et un transfert de connaissances au niveau national. De son côté le CERN, s'il se voyait dispensé de la construction et de la mise en œuvre d'un centre de calcul de dimension déraisonnable, dont il aurait eu à soutenir les coûts infrastructurels, se voyait aussi chargé d'indiquer comment ces ressources hétérogènes et distribuées pourraient être orchestrés pour élaborer de façon efficace les données du LHC.

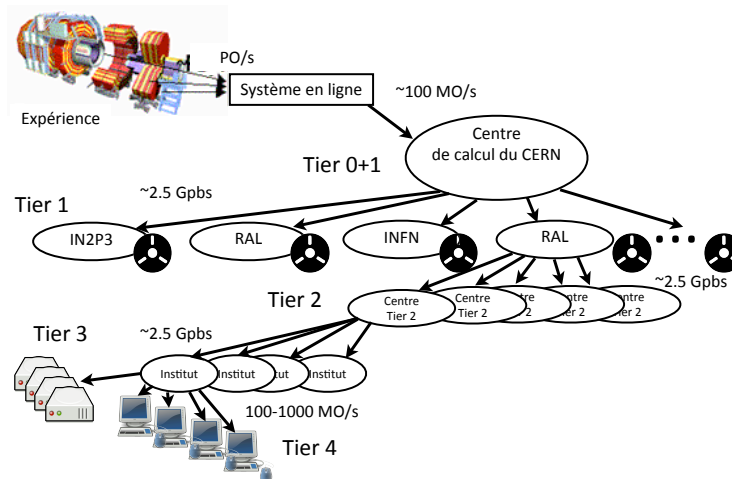


Figure 60 : Modèle de calcul pour LHC basé sur MONARC. Cette figure a été tirée du rapport final du projet MONARC. Les chiffres de la capacité du réseau en 2005 sont les estimations faites en 2000.

Cela représentait un changement de stratégie radical pour le calcul des expériences du CERN. Précédemment, le traitement des données était assuré par le centre de calcul du CERN, et seules quelques productions MonteCarlo et une partie de l'analyse étaient effectuées sur des ressources « locales ». Avec la nouvelle stratégie, il s'agissait de décentraliser également des opérations critiques telles que le calibrage et la reconstruction. Il n'y avait pas d'alternative, car les ressources du CERN ne pouvaient en aucun cas suffire pour le traitement des données si le système « distribué » devait échouer. De plus, un sérieux goulot d'étranglement se présentait dans la capacité des réseaux de recherche, qui offraient au mieux une capacité de quelques Megabits par seconde (Mbps), insuffisante pour les besoins de LHC, avec des perspectives de croissance qui, basées sur l'évolution des réseaux dans les années 90, étaient limitées et coûteuses. Cette préoccupation a profondément influencé la conception du système distribué, mais s'est démontrée finalement infondée. Même les projections les plus optimistes ne pouvaient prévoir l'essor des réseaux au début du millénaire qui nous a amenés à relier les principaux centres de calcul avec un réseau à 10Gbps.

Pour élaborer cette nouvelle stratégie, le CERN lança un projet de Recherche et Développement (R&D) appelé MONARC (en anglais *Models of Networked Analysis at Regional Centers for LHC Experiments*) [122] qui aboutit à un modèle de calcul distribué pour le LHC strictement hiérarchique et centré sur la nécessité de minimiser le transfert de données par le réseau. Cette structure hiérarchique a

comme centre le CERN, qui est la source primaire des données. Comme nous le montrons schématiquement dans la Figure 60, la conception de MONARC était basée sur une hiérarchie de « rangées » (en anglais *Tier*) de centres avec des fonctions spécifiques :

Tier 0. Le centre de calcul du CERN destiné au traitement quasi en-ligne des données récoltées par les détecteurs et leur stockage sur bande magnétique.

Tier 1. Des centres de calcul de grandes dimensions. Ces centres reçoivent les données brutes du CERN pour les stocker sur bande magnétique. Chaque centre reçoit une partie des données de façon à avoir deux copies des données brutes, une au CERN et la deuxième partagée entre les centres Tier 1. Dans la structure MONARC, le Tier 0 et les Tier 1 se partagent la responsabilité de l'archivage à long terme des données produites par les expériences. Ces centres effectuent le retraitement des données brutes et l'analyse dite « ordonnée », c'est-à-dire centralisée. A noter que le centre de calcul du CERN peut aussi opérer comme Tier 1. Il y a actuellement 11 centres Tier 1 sans compter le CERN.

Tier 2. Des centres de plus petite taille qui reçoivent les données reconstruites de leur Tier 1 de référence et qui exécutent des tâches de MonteCarlo et d'analyse. Les événements MonteCarlo produits peuvent être renvoyés à leur Tier 1 de référence pour y être archivés. Il y a actuellement environs 150 centres Tier 2.

Tier 3. Des centres de calcul départementaux des universités, de dimensions encore plus réduites. Ces centres participent à l'analyse.

Tier 4. Les postes de travail des physiciens, qui doivent être connectés à l'infrastructure distribuée pour pouvoir récolter les résultats et soumettre les tâches pour le travail en lots.

Les membres du groupe de travail MONARC se rendirent rapidement compte que si leur modèle était relativement simple et élégant, sa mise en œuvre aurait demandé un énorme développement informatique pour lequel il n'y avait ni un projet technique clair ni les ressources nécessaires à l'intérieur de la communauté de la Physique des Hautes Énergies. C'est là que la Physique des Hautes Énergies rencontra la Grille.

5.1.3 La Grille dans la Physique des Hautes Énergies

Parallèlement au développement de MONARC, le concept de Grille gagnait du terrain dans plusieurs communautés scientifiques. Ce fut un heureux hasard que la boîte à outils Globus sembla pouvoir fournir les éléments d'intergiciel nécessaires pour la mise en place du modèle MONARC. C'est ainsi que pendant la conférence CHEP 2000 (Computing in High Energy Physics) [123], qui se tint à Padoue en Italie, le projet de la Grille de Calcul pour LHC (LCG, en anglais *LHC Computing Grid*, renommé peu de temps après Projet Mondial pour la Grille de Calcul du LHC, WLCG, en anglais *World LHC Computing Grid*), décida d'adopter la Grille et en particulier Globus comme technologie pour le calcul du LHC. Porté

par le succès de la World Wide Web, le CERN décida donc de se lancer en pionnier²⁹ dans le développement d'une Grille de calcul.

Une fois la route tracée du point de vue de l'architecture, MONARC, et du point de vue de la technologie, Globus, il restait à trouver le personnel pour réaliser le projet. La Commission Européenne, préoccupée par un possible retard européen en matière de Grille face au démarrage de plusieurs projets américains dans ce domaine, accepta une proposition faite par le CERN avec d'autres partenaires de financer un projet de développement de Grille pour la Physique des Hautes Énergies, l'observation terrestre via satellite et la biomédecine, à hauteur d'environ 30 millions d'euros sur trois ans.

En 2001, le projet de Grille de Données Européenne (EDG, en anglais *European Data Grid*) [124] fut donc lancé, avec le financement de l'UE et le CERN en tant que partenaire principal.

5.1.4 Le développement de l'intergiciel de Grille de LHC

Le premier problème auquel le projet EDG dut se confronter fut que, en plus des problèmes de performance et de fiabilité, la boîte à outils Globus développée à la suite des expériences de meta-computing et centrée sur le calcul intensif manquait de certaines composantes fondamentales pour la construction d'une Grille destinée à traiter de grandes quantités de données, c'est-à-dire une Grille « centrée sur les données » (en anglais *Data Centric*). Cela avait été compris, et le nom du premier projet pan-Européen de Grille « European Data Grid » exprimait bien ce fait. Pour une telle Grille, il est d'une importance primordiale de coordonner le placement des données et la puissance de calcul afin d'optimiser les performances de traitement des données. Cela a conduit à l'élaboration du concept de « Grille de Données » avec les composants suivants :

Infrastructure de sécurité. Ce composant est équivalent à celui de la boîte à outils Globus.

Élément de calcul (CE, en anglais *Computing Element*). Le CE est un service présent sur chaque site de la Grille et il est le point d'entrée des services pour l'exécution des tâches pour le compte des utilisateurs. Le CE traduit les demandes de ressources dans un format compréhensible pour le système local et il fournit les informations sur l'état d'avancement de l'exécution des tâches assignées au site.

Élément de stockage (SE, en anglais *Storage Element*) Le SE fournit une vue des données disponibles dans le stockage permanent d'un site. Il fournit les services d'allocation du stockage, accès aux données, et transfert des données vers et à partir du site.

Système d'information (IS, en anglais *Information System*). Il permet aux sites d'afficher sur la Grille les services et les ressources disponibles, et fournit des mécanismes permettant aux applications de localiser les sites avec les ressources et les services dont ils ont besoin.

²⁹ Il faut rappeler ici qu'il y avait déjà des projets d'expérimentation de Globus, et notamment en Italie avec l'initiative lancée par l'Institut National pour la Physique Nucléaire (INFN) appelée « INFN Grid » (http://en.wikipedia.org/wiki/INFN_Grid).

Outils de gestion des organisations virtuelles (VO, en anglais *Virtual Organisations*). La VO est une structure qui définit un ensemble d'utilisateurs qui collaborent. Un utilisateur est enregistré auprès de la VO, et il est alors en mesure d'utiliser les ressources et les services des sites qui fournissent des ressources à cette VO selon un ensemble de règles (de l'anglais *policies*) soit locales, soit globales au niveau de la VO.

Ces services sont déjà à un niveau de fonctionnalité plus élevé que les composantes de la boîte à outils Globus, mais ils ne sont pas encore suffisants pour une « Grille de données », et donc une couche supplémentaire de services à été développée. Les services les plus importants de cette couche sont :

Service fiable de transfert de fichiers (FTS, en anglais *File Transfer System*).

Les incidents tels que les pannes des périphériques de stockage, les entretiens prévus ou la surcharge des services de réseau peuvent provoquer des défaillances dans le transfert de fichiers. Dans un service réparti géographiquement, de très nombreux composants peuvent être impliqués, ce qui augmente la probabilité d'échec. Le service FTS a pour but de masquer ces problèmes à l'application. En cas d'échec du transfert, les demandes sont mises en attente et le transfert de données est réessayé selon la priorité de la demande et l'état du réseau.

Services de gestion du stockage (SRM, en anglais *Storage Resource Manager*).

Plusieurs systèmes différents de gestion du stockage sont utilisés par les sites. Par exemple, un Tier-1 doit fournir un système sécurisé d'archivage à long terme des données sur bande magnétique, tandis que pour un Tier-2 il suffit de fournir un espace tampon pour les données sans copie locale de sauvegarde. Les fonctions de SRM fournissent une interface unifiée aux différents systèmes de stockage, et ils sont accessibles depuis l'élément de stockage (SE). Grâce à eux, les applications et les administrateurs système peuvent manipuler le stockage en utilisant les mêmes interfaces sur tous les sites de la Grille.

Service de bases de données distribuées. Celles-ci nécessitent une interaction très étroite entre les sites d'hébergement, ce qui va quelque peu à l'encontre du principe selon lequel la grille est basée sur un faible couplage des services et des ressources.

Catalogue des fichiers et des copies (répliques) (en anglais *File Replica Catalogue*).

Ce service établit une correspondance entre le nom d'un fichier dans l'espace global de la Grille (nom logique du fichier, en anglais *Logical File Name*) et les noms réels (PFN, en anglais *Physical File Name*) des copies des fichiers (répliques) qui sont présents dans les éléments de stockage de la Grille. Le catalogue des fichiers peut être centralisé ou distribué et dans ce cas il est associé aux systèmes de stockage (SE) locaux. Le catalogue des fichiers et des répliques peut contenir d'autres informations sur chaque fichier logique, telles que les droits d'accès, la période de prise de données correspondant à ce fichier, la date du dernier accès et ainsi de suite. Ces données relatives aux données sont appelées « métadonnées » (en anglais *Metadata*).

Système de gestion de la charge de travail (WMS, en anglais *Workload Management System*). Il fournit des services de courtage de ressources entre les sites et les utilisateurs, en optimisant la correspondance des demandes des paquets de tâches (en anglais *jobs*) soumis par les utilisateurs et les ressources disponibles, conformément aux politiques établies par les sites et la VO. Selon leur fonctionnement, nous pouvons distinguer deux classes de planificateurs de ressources. Toutes deux ont une liste de tâches à exécuter soumises par les utilisateurs. Dans le cas du planificateur de type « poussée » (en anglais *push*), le système possède une vision globale des ressources de la Grille et éventuellement de la localisation de chaque fichier de données et il envoie (« pousse ») chaque tâche à un site spécifique avec le but de maximiser l'efficacité globale de la Grille. Dans le cas du planificateur « traction » (en anglais *pull*), le service se contente d'envoyer des tâches génériques appelées « pilotes » (en anglais *pilot jobs*) selon la capacité de chaque site et le nombre de tâches dans la queue centrale. Une fois qu'une tâche pilote démarre sur un site, elle consulte la liste centrale des tâches et « tire » la tâche qu'elle juge la plus appropriée au site local, sur la base de l'état du site et des fichiers disponibles à proximité. Le projet EDG a retenu la première architecture pour le développement du gestionnaire de ressources global

Le projet EDG définit donc une série de sous-projets pour développer ces services à partir des éléments de la boîte à outil Globus.

Après trois ans de travail, l'intergiciel livré fut jugé par les expériences LHC insuffisamment fiable pour satisfaire les besoins dans les cas d'utilisation les plus difficiles. Seule une partie des services fut actuellement développée et encore sur une seule plateforme (Linux). Une analyse approfondie des raisons de ce résultat est au-delà des objectifs de cette thèse. Il sera suffisant de dire que la confiance absolue dans la boîte à outils Globus, la communication difficile entre informaticiens et physiciens utilisateurs et la sous-évaluation de la difficulté pour réaliser une Grille de données de grandes dimensions ont été les facteurs déterminants.

Une autre difficulté venait du fait que la communauté des utilisateurs et des développeurs était dispersée dès le début du projet. Pour augmenter ses chances d'obtenir des financements publics, chaque projet visant à travailler sur la technologie de Grille devait déclarer que son but était de réaliser « La Grille », et non pas de travailler sur un seul aspect de la Grille. Donc, au début du millénaire, plusieurs projets de Grille indépendants furent lancés [125] des deux côtés de l'Atlantique avec les mêmes objectifs, trop ambitieux, et avec la même composante de base, Globus, qui montrait des carences de plus en plus évidentes. La communication entre ces projets fut minimale. Pour ne rien arranger, les projets « américains » avaient tendance à être spécifiques d'une expérience, tandis que les projets « européens » s'appuyaient sur une base nationale ou régionale, le seul projet pan-européen étant EDG. Un autre projet « supranational », mais indépendant de EDG, fut lancé par les pays nordiques (Suède, Norvège, Finlande et Danemark). Les expériences, quant à elles, s'inquiétèrent du manque de maturité de l'intergiciel produit par les différents projets, en particulier EDG, et de la fragmentation de « la Grille » en plusieurs

Grilles indépendantes qui ne pouvaient pas « inter-opérer » bien que toutes basées sur Globus. Donc, ajoutant à la confusion, les expériences se mirent à développer leur propre solution « de secours », plus ou moins basée sur Globus et sur les parties jugées fiables de l'intergiciel produit par les différents projets (AliEn [126], Panda [127], Dirac [128]). Cela était en complète contradiction avec l'idée originale de MONARC et le concept même de Grille et eut l'effet de fragmenter davantage la communauté en réduisant les ressources des projets communs et donc leurs chances de succès³⁰.

Face à cette situation, l'auteur, qui était le coordinateur du sous-projet des utilisateurs pilotes dans le projet EDG, fit aux expériences LHC la proposition [129] d'unir leurs forces pour définir une couche d'« intergiciel supérieur » (en anglais *Upper Middleware*) commune fournissant la fonctionnalité manquante. A cette fin, il fut mandaté par le projet WLCG pour éditer un document listant les requêtes communes aux quatre expériences LHC. Le document fut produit [130], mais la situation « politique » était telle qu'il fut pratiquement ignoré.

Entre temps, l'industrie informatique n'était pas restée insensible à l'énorme intérêt généré par le concept de Grille, et pendant quelque temps les nouveaux produits acquirent le terme « Grid », ou au moins la lettre « g » dans leur nom³¹. A noter que presque aucun de ces produits n'était, à proprement parler, un logiciel de Grille, mais tout au plus un logiciel destiné à une ferme d'ordinateurs locaux.

De son côté, le projet Globus annonçait en 2002 un effort de standardisation avec une nouvelle conception des services de base de la Grille appelée Infrastructure Ouverte des Services de Grille (OGSI, en anglais *Open Grid Service Infrastructure*). Cela devait fournir le services de base pour l'Architecture Ouverte des Services de Grille (OGSA, en anglais *Open Grid Service Architecture*) introduite avec la nouvelle version 3.0 de la boîte à outils Globus.

La nouvelle de l'introduction de OGSI-OGSA eut l'effet d'amener encore plus d'instabilité et de confusion pour le projet de Grille du LHC. D'autant plus que, sous l'influence de ses nouveaux partenaires industriels, l'Alliance Globus annonçait en 2004 l'abandon des travaux sur OGSI et la réalisation des services OGSA à partir de la norme industrielle de l'infrastructure des ressources des services de la Toile (en anglais *Web Services Resource Framework*), dans la version 4.0 de la boîte à outils Globus.

Le projet WLCG et les expériences étaient de plus en plus pressés de pouvoir mener des tests en « grandeur nature », appelés « *Data Challenges* », car le

³⁰ Il faudrait peut-être ajouter qu'une partie du personnel financé par les projets de Grille fut utilisé par les expériences pour développer leur logiciel et pour faire tourner leurs tâches sur la Grille plutôt que pour développer le logiciel de Grille, surtout dans le cas des projets américains qui étaient spécifiques d'une seule expérience. Les raisons en sont à chercher dans la ligne de démarcation tenue à l'intérieur d'une même expérience entre tester la Grille, développer des applications pour la Grille et développer l'intergiciel et dans le manque chronique de personnel dans les expériences.

³¹ Condor-G, Oracle 10g, Sun Grid Engine, Platform Grid Enterprise Orchestrator, Xgrid etc.

système global était beaucoup plus complexe que n'importe quel système distribué réalisé jusque-là, et de plus il était indispensable pour l'analyse des données du LHC. Un échec de la Grille aurait été un coup sérieux à l'image du CERN et de la Physique des Hautes Énergies.

Pour corriger les problèmes de l'intergiciel de EDG et pour continuer à soutenir le développement d'une infrastructure de Grille en Europe, l'Union Européenne accepta la proposition de financer la continuation de EDG sous la forme de projets triennaux EGEE (en anglais *Enabling Grids for E-science in Europe*) I, II et III [131], le dernier se terminant en avril 2010, suivi des projets EMI et EGI [132].

En même temps, il y eut une relative consolidation des projets de Grille, ne laissant plus que trois acteurs majeurs : EDG-EGEE en Europe, OSF [133] (en anglais *Open Science Foundation*) en Amérique et NorduGrid-ARC dans les pays scandinaves. Malheureusement, l'interopérabilité de ces projets était minimale, voire nulle, chaque Grille ne pouvant pas partager ses ressources avec les autres.

5.1.5 Le développement de AliEn

Face à cette situation, l'équipe Core Offline dirigée par l'auteur se trouvait dans une situation de risque technologique grave, car il fallait prendre rapidement une décision sur la technologie de Grille à adopter dans une situation assez confuse. N'importe quelle alternative impliquait de lourds investissements de personnel, et un mauvais choix pouvait avoir des conséquences très dommageables sur le développement du projet. La comparaison entre les premières évaluations de la puissance de calcul nécessaire à l'expérience ALICE et les estimations de l'offre des différents organismes de financement révélait que seule une utilisation extrêmement efficace des ressources de calcul permettrait à ALICE de réaliser son programme de physique.

En effet, ALICE se trouve dans une situation particulièrement difficile en ce qui concerne ses ressources de calcul. Les besoins sont comparables à ceux des autres expériences « majeures » au LHC (ATLAS et CMS), mais le nombre de ses collaborateurs n'est qu'approximativement la moitié. Ainsi, le calcul de ALICE est deux fois plus coûteux par participant que celui des autres expériences. Les organismes de financement dans les différents pays ont choisi de distribuer les ressources de calcul en tenant compte du nombre de physiciens nationaux dans chaque collaboration, et donc ALICE a toujours eu du mal à satisfaire ses besoins.

Au vu de cette situation, dès 2001 il a donc fallu démarrer le plus rapidement possible le développement d'un système de calcul distribué pour fédérer toutes les ressources disponibles et les utiliser efficacement. Les choses étaient compliquées par le fait que, même avec un démarrage prévu du LHC encore lointain, la rédaction des rapports techniques des différents sous-détecteurs exigeait une disponibilité immédiate d'une grande quantité de ressources de calcul pour la simulation et la reconstruction. Le groupe Core Offline devait alors à la fois fournir immédiatement une façon d'utiliser des ressources de calcul distribuées et développer une Grille extrêmement efficace disponible le moment venu pour le traitement des données réelles, et le tout dans le contexte difficile que nous avons décrit dans le chapitre précédent.

Après une brève mais intense période de discussion au début de 2001, il fut décidé de développer un système de Grille complet capable d'inter-opérer avec

les autres systèmes de Grille et disposant d'une interface utilisateurs stable anticipant les développements futurs du monde de la Grille. C'était là un projet très ambitieux et comme tel il fut sérieusement critiqué. Le groupe Core Offline de ALICE fut accusé de dupliquer le travail en cours dans les « projets communs » et pire, de vouloir faire avec deux ou trois professionnels le même travail que des dizaines d'informaticiens peinaient à réaliser dans les différents projets de Grille. Il y eut cependant aussi du vrai dans ces critiques. La nécessité de répondre aux besoins immédiats de l'expérience décida l'équipe Core Offline à se lancer dans un tel projet.

La première étape logique fut de créer une abstraction de chaque composant de l'intergiciel de Grille et d'en fournir une implémentation. Cela était similaire à ce qui se faisait dans les projets « officiels », mais au lieu de développer chaque composant, il fut décidé de l'« assembler » à partir d'un ou plusieurs paquets « Open Source ». Le projet démarra au printemps 2001 et il fut nommé AliEn³² (**A**lice **E**nvironment, c'est-à-dire l'environnement pour le calcul distribué de l'expérience ALICE). En automne 2001, le système était déjà capable de distribuer le travail de simulation sur une dizaine de centres de calcul et d'offrir une vue unique des fichiers générés sur les différents éléments de stockage. Le « miracle » devait beaucoup au fait que 95 % du code venait de paquets Open Source et seulement 5 % avait été développé par l'équipe Core Offline, rédigé en langage Perl et en utilisant SOAP [134] (protocole simple d'accès aux objets, en anglais *Simple Object Access Protocol*) comme protocole de communication. Le code développé fournissait la « colle » pour faire fonctionner ensemble les différents services et il définissait l'« architecture » du système. De ce point de vue, il constituait un développement à très haute valeur ajoutée selon un modèle radicalement différent de celui suivi par les autres projets de Grille. AliEn fut conçu et réalisé dès le départ dans le but d'être utilisé par n'importe quelle organisation virtuelle ayant besoin des ressources de calcul de Grille, et pas seulement pour la collaboration ALICE.

La Figure 61 nous offre un aperçu des services de la première version de AliEn. L'idée était de fournir un point d'entrée unique à l'utilisateur et de structurer les services de façon pyramidale en partant des services les plus simples jusqu'aux plus complexes. Les avantages de cette méthode de développement étaient, et demeurent, multiples. En premier lieu, la petite quantité du code développé pour « coller » les composants de AliEn permet de soigner sa qualité et sa fiabilité et de se concentrer sur les aspects architecturaux. La qualité des paquets « Open Source » utilisés varie, mais ils sont utilisés par une communauté beaucoup plus large que la Physique des Hautes Énergies, ce qui garantit en général une bonne qualité, certainement supérieure à celle du code développé expressément pour un projet de Grille et utilisé seulement à l'intérieur de ce projet. De plus, la maintenance de ce logiciel est assurée par ses auteurs et non par le projet de Grille. Le fait que la partie du code développé par le projet est limitée, permet de prototyper rapidement différentes architectures et idées. La conception hautement modulaire et « abstraite » des services permet aussi de remplacer sans trop d'efforts un ou plusieurs paquets, identifiés comme problématiques, par des paquets équivalents disponibles en « Open Source ».

³² Avec un évident clin d'œil pour souligner la diversité et nouveauté de cette approche.

Pour limiter la charge des centres de calcul et aussi pour nous affranchir le plus possible des spécificités des différents centres, nous avons développé le concept de VOBox (boîte de l'organisation virtuelle, en anglais *Virtual Organisation Box*), c'est-à-dire un ordinateur qui contient tous les services de AliEn et qui, une fois connecté au site, est capable de relier le site à la Grille de ALICE. Le concept de VOBox fut vivement critiqué par les centres de calcul, qui faisaient objection à l'idée de devoir dédier une machine spécialement configurée pour chaque organisation virtuelle. Néanmoins, le concept se révélait très efficace et aujourd'hui toutes les expériences de LHC utilisent leur VOBox. L'intergiciel officiel de WLCG comprend à présent une VOBox générique avec des services de base qui sont complétés par les services spécifiques de chaque organisation virtuelle.

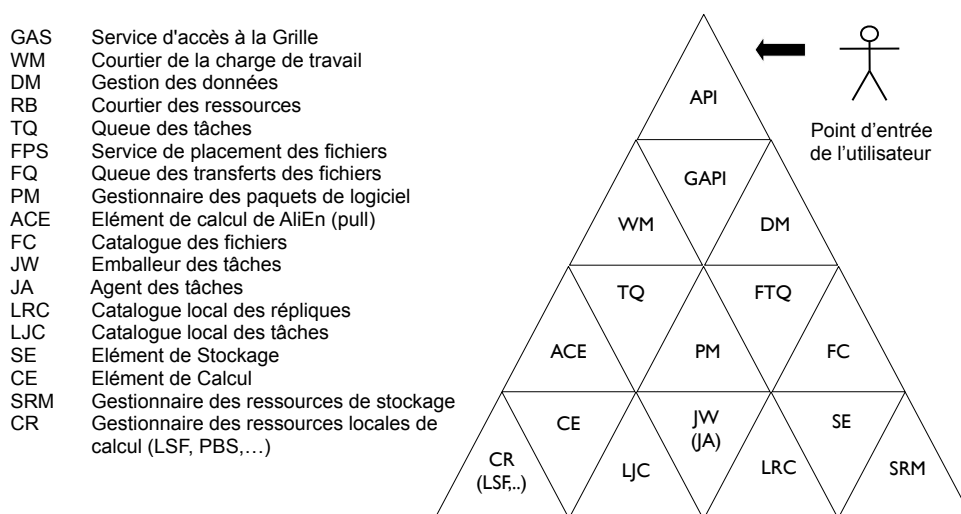


Figure 61 : Aperçu des services de AliEn.

Un autre point de la plus haute importance est la possibilité d'installer le logiciel ainsi développé sur différentes architectures et systèmes d'exploitation. Les paquets « Open Source » sont en général assez faciles à porter sur divers systèmes, et le code de AliEn grâce à sa taille réduite et au fait qu'il est écrit en Perl, peut facilement fonctionner sur différentes architectures.

Tout cela a permis à l'expérience ALICE de se doter d'une plateforme de Grille assez avancée et à « maintenance limitée ». Nous pouvons voir cette approche comme une réinterprétation des modèles de développement LCP dont nous avons parlé dans la Section 3.1.

Il faut remarquer ici que cette approche contenait un élément « hérétique » dans son pragmatisme, car il utilisait Globus avec les services de la Toile, et il n'était pas basé sur la « doxa » qui exigeait une stricte adhésion au modèle OGSi et OGSA pour tout système de Grille.

Bien que AliEn fournisse à ALICE une plateforme d'intergiciel efficace et évolutive qui répondait à des besoins immédiats, le problème se posait de la relation entre cet intergiciel et les intergiciels développés par les projets « officiels », qui, bien que relativement en retard par rapport à AliEn, étaient destinés à être installés sur les centres de calcul de WLCG, vu l'énorme

investissement déployé par la communauté pour leur développement. Ici aussi la modularité de l'architecture de AliEn était un atout. L'idée était que, dès qu'un des services des projets « officiels » devenait disponible et fiable, il pouvait remplacer un ou plusieurs paquets « Open Source » qui faisaient partie de la « pyramide » de AliEn (voir Figure 61). AliEn devait ainsi se « réduire » avec le temps de façon à passer progressivement la charge du support de l'intergiciel de l'équipe Core Offline aux projets de Grille. Et mieux encore, nous espérions voir AliEn se « déplacer » vers le haut, dans le sens de pouvoir continuer un développement de services à très haute valeur ajoutée au fur et à mesure que la fonctionnalité des services « officiels » se développait. Malheureusement, cette évolution ne se produisit pas et, si la fonctionnalité de AliEn est en évolution continue, la taille du code l'est aussi, avec une augmentation de la charge pour l'équipe Core Offline.

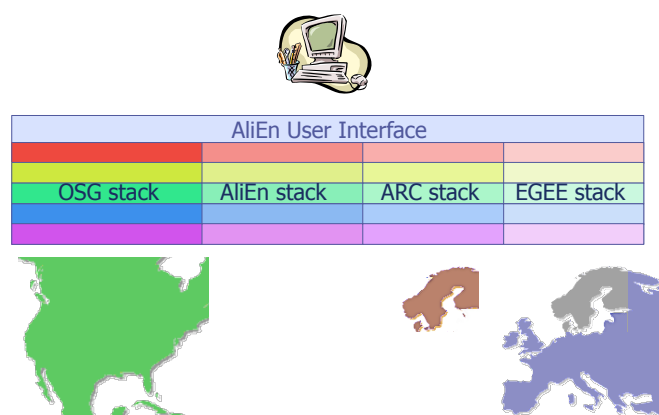


Figure 62 : AliEn vu comme Grille superposée.

Cette option architecturale nous a permis d'expérimenter très tôt l'interopérabilité entre différentes Grilles. En effet AliEn pouvait être utilisé comme un système qui envoie des tâches à d'autres Grilles, réalisant une « Grille des Grilles ». Ce système pouvait être étendu à une structure « auto-similaire » de Grille de profondeur arbitraire [135]. Cette option a effectivement été testée, mais elle n'a pas été retenue, car elle impliquait une interface « lourde » pour passer l'information entre les différentes Grilles et AliEn. Une deuxième option qui se révéla beaucoup plus féconde fut l'idée de Grille « superposée ». L'idée est de substituer aux services de AliEn les services propres à chaque Grille dès que possible. L'utilisateur de AliEn ne voit qu'une seule Grille, mais des Grilles différentes, voire incompatibles peuvent être exploitées comme une seule ressource de calcul distribuée et homogène. Cela est illustré de façon symbolique dans la Figure 62, tirée d'une présentation de l'auteur datant de 2002.

Cette dernière solution a été retenue et est utilisée encore aujourd'hui pour la Grille de ALICE. Les utilisateurs ne voient qu'une seule Grille qui fédère toutes les ressources disponibles à la Collaboration, bien qu'elles appartiennent à des Grilles indépendantes et incompatibles.

Le logiciel officiel développé par les projets européens de Grille était considéré par les expériences comme insuffisamment fiable, trop instable et avec trop peu de fonctionnalités. Les développeurs, de leur côté, étaient dubitatifs devant les changements de direction de Globus et les demandes de collaborations en

évolution constante au fur et à mesure que les expériences prenaient conscience de leurs besoins. Pour atténuer la confusion et proposer des solutions remettant le projet EGEE sur la bonne voie, le projet WLCG nomma un groupe d'évaluation des exigences techniques (RTAG, en anglais *Requirement Technical Assessment Group*) chargé d'élaborer un plan directeur pour les composants de la Grille nécessaires au traitement efficace des tâches d'analyse, considéré comme la tâche la plus difficile à réaliser sur la Grille. Le rapport final de ce groupe [136], composé d'experts des quatre expériences, recommandait l'adoption d'une architecture basée sur le système AliEn, ce qui constituait une approche radicalement différente par rapport au projet EDG, entièrement basé sur la boîte à outils Globus.

Le rapport créa un malaise considérable des deux côtés de l'Atlantique, car l'architecture du système AliEn était considérée comme hérétique par plus d'un aspect. AliEn utilisait seulement quelques composants de la boîte à outils Globus, (GSI et GridFTP) et encore de façon optionnelle. Le fait même qu'une poignée de développeurs ait pu, avec cinquante mille lignes d'un code développé en deux ans, offrir quasiment toutes les fonctionnalités attendues de la Grille, constituait implicitement une critique des grands projets de Grille qui peinaient à livrer des produits fiables, et cela ne fut guère apprécié par les membres de ces projets.

Cependant, le rapport ARDA ne pouvait pas être facilement ignoré, et le projet EGEE, démarré en 2003, développa un prototype d'intergiciel basé sur AliEn (appelé gLite) qui fut présenté à la conférence du projet EGEE à la fin de l'année 2004 [137]. Ce succès technique fut cependant obtenu au prix d'une fracture dans la communauté de la Grille pour LHC. Peu après la conférence EGEE, sous la pression de certains partenaires d'EGEE qui estimaient que la poursuite du prototype constituait un risque, l'activité de développement fut brusquement interrompue et la priorité fut donnée à la restructuration des composants du projet précédent qui avaient été identifiés comme essentiels pour satisfaire les demandes des expériences.

Selon l'auteur, cette décision fut largement guidée par des aspects « politiques » visant à ne pas désavouer les projets précédents de Grille, qui avaient fait l'objet d'une considérable exposition médiatique. En outre, plusieurs développeurs refusaient d'abandonner Globus comme fondation de l'intergiciel de Grille, au point de prétendre qu'un système distribué qui ne serait pas basé sur Globus ne pouvait pas être appelé Grille.

Cette décision fut une occasion manquée de faire de AliEn la base de la Grille de LHC, et le poids du développement et de la maintenance de AliEn est resté avec la collaboration ALICE, avec l'aide du département IT du CERN.

AliEn a été continuellement développé de 2006 à aujourd'hui et de nouvelles versions sont déployées avec une fréquence annuelle. La Figure 63 donne un aperçu de l'évolution de AliEn. Bien que le système AliEn offre une gamme de fonctionnalités très étendue par rapport au système initialement déployé en 2001, l'architecture générale du système est restée très proche de la conception originelle. Cela montre que cette conception est suffisamment flexible et robuste pour s'accommoder de modifications importantes sans changer les principes de base.

Une étape fondamentale dans cette évolution a été franchie lors de l'introduction du système de contrôle MonALISA [138] en 2006. Grâce à MonALISA, nous contrôlons par le détail le fonctionnement de notre Grille. MonALISA met à notre disposition une grande quantité d'informations qui peuvent être utilisées pour l'optimisation du fonctionnement de la Grille. Nous décrivons une application du contrôle pour l'optimisation du traitement des données dans le prochain paragraphe.

Un autre jalon dans l'évolution du système de calcul distribué de ALICE a été la décision prise en 2005 d'adopter xRootd [139] comme seul protocole d'entrée et sortie pour les données. Cela a permis d'uniformiser l'accès aux données avec une simplification substantielle du code et des procédures.

5.2 Structure de la Grille de ALICE

Après avoir évoqué l'histoire du système AliEn, nous décrivons par la suite l'architecture actuelle de la Grille de ALICE. Malgré le refus de faire de AliEn la base de l'intergiciel du LHC, AliEn a introduit plusieurs éléments architecturaux qui ont été successivement adoptés par les autres expériences, qui les ont adaptés à leur intergiciel.

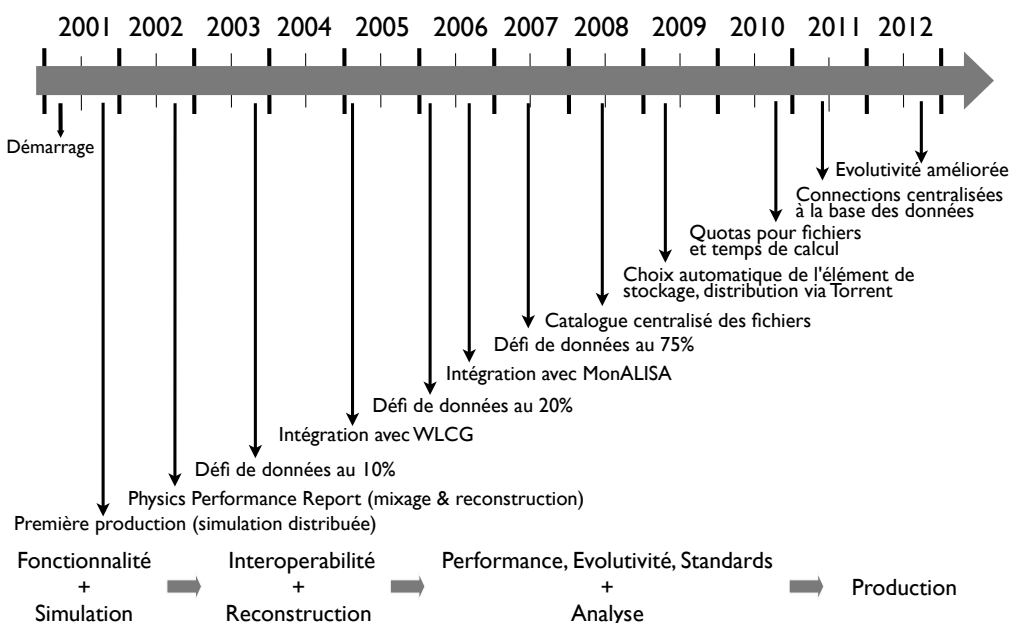


Figure 63 : Aperçu de l'évolution de AliEn depuis le début du projet jusqu'à aujourd'hui.

AliEn est un système intégré pour la gestion des fichiers et des tâches dans un système de calcul global et distribué. La structure de AliEn est modulaire et il est possible de remplacer les services de AliEn par des services d'autres Grilles grâce à des interfaces virtuelles ou à des « adaptateurs ». AliEn permet l'exécution des tâches par lots et l'accès aux fichiers par lots ou de façon interactive.

Le noyau de AliEn est constitué par deux services qui interagissent, le catalogue des fichiers et le système de gestion de la charge de travail. Autour de ces deux services, d'autres services collaborent avec eux. Nous donnons ici une description de chacun de ces deux services majeurs.

5.2.1 Le catalogue de fichiers de AliEn

Le catalogue des fichiers contient l'information relative à tous les fichiers qui sont disponibles dans le système distribué de calcul de ALICE. Pour chaque fichier il existe une ou plusieurs copies (« répliques ») sur des éléments de stockage de la Grille de ALICE. Chacune de ces copies a un « nom physique » ou PFN (voir section 5.1.4) sur l'élément de stockage sur lequel elle réside. Chaque fichier a un identifiant unique appelé GUID (identifiant global unique, en anglais *Globally Unique Identifier*). Le catalogue de fichiers de AliEn est l'espace « sémantique » pour tous les fichiers de ALICE, qui sont identifiés par leur « nom logique » ou LFN. Le catalogue lui-même est une base de données relationnelle qui contient pour chaque fichier la correspondance entre un ou plusieurs noms logiques, le GUID du fichier et les noms physiques ainsi que les éléments de stockage sur lesquels les différentes copies (répliques) sont présentes. Cette base de donnée permet d'ajouter d'autres informations à chaque fichier (métadonnées), incluant le protocole d'accès pour chaque copie physique, selon la nature de l'élément de stockage sur lequel elle réside (xRootd, torrent, srm, fichier posix), le propriétaire du fichier, les droits d'accès, etc. Il est donc possible, à partir du nom logique d'un fichier ou de son GUID, d'obtenir les éléments de stockage où des répliques de ce fichier sont stockées, le nom physique du fichier sur chaque site et le protocole à utiliser pour y accéder.

```

psaiz@pctgd26:~
[pcitgd26] /home/psaiz > aliensh
[ aliensh 1.0.14x (C) AliEn/Alice, Andreas, Joachim, Peters@cern.ch/Derek, Feichtinger@cern.ch]
aliensh:[alice] [1] /alice/cern.ch/user/p/psaiz > ls -l
drwxr-xr-x  psaiz  psaiz      0 Aug 04 08:33  bin
drwxr-xr-x  psaiz  psaiz      0 Mar 26 14:19  dasdas\@das
drwxr-xr-x  psaiz  cc        59 Apr 27 11:26  nuevo.jdl
drwxr-xr-x  psaiz  psaiz     187 May 26 09:30  nuevo2.jdl
drwxr-xr-x  psaiz  psaiz      0 Mar 15 10:22  packages
drwxr-xr-x  psaiz  psaiz     187 Jun 08 10:12  quotafile
drwxr-xr-x  psaiz  psaiz      0 Mar 15 10:13  test
drwxr-xr-x  psaiz  psaiz      0 Mar 15 10:06  triggers
drwxr-xr-x  psaiz  psaiz      0 Mar 15 15:07  tmp
aliensh:[alice] [2] /alice/cern.ch/user/p/psaiz > cd bin
aliensh:[alice] [3] /alice/cern.ch/user/p/psaiz/bin > mkdir test
aliensh:[alice] [4] /alice/cern.ch/user/p/psaiz/bin/test > ls -l
drwxr-xr-x  psaiz  psaiz      0 Aug 04 08:39  checkStartTime.sh
drwxr-xr-x  psaiz  psaiz     119 Aug 04 08:39  alienroot
drwxr-xr-x  psaiz  psaiz     850 Mar 11 13:55  analysis.sh
drwxr-xr-x  psaiz  z2        65 Aug 04 08:39  certificate
drwxr-xr-x  psaiz  psaiz     612 Aug 04 08:39  checkStartTime.sh
drwxr-xr-x  psaiz  psaiz      81 Aug 04 08:39  createFile
drwxr-xr-x  psaiz  psaiz     146 Aug 04 08:39  gsiProduction
drwxr-xr-x  psaiz  psaiz     2401 Aug 04 08:39  hello
drwxr-xr-x  psaiz  psaiz     956 Aug 04 08:39  peekaboo.csh
drwxr-xr-x  psaiz  psaiz     2519 Aug 04 08:39  peekaboo.csh"
drwxr-xr-x  psaiz  psaiz      767 Aug 04 08:39  pp_sim.sh
drwxr-xr-x  psaiz  psaiz      203 Aug 04 08:39  pwd.sh
drwxr-xr-x  psaiz  cc       1024 Aug 04 08:39  runtest.sh
drwxr-xr-x  psaiz  psaiz      622 Aug 04 08:39  Sh_AnaESD_MC_CheckGenCascade.sh
drwxr-xr-x  psaiz  psaiz      0 Jun 15 18:34  test
drwxr-xr-x  psaiz  z2       2294 Aug 04 08:39  tpcRecAlien.sh
aliensh:[alice] [5] /alice/cern.ch/user/p/psaiz/bin/test >
    
```

Figure 64 : Exemple de l'interface de ligne de commande au catalogue de fichiers AliEn.

Cet espace de noms est présenté aux utilisateurs comme un système hiérarchique de fichiers UNIX. Les fichiers peuvent être inspectés et manipulés par une interface ligne de commandes style UNIX (voir Figure 64) à partir de l'invite de commande de l' « interface AliEn » (AliEn shell³³). Une interface

³³ Le shell (anglais pour « coquille ») est un programme qui fournit une interface pour l'utilisateur d'un système ou service informatique qui, à travers le shell, peut interagir avec lui. En particulier on appelle shell la partie la plus « externe » (d'où le nom de coquille) d'un système d'exploitation, c'est-à-dire son interface utilisateur.

graphique est aussi disponible grâce à MonALISA (voir Figure 65). À chaque utilisateur est allouée une quantité maximale d'espace utilisable pour ses fichiers (quota) qui est gérée grâce au catalogue. Le catalogue permet aussi de gérer le transfert des fichiers avec le système de stockage de masse (bandes magnétiques).

La fonction du catalogue est de fournir à tous les utilisateurs la même vision de l'ensemble des fichiers de ALICE indépendamment de l'emplacement de l'utilisateur et des fichiers. Tous les fichiers utilisés par le système hors-ligne de ALICE sont dans ce catalogue, y compris les entrées et sorties des tâches, les fichiers contenant les instructions pour exécuter les tâches par lots, les fichiers contenant les différentes versions de AliRoot pour tous les architectures et systèmes d'exploitation, le dictionnaire des métadonnées et les métadonnées. Il est aussi possible de définir des collections de fichiers, c'est-à-dire des fichiers qui contiennent une liste de fichiers à traiter, et ces collections sont aussi répertoriées dans le catalogue.

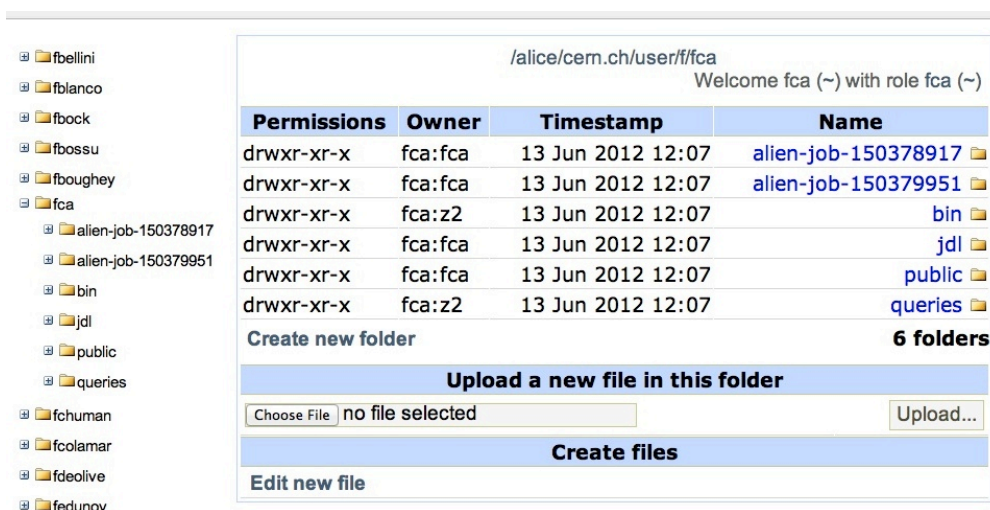


Figure 65 : Aperçu de l'interface graphique au catalogue de fichiers de AliEn.

L'accès au catalogue est possible avec plusieurs protocoles sécurisés : nom et mot de passe, certificats X.509 [140]³⁴, clefs SSH³⁵ ou identificateur (jeton) d'une tâche (en anglais *job token*). Le catalogue permet de définir plusieurs « rôles », tels que administrateur ou utilisateur, selon la politique en la matière de l'organisation virtuelle à laquelle appartiennent les fichiers. La protection des fichiers dans le catalogue est basée sur le modèle UNIX (utilisateur, groupe, monde) mais il est aussi possible de définir des listes de contrôle d'accès (ACL, en anglais *Access Control List*). En ALICE nous n'avons que les rôles d'administrateur et d'utilisateur et nous utilisons les protections standard UNIX.

Le catalogue offre une interface « programmatique » qui peut être utilisée par tous les utilisateurs de ALICE et qui est aussi intégrée dans le logiciel Root. Grâce

³⁴ X.509 est le nom d'un standard de cryptographie de l'Union internationale des télécommunications pour les infrastructures à clés publiques (PKI). La spécification d'un format standard de certificat électronique fait aussi partie du X.509.

³⁵ Protocole de communication sécurisé.

à cette interface il est possible, à partir d'un programme, de manipuler, lire et écrire des fichiers dans le catalogue.

L'interface au catalogue offre la plupart des instructions de l'interface UNIX au système de fichiers, plus des instructions spécifiques de l'environnement Grille. En outre elle permet de manipuler et surveiller les tâches par lots du système de gestion de la charge de travail.

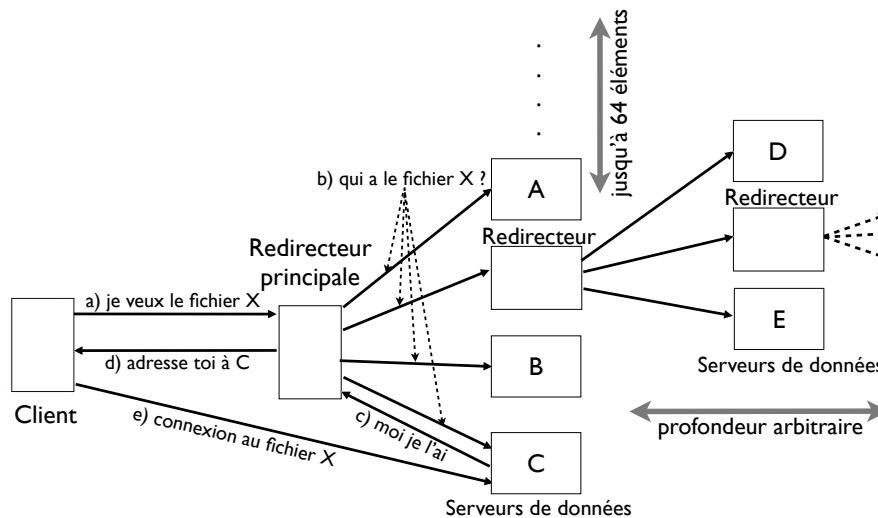


Figure 66 : Schéma du système de gestion des fichiers xRootd. La première fois que le client demande d'accéder au fichier X (a), le redirecteur demande à tous les serveurs de données qui a le fichier X (b). Quand un serveur répond (c), il communique l'information au client (d) qui ouvre une connexion directe au serveur pour accéder au fichier. L'information est sauvegardée dans la mémoire du redirecteur, et la deuxième fois qu'un client demande le fichier X les étapes b) et c) ne sont pas exécutées.

Chaque fichier (identifié par son GUID) possède des métadonnées qui lui sont associées. Font partie de ces métadonnées les informations sur le propriétaire du fichier et sur ses protections, c'est-à-dire qui est autorisé à faire quelle opération, selon le modèle UNIX. Inclus dans les métadonnées, le code de hachage md5sum³⁶ du fichier est utilisé pour s'assurer de l'intégrité de chaque nouvelle réplique. Chaque utilisateur peut ajouter ses propres métadonnées à ses fichiers. Une extension de la commande UNIX `find` permet d'interroger les métadonnées pour créer des listes de fichiers répondant à certains critères. Il existe aussi une interface à partir d'un programme pour interroger les métadonnées, utilisée par les programmes de simulation, reconstruction et analyse de AliRoot pour

³⁶ Ce code de hachage utilise l'algorithme MD5 (Message Digest 5) qui permet de calculer l'empreinte numérique d'un bloc de données. Si un changement, même petit, intervient dans les données, le code de hachage change et il permet de détecter une erreur de copie. Depuis son invention en 1991, on a montré qu'il est possible de créer deux fichiers différents avec la même empreinte numérique, donc MD5 n'est plus considéré comme un algorithme fiable du point de vue cryptographique. Néanmoins, la probabilité qu'une erreur de copie produise le même hachage est très limitée et MD5 reste une bonne mesure du succès d'une opération de réplique.

accéder aux fichiers de calibrage et d'alignement. L'OCDB, dont nous avons parlé dans le Chapitre 4, est implémenté comme un sous-ensemble du catalogue AliEn où les métadonnées sont optimisées pour l'accès aux informations de calibrage et d'alignement.

Le catalogue des fichiers est physiquement installé sur un ensemble de serveurs au CERN, avec redondance et sauvegardes automatiques. Le serveur qui gère l'accès au catalogue par le biais des différentes interfaces dont nous avons parlé fait aussi partie de cet ensemble de serveurs qui sont gérés par l'équipe Core Offline.

L'accès aux fichiers de la part d'un programme comporte une série d'étapes :

1. Le programme déclare vouloir ouvrir un fichier en lecture ou écriture et communique avec le catalogue, c'est-à-dire avec le serveur central qui gère l'interface, en lui communiquant les fichiers (GUID ou nom logique) et les opérations qu'il veut effectuer.
2. Le serveur communique avec le catalogue et il vérifie que les opérations demandées sont possibles et permises.
3. En cas de réponse positive du catalogue, le serveur donne à l'utilisateur l'autorisation d'effectuer les opérations demandées, les noms physiques des fichiers, l'élément de stockage sur lequel se trouvent les fichiers et le protocole d'accès. En cas de non-autorisation, ces informations ne sont pas fournies.
4. Si autorisé, le programme ouvre une connexion aux fichiers avec le protocole indiqué et effectue les opérations.

Toutes les communications sont sécurisées par des certificats électroniques. Il est important de noter que toutes les informations sur la propriété et les droits d'accès aux fichiers se trouvent dans le catalogue, et les répliques des fichiers de ALICE dans les différents éléments de stockage appartiennent toutes à un seul utilisateur. Cela simplifie énormément la gestion des droits d'accès, mais a demandé le développement d'un protocole hautement sécurisé pour l'accès aux données. En effet, pour un utilisateur de ALICE, la seule façon d'accéder à ses fichiers sur n'importe quel élément de stockage de la Grille est à travers l'interface du catalogue de AliEn.

Un problème qui a demandé un considérable développement technique est le protocole d'accès aux fichiers. Nous n'en donnerons ici qu'un aperçu très succinct. Le protocole « naturel » d'accès aux fichiers par un programme est défini pour un système UNIX par le standard POSIX³⁷. Cela suppose un système de fichiers qui offre une interface POSIX, comme par exemple le système de fichiers UNIX (ufs, en anglais *UNIX file system*). Si cela marche pour une machine seule, dans le cas d'une ferme d'ordinateurs la question devient compliquée car les fichiers sont partagés par plusieurs machines, et donc il faut utiliser un

³⁷ POSIX (interface portable au système d'exploitation UNIX, en anglais *Portable Operating System Interface for UNIX*) indique un ensemble de standards établis à partir de 1988 par l'IEEE (Institute of Electrical and Electronics Engineers) sous la désignation IEEE 1003. Ces standards ont pour but la standardisation des interfaces des logiciels conçus pour fonctionner sur le système d'exploitation de la famille UNIX.

système de fichiers distribués qui permet un accès partagé. Cela pose problème, car les systèmes d'accès partagés offrant une interface POSIX tels que nfs³⁸ sont assez peu performants et peu évolutifs. Le système de fichiers UNIX est une très bonne base de données hiérarchique, mais il a la granularité naturelle des disques utilisés, et dans le cas d'un système de stockage de grandes dimensions il sera préférable d'avoir une vision hiérarchique unifiée de l'ensemble des fichiers. Comme nous l'avons vu plus haut, pour réduire le coût du stockage et garantir la sauvegarde des données sur le long terme, une partie des fichiers est copiée sur bande magnétique. Au vu du nombre de fichiers impliqués et de la fréquence des opérations il devient essentiel d'avoir des systèmes qui gèrent automatiquement et de façon transparente la migration et le rappel des fichiers des bandes magnétiques.

Pour répondre à ces demandes, la Physique des Hautes Énergies a développé plusieurs systèmes de gestion de fichiers et de bandes (CASTOR [141], dCache [142], DPM [143], SAM [144], enstore [145], SRM [146], xRootd [139]) qui ont chacun leur propre interface d'application et leur protocole. Cela pose problème, car une tâche doit adapter sa stratégie d'accès aux fichiers selon le système de gestion de fichiers adopté par le site où elle est envoyée. Cela est contraire au concept même de Grille, et très compliqué à mettre en pratique. L'histoire des gestionnaires de fichiers et bandes dans la Physique des Hautes Énergies est très longue et compliquée, et nous ne souhaitons pas la retracer ici. Nous notons seulement que, une fois de plus pour des raisons de sources de financement et de politiques nationales ou « de laboratoire », il n'a pas été possible de développer un seul produit et il a fallu un processus contradictoire, parfois hostile et en tous cas assez pénible pour arriver au moins à un ensemble cohérent de solutions, sinon à une solution unique. Si nous pouvons considérer que l'exploration de différentes options est nécessaire pour parvenir à une solution optimale, et que la physique n'a pas souffert de ce cheminement assez « entropique », nous devons aussi déplorer que nous ayons raté l'occasion de mener une investigation concertée et rationnelle des différentes options techniques.

Face à cette situation, l'équipe Core Offline de ALICE a décidé en 2005 d'utiliser exclusivement xRootd. Si cette décision a en effet résolu le problème de l'hétérogénéité de l'accès aux fichiers, elle a nécessité le développement et le déploiement des interfaces qui donnent accès aux différents systèmes de gestion de fichiers via le protocole xRootd. Le travail a été considérable et, au-delà des problèmes techniques, il a rencontré une très forte résistance « politique », car ALICE était à l'époque la seule expérience à prendre une position si extrême. La situation s'est normalisée au fil des ans, car les autres expériences ont reconnu la validité technique de xRootd et ont commencé à l'intégrer elles aussi dans leurs systèmes hors-ligne. Le CERN a décidé en 2009 d'utiliser xRootd dans la gestion de toutes les données de LHC dans le cadre de la mise à jour de son système de gestion de données CASTOR. Maintenant, la majorité des centres de calcul qui

³⁸ nfs (système de fichiers en réseau, en anglais *Network File System*) est le nom d'un protocole introduit par Sun Microsystems qui permet à plusieurs ordinateurs de se partager des fichiers via un réseau.

participent à la Grille de ALICE ont installé le système xRootd, et donc la nécessité des interfaces est en train de disparaître.

Une description du système xRootd est au-delà du cadre de ce document, et nous nous contenterons de donner ici le principe de base de sa fonctionnalité (voir Figure 66). xRootd est un système client-serveur développé initialement pour l'expérience BaBar à SLAC. Lorsque le client demande un fichier au serveur, la demande est reçue par un « redirecteur » qui est connecté à des serveurs de données et à d'autres redirecteurs secondaires. Le redirecteur consulte une liste dans sa mémoire qui associe les noms des fichiers avec les serveurs de données où ces fichiers sont stockés. Si le nom du fichier est trouvé, le redirecteur retourne au client l'adresse du serveur et le chemin au fichier, et le client ouvre une connexion directement vers le serveur de données à travers le protocole xRootd. Si le nom du fichier n'est pas trouvé en mémoire, le redirecteur demande à tous les serveurs de données s'ils ont le fichier. Si un serveur répond, l'information est sauvegardée en mémoire et communiquée au client. Si le client rencontre une erreur dans l'ouverture du fichier, le redirecteur distribue une autre demande, met à jour l'information en mémoire et communique au client la nouvelle destination où le fichier peut être trouvé. Un fichier peut être présent sur plusieurs serveurs et le système dirige le client vers le serveur le moins chargé. Jusqu'à 64 serveurs peuvent être connectés à un redirecteur. Un nombre arbitraire de serveurs peut être remplacé par des redirecteurs secondaires, créant une arborescence sans limite de profondeur, ce qui permet d'agrandir le système indéfiniment. Il est important de noter que, du fait que le protocole xRootd utilise le protocole de communication réseau TCP/IP qui est la base d'Internet, il n'est pas nécessaire que les éléments du système soient « locaux », et donc il est possible d'imaginer une redirection entre les éléments de stockage de la Grille, par exemple en cas d'erreur ou de panne d'un des systèmes de stockage, sans repasser par le catalogue de fichier de AliEn. Cette possibilité a en effet été testée par ALICE avec de bons résultats, mais son intégration avec la fonctionnalité du catalogue de fichiers va demander encore du travail.

Une des caractéristiques les plus importantes de xRootd est sa capacité de fournir un accès optimisé aux fichiers éloignés. Comme nous l'avons vu, pour xRootd le fait qu'un fichier soit local (dans le même centre de calcul) ou éloigné n'est pas un problème. Ce qui change, c'est la performance de l'accès. Le problème ici n'est pas tellement la vitesse du réseau, que la vitesse de la lumière, ou plutôt le délai entre les deux hôtes (RTT ou temps d'aller et retour, en anglais *Round Trip Time*). En effet, si nous imaginons de lire un fichier de Root de 300 Mo qui contient un `TTree` avec 200.000 éléments, cela demande ~200.000 opérations « atomiques » de lecture. Sur un réseau Go local, le temps nécessaire est typiquement de 30s. Si maintenant nous voulons faire la même opération à partir du CERN pour le même fichier à SLAC, nous avons un réseau avec un débit de 100 Mb/s et un délai entre hôtes de ~160 ms. Le temps nécessaire est donc de 32.000 s pour les demandes de lecture et de 750 s pour transférer les données, ce qui fait neuf heures. L'équipe de xRootd a trouvé une solution assez élégante à ce problème [147]. Premièrement, chaque opération de lecture ne récupère pas seulement ce qui est demandé, mais beaucoup plus, et le reste est gardé dans une région tampon dans la mémoire du client. Une deuxième demande de lecture a

une grande probabilité de trouver les données déjà en mémoire sans générer une autre demande au serveur, ce qui entraînerait un délai égal au RTT. Les transferts sont effectués en parallèle en exploitant le fait que les réseaux à longue distance offrent la possibilité d'ouvrir plusieurs connections chacune à la vitesse nominale. Et pour finir, il existe aussi la possibilité d'effectuer les opérations de lecture de façon asynchrone en parallèle au traitement des données. Dans notre exemple, cela permet d'analyser le fichier éloigné en 60-90 s, ce qui rend l'utilisation de fichiers non locaux envisageable.

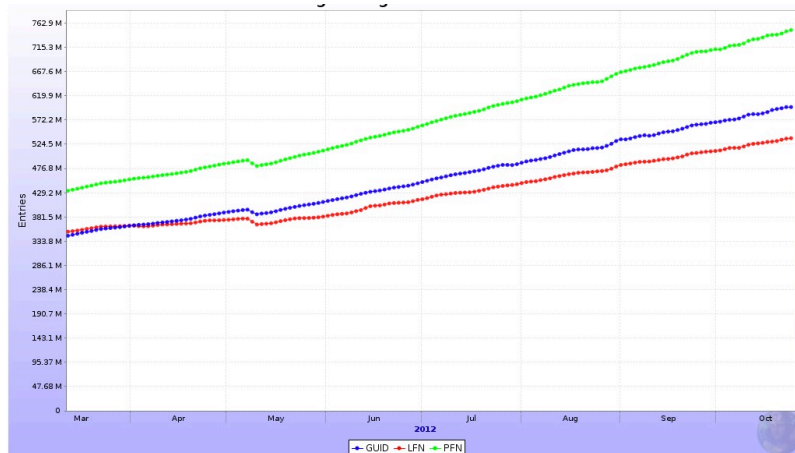


Figure 67 : Évolution du nombre d'entrées dans le catalogue de fichiers de AliEn. Le nombre de GUIDs est supérieur au nombre de noms logiques car les fichiers générés par des tâches en erreur ou les fichiers en cours d'effacement sont encore présents dans le catalogue, mais ils ne sont pas « exposés » aux utilisateurs avec un nom logique.

Le catalogue de fichier de ALICE est toujours en évolution pour répondre aux exigences des utilisateurs et anticiper l'évolution des demandes. Un des développements les plus importants est le choix automatique du système de stockage pour une demande de lecture ou d'écriture rendu possible par xRootd et MonALISA.

Lorsque une application demande au catalogue de fichiers de AliEn d'ouvrir un fichier, le catalogue a la possibilité de proposer un des éléments de stockage qui contiennent une réplique de ce fichier, même si ce n'est pas dans le même centre de calcul où le programme est en train de s'exécuter. La question est donc : quel élément de stockage faut-il proposer ? Comme nous le verrons plus loin, le système de Grille de ALICE est contrôlé par le système MonALISA. En particulier, MonALISA suit la performance de tous les éléments de stockage et définit de façon dynamique une « topologie » de la Grille où la « distance » entre un centre de calcul et les éléments de stockage est basée sur la performance du réseau et du système de stockage lui-même, ainsi que sur le RTT [148]. Cette information est déposée dans une base de données qui est consultée par le catalogue de fichiers de AliEn lorsque il doit proposer un élément de stockage. Au cas où l'élément choisi n'est pas utilisable parce que le fichier n'est pas présent ou bien l'élément de stockage lui-même n'est pas opérationnel, le système propose automatiquement le prochain élément de stockage le plus proche, et ainsi de suite.

Le nombre de répliques stockées pour chaque fichier résulte d'un compromis entre l'espace de disque disponible sur la Grille de ALICE et l'efficacité recherchée. Plus le nombre de répliques est grand, moindre sera la compétition pour le même fichier, mais plus rapidement l'espace sur disque sera rempli. Le standard dans ALICE est de faire trois répliques de chaque fichier sauvegardé dans le catalogue des fichiers, sur les éléments de stockage plus proches, selon la métrique expliquée plus haut.

Il est important à ce point d'expliquer la raison sous-jacente de certains choix architecturaux pour la réalisation du catalogue de AliEn. Au moment de sa conception, les avantages d'avoir un « espace des noms » unique pour tous les fichiers de la collaboration étaient évidents pour l'auteur et son équipe. Cela avait comme corollaire l'utilisation d'un catalogue unique, car l'utilisation de plusieurs catalogues aurait impliqué des délais de synchronisation qui auraient pu affecter l'exécution des tâches par lots. Supposons qu'un fichier soit produit par une tâche, et que la tâche suivante essaye de lire ce fichier. Si le catalogue auquel s'adresse la tâche suivante n'est pas encore synchronisé avec le catalogue où le fichier a été introduit à sa création, cette tâche donnera une erreur. Avec un catalogue unique, ce danger est complètement écarté. L'alternative était de développer un système de verrouillage distribué, ce qui aurait été extrêmement compliqué et long à mettre en œuvre.

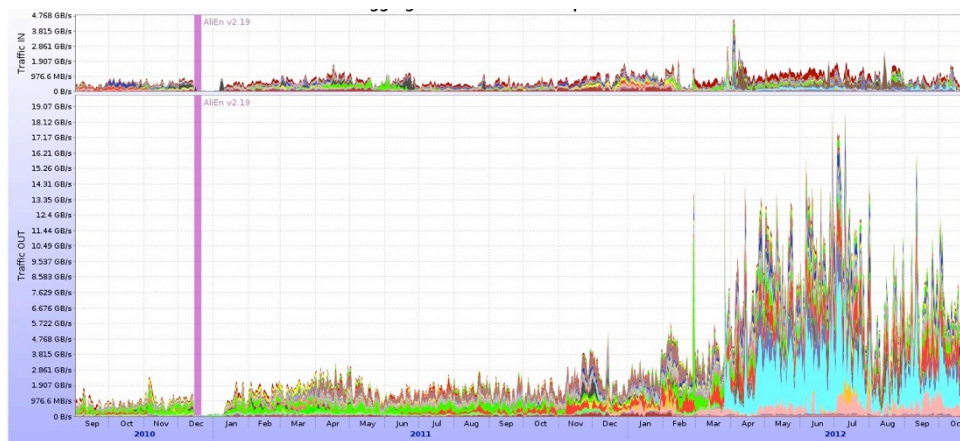


Figure 68 : Trafic intégré des éléments de stockage de la Grille de ALICE à partir du démarrage de l'accélérateur.

Un autre problème qui se pose, même avec un catalogue unique, est celui de la mise à jour des fichiers. Si un fichier est modifié, toutes ses répliques deviennent « invalides », et donc il faut les identifier, les effacer et les recréer. Ici encore, le problème de synchronisation d'une structure distribuée se pose. En outre, supposons qu'un des éléments de stockage soit non fonctionnel (panne ou manutention). Il faut en ce cas garder une « queue » de demandes de réplication à exécuter au moment où l'élément de stockage est à nouveau opérationnel. Tout cela aurait impliqué une architecture très compliquée et, par conséquent, longue à mettre en place, et qui serait restée très probablement fragile pendant longtemps. Pour pallier ce problème, nous avons donc décidé que les fichiers dans le catalogue ne peuvent pas être modifiés. Si un fichier doit être modifié, il faut en créer une nouvelle version.

Une fois ces décisions prises, il restait le problème que le fait d'avoir un catalogue unique constitue un point unique de défaillance et potentiellement un goulot de performance. Nous avons donc concentré nos efforts sur la robustesse du catalogue et sur l'optimisation de ses performances, sans oublier l'extension de ses fonctionnalités pour satisfaire les demandes des utilisateurs. Cette stratégie relève elle aussi des principes LCP expliqués dans le chapitre précédent : démarrer le développement avec un prototype à la fonctionnalité limitée, mais qui répond aux demandes des utilisateurs, et le développer au fur et à mesure que le problème et les demandes évoluent. Cela implique d'avoir fait des choix architecturaux solides au début et d'avoir les compétences et le « courage », en langage XP, d'opérer de vastes restructurations si nécessaire. Le catalogue des fichiers de AliEn a été en effet restructuré avec des changements profonds de l'organisation des tables, mais en faisant bien attention de causer le minimum de perturbations pour les utilisateurs.

La Figure 67 donne un aperçu de l'évolution du nombre d'entrées dans le catalogue des fichiers de AliEn pendant les six derniers mois. Le catalogue contient à présent 540 millions de fichiers et il s'accroît à un taux moyen de 840.000 fichiers par jour. La taille physique du catalogue est de 350 Go. La Figure 68 donne une idée de l'activité globale de la Grille de ALICE en termes d'accès aux données : depuis le démarrage de l'accélérateur, les éléments de stockage de la Grille de ALICE ont reçu des données en écriture à un taux moyen de 1,1 Go/s et ont servi des données en lecture à un taux moyen de 6,7 Go/s, dans les six derniers mois les taux moyens sont de 1,1 Go/s et 9,5 Go/s, en raison de l'intensification de l'activité d'analyse. Pour donner une idée de l'ampleur de ce trafic, nous pouvons estimer que la seule expérience ALICE génère un pour mille du trafic total d'Internet sur toute la planète.

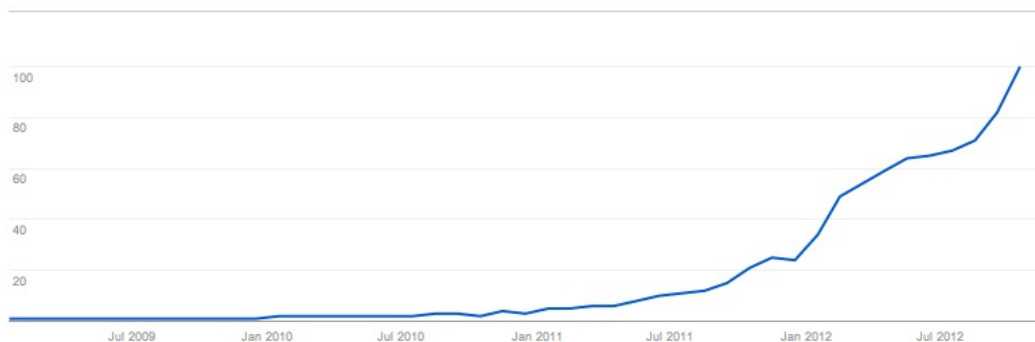


Figure 69 : Résultat de la recherche du terme « big data » sur Google trends montrant l'intérêt croissant pour le traitement de grandes quantités de données.

La gestion de l'accès au catalogue est effectuée par les serveurs centraux de ALICE. Il y a dix serveurs répondant aux demandes des tâches par lots qui sont exécutées sur les éléments de calcul de la Grille de ALICE et qui reçoivent 7,2 millions de connections par jour, c'est-à-dire une fréquence de 83 Hz par serveur. Un serveur additionnel est dédié à l'accès interactif des utilisateurs et il reçoit 950.000 demandes par jour, ce qui correspond à une fréquence de 11 Hz. Grâce à un travail constant d'optimisation, aucun de ces services n'est surchargé et il y a encore une bonne marge d'évolution.

Il est aussi intéressant de noter que le concept de « grandes données » (en anglais *big data*) est devenu très courant sur Internet et il est maintenant reconnu comme une des frontières du calcul à haute performance. Nous pouvons voir dans la Figure 69 le résultat d'une recherche faite avec « Google trends » [149] pour le terme « big data » montrant l'intérêt croissant pour le traitement des grandes quantités de données. Dans la Physique des Hautes Énergies, le traitement des grandes bases des données était déjà pris en considération en 1999 par le projet MONARC [122], et le premier projet de Grille dirigé par le CERN en 2001 s'appelait déjà Grille de Données Européenne. Une fois de plus nous pouvons constater que le calcul à haute performance en général et le calcul pour la Physique des Hautes Énergies en particulier sont aujourd'hui le précurseur de l'informatique grand public de demain.

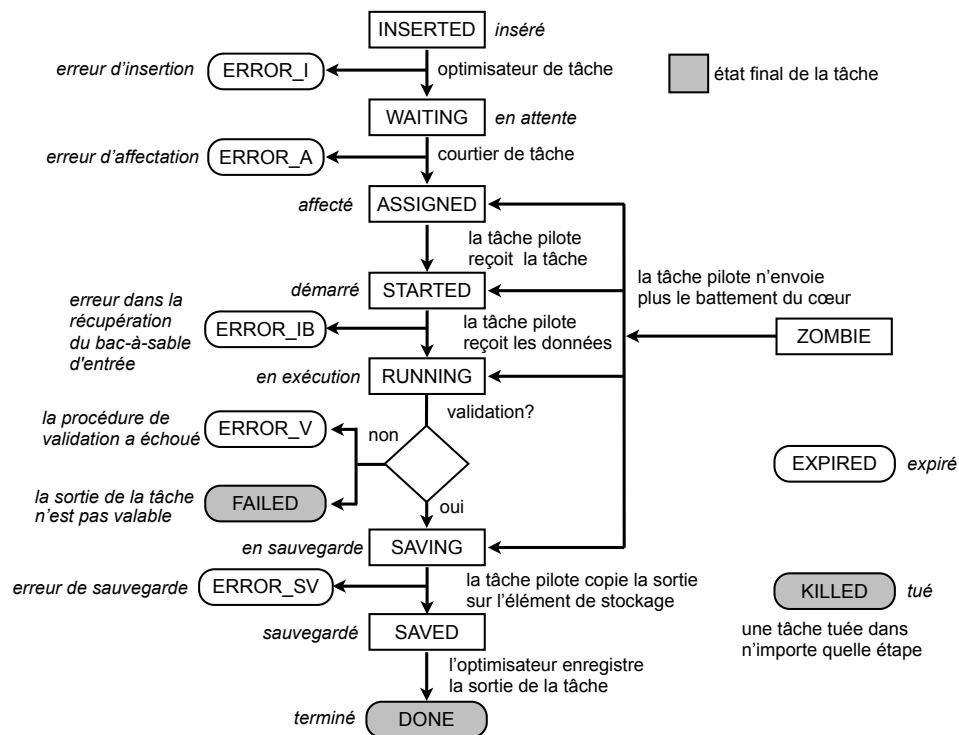


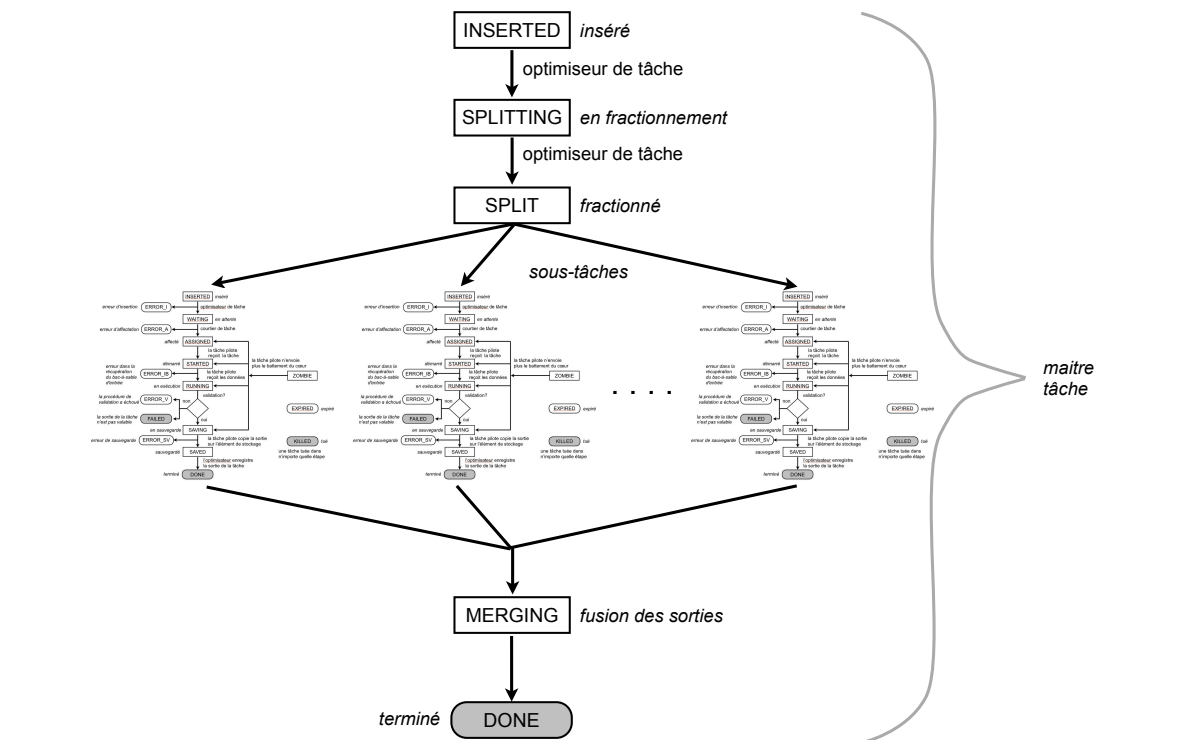
Figure 70 : Flux d'exécution d'une tâche du gestionnaire de charge de travail de AliEn. La tâche pilote envoie régulièrement des signaux au gestionnaire des ressources de calcul pour l'informer qu'elle est active. Ces signaux s'appellent « battements du cœur » (en anglais *heartbeat*). Si ces signaux cessent, la tâche est déclarée « ZOMBIE » (mort vivant) et après un certain temps elle est déclarée expirée et considérée comme perdue.

5.2.2 Le gestionnaire de charge de travail de AliEn

Le deuxième composant majeur du système AliEn est le système de gestion de charge de travail. Ce système permet aux utilisateurs de soumettre des tâches à exécuter par lots par le système informatique distribué de ALICE, inspecter leur état d'avancement et récupérer les résultats.

Une tâche à exécuter par le système est décrite dans un langage spécifique (JDL) qui pour AliEn est basé sur le langage Classads développé par le projet Condor

La demande est reçue par les serveurs centraux et, après avoir été analysée pour en vérifier la validité syntaxique, elle est placée dans une queue. La priorité avec laquelle cette demande sera traitée est déterminée par la priorité et les quotas de l'utilisateur. Si la demande contient une liste de fichiers à traiter, le service d'optimisation consulte le catalogue des fichiers pour savoir sur quel élément de stockage se trouvent les fichiers à traiter. À ce point, le gestionnaire de charge de travail envoie une tâche dans l'élément de calcul proche de l'élément de stockage qui contient les fichiers de la tâche. Le service élément de calcul sur la VOBx soumet une tâche au gestionnaire local de charge de travail par lots. La tâche envoyée ne contient pas les instructions de la tâche à exécuter, mais est une tâche générique appelée « tâche pilote » (en anglais *pilot job*) ou agent de tâche (en anglais *job agent*).



l'appariement est, pour la tâche pilote, de récupérer les paquets de logiciels demandés par la tâche originale. Cela est effectué par le module packman (gestionnaire de paquets de logiciel, en anglais *PACKage MANager*) à travers une version du logiciel BitTorrent [151] modifiée pour ALICE. La tâche pilote récupère ensuite les fichiers associés à la tâche originale, qui sont collectivement appelés bac-à-sable (en anglais *sandbox*), et qui sont spécifiés dans le JDL. Ensuite, la tâche pilote récupère la charge de travail (en anglais *payload*) contenue dans le JDL sous forme d'instructions et l'exécution commence. Quand l'exécution est terminée, les fichiers de sortie sont validés et enregistrés dans le catalogue de fichiers de ALICE. Si l'utilisateur a spécifié une procédure de validation, celle-ci est exécutée et le résultat est évalué. À la fin de l'exécution d'une tâche, la tâche pilote interroge encore la queue centrale pour voir s'il y a d'autres tâches du même utilisateur qui peuvent être exécutées dans le temps restant alloué à la tâche pilote. Un aperçu du flux d'exécution d'une tâche et des différentes conditions d'erreur est schématisé dans la Figure 70.

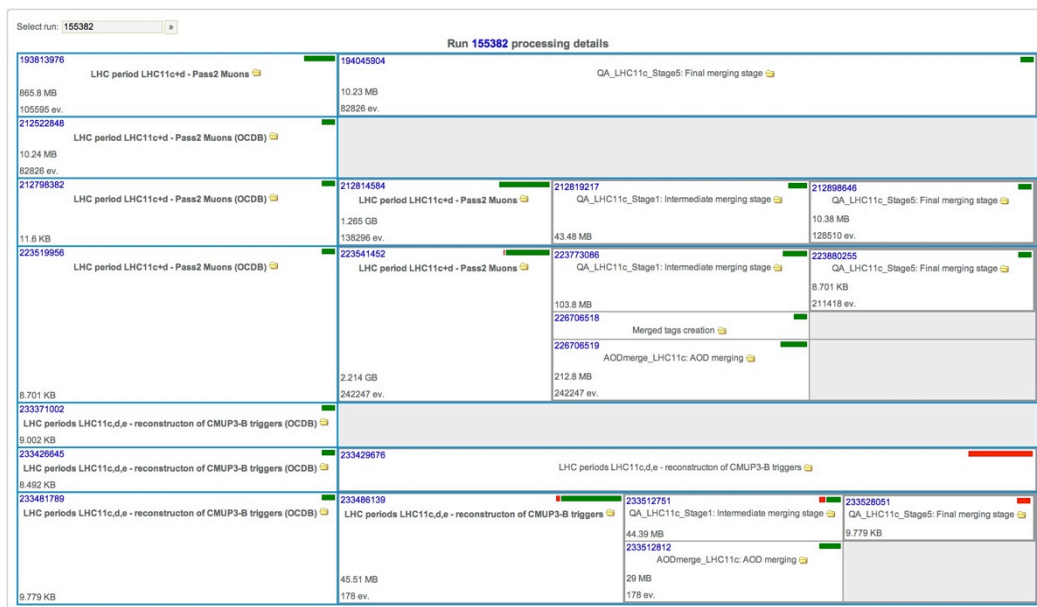


Figure 72 : Vue d'ensemble des tâches impliquées dans la reconstruction d'une période de prise de données. La dépendance entre ces tâches peut être exprimée comme un graphe acyclique orienté. L'ensemble est visualisé avec le système de surveillance MonALISA.

Vu que le nombre de fichiers pouvant être traités par une seule tâche peut être très élevé, il est assez improbable qu'ils se trouvent tous sur le même élément de stockage. Dans ce cas, le service d'optimisation des tâches (en anglais *job optimiser*) détermine les fichiers qui sont sur les mêmes éléments de stockage et divise la tâche en sous-tâches qui traitent des groupes de fichiers situés sur le même élément de stockage. La tâche originale devient alors une « tâche maître » (en anglais *master job*), c'est-à-dire une tâche virtuelle qui fonctionne comme un conteneur pour les sous-tâches générées. Chaque sous-tâche est ainsi destinée à être exécutée dans le site proche de l'élément de stockage qui contient ses fichiers et le processus de courtage de tâche peut procéder comme nous l'avons

décrit précédemment. Quand toutes les sous-tâches d'une tâche maître sont terminées, le système démarre, le cas échéant, une tâche séparée qui fusionne les différents fichiers partiels produits par les sous-tâches avant de les enregistrer dans le catalogue des fichiers de AliEn. La Figure 71 montre le schéma de fonctionnement d'une tâche maître.

Dans un contexte complexe tel que le traitement des données d'une expérience de Physique des Hautes Énergies, il n'est pas rare que la sortie d'une ou plusieurs tâches devienne l'entrée d'une ou plusieurs tâches et ainsi de suite. Nous pouvons représenter cela de façon formelle comme un graphe orienté acyclique (DAG, en anglais *directed acyclic graph*)³⁹. Il est possible avec le gestionnaire de charge de travail de AliEn d'exprimer cette dépendance et de créer des suites de tâches de complexité arbitraire. Dans ce cas, une tâche est démarrée automatiquement quand toutes les tâches dont elle dépend sont terminées avec succès. Il est aussi possible de faire démarrer une tâche quand un certain pourcentage des tâches dont elle dépend est terminé avec succès. Nous pouvons voir dans la Figure 72 un exemple d'une suite de tâches arrangées dans un graphe de dépendances orienté acyclique pour la reconstruction d'un période de prise de données.

Comme nous l'avons fait pour le catalogue des fichiers, nous ferons ici quelques remarques sur les choix architecturaux du gestionnaire de ressources de calcul de AliEn. La première version du courtier des tâches proposée par EDG était un service complètement centralisé. L'idée était de disposer d'un service avec une vue globale de la Grille qui pouvait trouver le meilleur endroit pour chaque tâche. Lorsque ce service recevait une demande de travail, il interrogeait tous les centres de calcul de la Grille pour déterminer lequel était le plus apte à répondre à cette demande, selon la disponibilité du centre, l'autorisation d'exécuter des tâches par l'organisation virtuelle à laquelle la tâche en question appartenait, la charge de travail déjà présente sur le centre, la disponibilité des paquets de logiciels demandés, etc. Son choix fait, ce service devait redemander au centre en question s'il était encore disposé à accepter cette tâche et, si la réponse était positive, démarrer la procédure d'authentification et d'autorisation pour pouvoir soumettre la tâche, chaque transaction étant sujette à des délais de communication entre hôtes.

Il est évident qu'une interrogation directe des centres de calcul n'est pas envisageable, car cela prendrait beaucoup de temps. Donc, la solution retenue fut d'interroger le service d'information de la Grille, mais pour obtenir des informations fiables, ce service devait être mis à jour avec une grande fréquence, ce qui avait pour résultat de l'engorger rapidement avec un grand nombre de mises à jour. Le danger d'opérer avec des informations périmées était de générer des erreurs lors de la soumission des tâches, ce qui aura pu entraîner une corruption de la queue des demandes et des délais supplémentaires pour la reprise sur erreur. Donc, ni le service de gestion et courtage de la charge de travail, ni le système d'information de la Grille ne pouvaient évoluer et suivre la

³⁹ Un graphe est un ensemble de points, parmi lesquels certaines paires sont reliées par des liens. Si les liens sont orientés, nous parlons de graphe orienté. Si un graphe orienté ne présente pas de cycle, c'est-à-dire s'il n'est pas possible, en partant d'un point, d'y revenir en suivant la direction des liens, nous parlons de graphe orienté acyclique.

probable augmentation du nombre de demandes et de centres de calcul. En effet, le gestionnaire de charge de travail introduit par EDG et EGEE ne fut jamais en mesure d'effectuer un courtage basé sur l'emplacement des fichiers, et il est maintenant abandonné.

La solution choisie par AliEn fut d'avoir un gestionnaire de charge de travail « sans état »⁴⁰ et de déléguer aux sites le courtage de la charge de travail. Dans cette option, il n'y a pas d'optimisation « globale » car chaque site ne voit que son propre état et la queue centrale, qui n'a pas d'état. Cette option pouvait être considérée, en principe, comme sub-optimale. En effet, une tâche est exécutée non pas par le « meilleur » centre de calcul « dans l'absolu », mais par le premier centre de calcul qui peut l'exécuter. Les tenants d'une optimisation globale objectaient que cela aurait conduit à une utilisation sub-optimale de la Grille et donc à une perte de ressources. Si par principe cela est vrai, en pratique « le meilleur » endroit où exécuter une tâche est un concept indéfini dans un système complexe et dynamique comme une Grille de calcul de grandes dimensions. S'assurer qu'une tâche puisse se terminer dans de bonnes conditions est déjà faire un grand pas vers une utilisation efficace de la Grille. De plus, l'idée de déléguer aux sites les opérations de courtage permet d'introduire un contrôle très fin du courtage par rapport à l'état du site, en tenant compte d'une foule de paramètres qu'il serait impossible de prendre en considération par un service centralisé. De plus, les informations que le service de courtage local a sur l'état du site sont, par définition, toujours à jour. Un courtage local permet aussi aux administrateurs de système d'intervenir sur le fonctionnement de leur site sans devoir passer par les services centraux.

La deuxième option originale qui fut introduite par AliEn est l'utilisation des tâches pilotes. Cela peut sembler une complication, car elle introduit une série de transactions additionnelles avec le gestionnaire central. Face à cet inconvénient, les tâches pilotes offrent plusieurs avantages qui améliorent considérablement l'efficacité d'utilisation de la Grille. Si une tâche échoue après un certain temps pour un défaut de configuration du site, le temps de calcul utilisé par cette tâche est perdu, sans compter qu'il faut s'assurer qu'il n'y a pas eu d'enregistrement partiel d'un ensemble de fichiers dans le catalogue qu'il faut « nettoyer ».

En plus, la queue du gestionnaire des ressources n'est pas complètement sans état, car elle garde l'état de chaque tâche pendant son exécution. Chaque fois qu'une tâche échoue pour un problème de configuration du site, la tâche doit être mise à disposition à nouveau pour être exécutée par le gestionnaire, en augmentant sa charge de travail. Si un site rencontre un problème qui cause l'échec immédiat des tâches, le système peut interpréter cela comme un signe de grande efficacité du site, car il traite très rapidement chaque tâche qui lui est attribuée, et lui assigner davantage de tâches qui échoueront à leur tour. Cela va causer des « trous noirs » qui peuvent bloquer tout le système, car la queue du gestionnaire va être submergée de demandes de mise à jour des tâches envoyées à ce site, qui vont passer de WAITING à ASSIGNED, STARTED, RUNNING, FAILED et encore à WAITING en boucle (voir Figure 70).

⁴⁰ Un service informatique est « sans état » (en anglais *stateless*) quand il n'a pas de paramètres internes qui dépendent de son historique. Il est « avec état » (en anglais *statefull*) dans le cas contraire.

L'utilisation des tâches pilotes minimise ces effets, car si le site n'est pas correctement configuré, la tâche pilote échoue sans avoir interagi avec la queue du gestionnaire central et sans avoir consommé de précieuses ressources de calcul ou créé une situation inconsistante par rapport au catalogue des fichiers. Donc, le prix à payer pour le mécanisme des tâches pilotes est bien compensé par une diminution des échecs d'exécution. Le fait qu'une tâche pilote puisse exécuter plusieurs demandes augmente davantage l'efficacité du mécanisme, car il réduit les frais de soumission des tâches. En plus, la tâche pilote offre une enveloppe très utile pour effectuer des opérations standard telles que l'installation du logiciel. Bien sûr, il y a encore des tâches qui échouent pour maintes raisons, mais les tâches pilotes rendent le système très stable et, en même temps, évolutif.

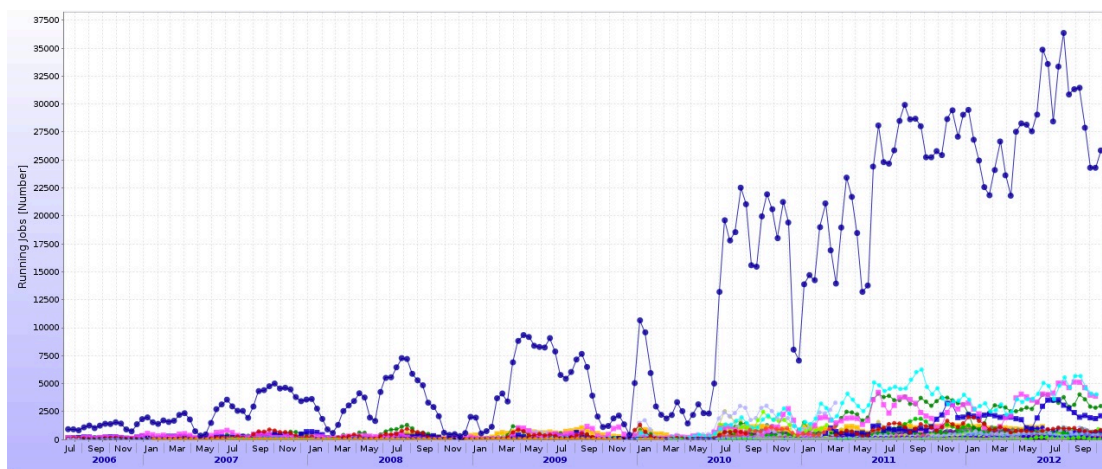


Figure 73 : Évolution du nombre de tâches sur la Grille de ALICE depuis 2006.

Une question très importante pour le fonctionnement de la Grille est la distribution du logiciel, car il y a entre une et trois nouvelles versions de AliRoot à distribuer sur plus de quatre-vingt centres de calcul. La solution qui est utilisée par AliEn consiste à installer le logiciel de AliRoot sur chaque machine chaque fois qu'une tâche démarre son exécution. Cette solution peut sembler sub-optimale, car le même logiciel est installé sur toutes les machines qui exécutent AliRoot dans un centre de calcul, et on pourrait penser qu'il suffirait de l'installer une fois seulement dans une zone de disque partagée. En effet, l'utilisation d'un espace de disque partagé, qui était la première solution retenue par ALICE, impose d'avoir la VOBox, qui gère cet espace, et tous les nœuds du centre sur le même réseau, avec des problèmes de performance et de sécurité. En utilisant BitTorrent, la première tâche qui demande une nouvelle version de AliRoot sur un centre doit l'importer des serveurs centraux de ALICE au CERN, et pour cela il n'y a pas de différence avec l'utilisation d'un espace de disque partagé. Les autres tâches qui utilisent la même version utiliseront comme source, grâce à BitTorrent, les copies déjà existantes sur le site. La charge pour le réseau de la ferme d'ordinateurs du site est la même, car même en utilisant un espace partagé, le logiciel doit être copié dans la mémoire de la machine l'exécutant à travers le réseau. L'élimination de la nécessité d'un espace de disque partagé est une simplification de taille. Pour rendre le système le plus efficace possible, la

taille de la version d'« exportation » de AliRoot a été soigneusement optimisée et elle est maintenant de 400 Mo avec un taux de croissance assez lent. Le système dans sa globalité est très efficace, car chaque version du logiciel n'est pas « installée » centralement, mais « mise à disposition », et seul le logiciel effectivement demandé est transféré et installé. Même s'il était possible d'utiliser chaque centre comme source, pour des raisons de sécurité seuls les serveurs du CERN et les machines sur le même réseau local sont acceptés comme sources.

Encore une fois, la stratégie utilisée a été de réaliser un système assez simple et robuste et de le faire évoluer au gré de l'expérience acquise et des demandes des utilisateurs. La robustesse et la relative simplicité du système de départ ont permis de l'utiliser à grande échelle et très tôt, de manière à récolter une grande quantité d'informations sur sa performance et ses caractéristiques, ce qui a permis d'en guider l'évolution. Ici aussi la stratégie LCP présentée dans la Section 3.1 a fait ses preuves.

Là aussi nous avons opté pour une solution centralisée qui simplifie considérablement la gestion du système. Par exemple, la gestion des priorités est très compliquée si nous avons à manipuler plusieurs queues de travail, car elles doivent se synchroniser pour gérer non seulement la priorité absolue des demandes qui leur sont propres, mais aussi la priorité relative des demandes dans des queues différentes. Il en va de même pour les quotas. Avec une seule queue, tout cela est assez simple. Pour éviter que cette queue centrale devienne un goulot d'étranglement de la performance, nous avons opté pour une décentralisation maximale des services, en gardant une perspective globale sur le système, et nous avons pris soin de réduire au maximum l'information de l'état gardé par le système. Ce qui est remarquable, c'est que chaque mesure que nous avons prise pour minimiser la charge de travail du gestionnaire de ressources n'a pas diminué la fonctionnalité ou l'efficacité du système, mais au contraire a ouvert de nouvelles perspectives, comme on l'a vu avec l'introduction des tâches pilotes.

Le gestionnaire de charge de travail de ALICE est en fonction depuis 2002. La Figure 73 montre l'évolution du nombre de tâches sur la Grille de ALICE depuis 2006. Dans les six derniers mois, le nombre moyen de tâches a été de 32.000.

Nous pouvons nous demander maintenant s'il n'y a pas vraiment de place dans cette architecture pour une optimisation globale. La réponse est oui, mais la route est encore longue. Nous en avons vu un premier exemple avec le mécanisme de découverte automatique des éléments de stockage proches d'un élément de calcul donné. Il est évident qu'en suivant le même principe il y a encore beaucoup de possibilités d'optimisation, sans pour autant changer la philosophie de base du courtage de tâches de AliEn. Grâce au système de surveillance MonALISA, dont nous parlerons brièvement dans la prochaine section, nous disposons d'une information très complète et très détaillée du fonctionnement de la Grille de ALICE. En nous basant sur cette information nous pouvons « diriger » le comportement du gestionnaire de ressources de façon à optimiser l'utilisation des ressources, sur la base de l'histoire de leur fonctionnement et de leur état actuel. Cette optimisation doit être aussi peu invasive que possible et respecter les différentes échelles de temps d'acquisition et de traitement de données, de façon à ne pas tomber dans le paradoxe du

gestionnaire des ressources de EDG, qui en voulant être « optimal » n'était pas capable de fournir assez de travail à la Grille.

Donc, il faudra bien garder les différents systèmes découplés de façon à « diriger » le gestionnaire des ressources vers une situation de plus grande efficacité, en surveillant en continu les effets à travers un mécanisme de rétroaction encore à définir. Nous sommes en face d'un système hautement non linéaire et complexe où toute action rigide ou simpliste risque de produire l'effet contraire à celui souhaité. Ce système mériterait une analyse « théorique » pour en déterminer les composants principaux et les « attracteurs ». Un élément primordial de cette analyse serait une simulation prédictive et détaillée de la Grille, qui malheureusement n'a jamais été développée.

5.2.3 Le système de contrôle MonALISA

Le système MonALISA (MONitoring Agents using a Large Integrated Services Architecture) de contrôle de la Grille de ALICE n'a pas été conçu par le projet Offline de ALICE. Néanmoins, ALICE est la seule expérience qui a adopté intégralement le système pour en faire un élément fondamental de son infrastructure distribuée de calcul. Au cours des années passées depuis l'adoption de MonALISA par ALICE en 2006, et grâce à la présence en son sein d'un des développeurs du système, le groupe Core Offline de ALICE a été à l'origine de plusieurs développements importants du système. Nous donnerons donc une brève description de MonALISA au vu de son rôle dans la Grille de ALICE.

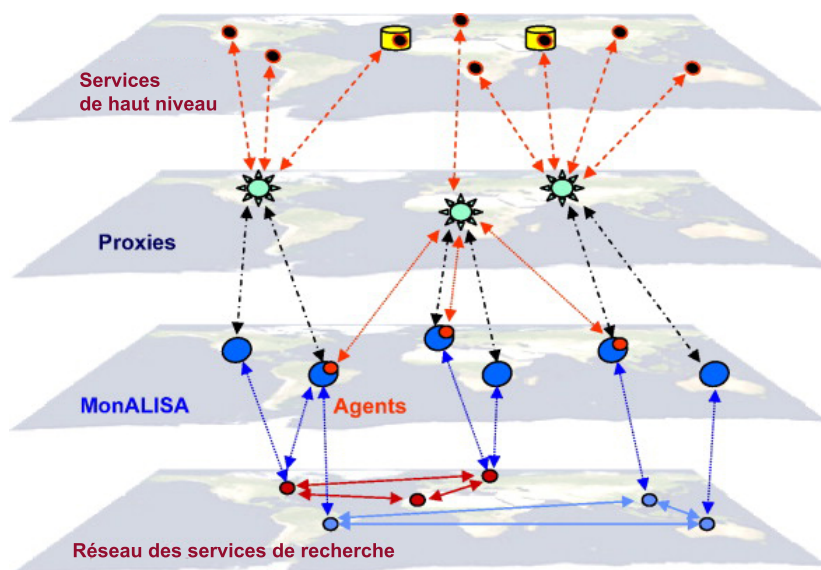


Figure 74 : Les quatre couches, les services de base et les composants de l'infrastructure de MonALISA. Le terme Proxies indique la couche de services de procuration.

L'objectif principal du développement de MonALISA [152, 153, 154] est de fournir un cadre souple, capable d'utiliser en temps quasi réel des informations pour un suivi complet d'un grand nombre de tâches, d'installations de calcul et de réseaux étendus afin de contrôler et optimiser les flux de travail des systèmes complexes distribués. Le système est conçu comme un ensemble de sous-

systèmes basés sur des agents autonomes et auto-descriptifs qui sont enregistrés en tant que services dynamiques. Ces services sont en mesure de collaborer et de coopérer à l'exécution d'une vaste gamme de tâches distribuées de collecte et de traitement d'informations.

Une telle architecture à agents multiples [155] est bien adaptée au fonctionnement et à la gestion des systèmes distribués à grande échelle, en fournissant des services d'optimisation globale capables d'orchestrer des ressources de calcul, de stockage et de réseau afin de soutenir des flux de travail complexes. En surveillant en temps réel l'état des sites de la Grille et leurs connexions réseau de bout en bout, les services de MonALISA sont en mesure de détecter rapidement des conditions problématiques, d'aider à les diagnostiquer et, dans de nombreux cas, à les résoudre, avec pour conséquence une augmentation de la fiabilité globale et la possibilité de gestion des systèmes informatiques distribués. L'architecture de MonALISA, présentée dans la Figure 74, est basée sur quatre couches de services globaux développés à partir de la technologie Java.

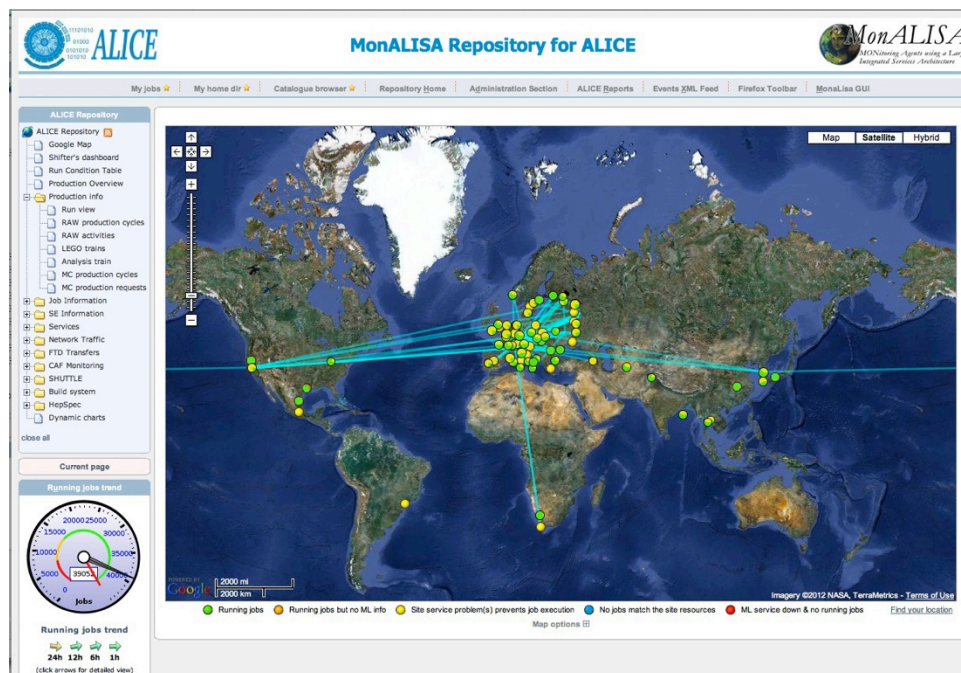


Figure 75 : La page principale du système de surveillance MonALISA pour l'expérience ALICE.

Le réseau des services de recherche et découverte (en anglais *Look Up Services*, LUS) fournit la découverte et l'enregistrement dynamiques de tous les autres services et agents. Les services de MonALISA sont en mesure de se découvrir les uns les autres dans l'environnement distribué et d'être découverts par les clients intéressés. L'enregistrement utilise un mécanisme de bail dynamique qui permet d'avoir un aperçu de l'état du système.

La deuxième couche représente le réseau des services de MonALISA, qui accueille de nombreuses tâches de surveillance, grâce à l'utilisation d'un moteur d'exécution multi-filière, ainsi qu'une variété d'agents faiblement couplés qui analysent les informations collectées en temps réel. Les informations recueillies

peuvent être stockées localement dans des bases de données. Les agents sont en mesure de traiter l'information au niveau local et de communiquer avec d'autres services ou agents pour effectuer des tâches d'optimisation globale. Un service dans le cadre MonALISA est un composant qui interagit de façon autonome avec d'autres services, soit par procuration, soit par l'intermédiaire des agents dynamiques qui utilisent des protocoles auto descriptifs. En utilisant le réseau de services de recherche, un registre de services distribués et les mécanismes de découverte et de notification, les services sont en mesure d'accéder l'un à l'autre de façon transparente.

La troisième couche est composée de services mandataires (en anglais *Proxy*)⁴¹. Elle fournit un multiplexage intelligent de l'information demandée par les clients ou par d'autres services et est utilisée pour une communication fiable entre les agents. Elle peut également être utilisée pour imposer un contrôle d'accès pour assurer un accès sécurisé à l'information recueillie.

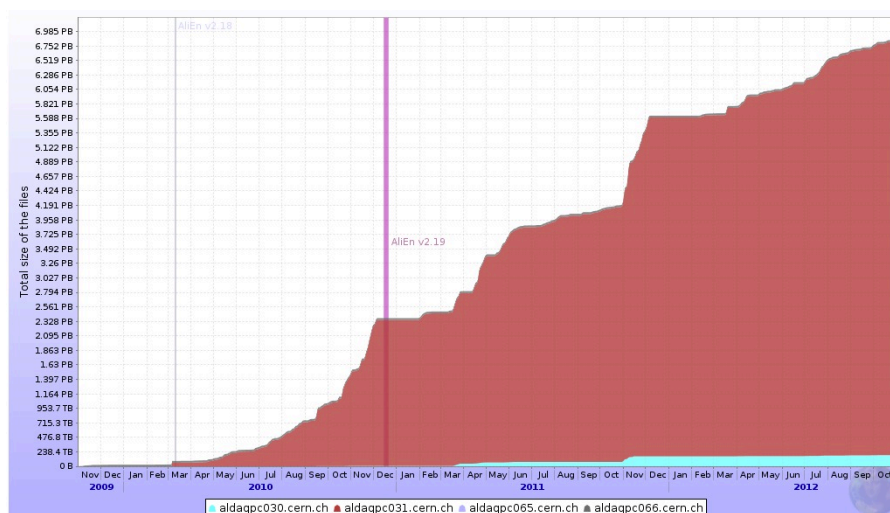


Figure 76 : Données brutes récoltées par ALICE.

Les services de niveau supérieur s'adressent à la couche des services mandataires pour accéder aux informations recueillies. Un mécanisme d'équilibrage de charge est utilisé pour allouer dynamiquement ces services au meilleur service mandataire. Les clients d'autres services ou leurs mandataires peuvent obtenir toutes les données en temps réel ou historiques en utilisant un mécanisme de prédicats pour demander ou souscrire à certaines valeurs mesurées. Les demandes sont mises dans une file d'attente et servies par une des filières d'un réservoir de filières géré par le système.

La particularité de MonALISA par rapport à d'autres systèmes de surveillance est de recueillir tout type d'information dans une approche synergique, nécessaire pour contrôler et optimiser l'exécution des applications de données intensives sur les systèmes distribués à grande échelle. De plus, le mécanisme de sélection

⁴¹ Dans les réseaux informatiques un mandataire (en anglais *Proxy*) est un composant logiciel qui joue le rôle d'intermédiaire pour accéder à un autre réseau. Dans une Grille un mandataire est un composant qui suppose l'identité d'un utilisateur et il peut faire certaines actions en son nom pendant un certain temps.

et de construction des informations demandées par d'autres services se fait dans la mémoire de chaque service de MonALISA et de cette façon il est possible de traiter plusieurs dizaines de milliers de messages de surveillance par seconde en temps quasi-réel.

Les paramètres à surveiller sont passés aux modules de surveillance de MonALISA par le biais d'une interface d'application appelée ApMon. ApMon utilise des datagrammes UDP pour le transport de l'information (encodé en format XDR) et utilise un numéro de séquence pour vérifier l'intégrité de tous les rapports de surveillance.

Dans le cas d'une ferme de processeurs, les administrateurs système ont la possibilité de contrôler quels paramètres sont surveillés. La confidentialité des flux de données entre les services et les clients doit également être assurée. Pour cela, MonALISA met en œuvre une infrastructure de sécurité qui repose sur les certificats standard X.509v3 pour l'authentification, sur les listes de contrôle d'accès (ACL) pour l'autorisation et aussi sur le protocole de communication Secure Sockets Layer (SSL).

Le système est capable de stocker des données en temps quasi réel, de les agréger et de les récupérer rapidement sur demande, et de gérer les données stockées de façon transparente à travers une interface abstraite. Différents formats de stockage sont disponibles, la base de données d'arrière-plan par défaut pour tous les services MonALISA étant PostgreSQL. En plus de tout un système de stockage permanent, les paramètres surveillés sont conservés dans une mémoire tampon volatile servant de cache pour les données récentes. Cette mémoire volatile est suffisante pour fournir une fonctionnalité de surveillance rapide, capable de répondre à des demandes d'historique dans la limite de la quantité de données que le service est en mesure de garder en mémoire. Toutes les structures sont utilisées en parallèle. Les données sont écrites dans la base de données seulement à la fin d'un intervalle d'agrégation, en utilisant des opérations d'entrée et sortie asynchrones.

Lors des premiers tests à échelle réelle en 2005, il fut évident que l'utilisation d'un système distribué comprenant plusieurs milliers d'éléments était impossible sans un système approprié capable de surveiller l'état global de la Grille jusqu'à ses éléments de base, matériels et logiciels. Le système de surveillance doit suivre l'utilisation des ressources, la performance des logiciels et tout autre événement d'intérêt pour les développeurs, les opérateurs et les utilisateurs. Dans le même temps, le système de surveillance doit être non intrusif et utiliser une quantité minimale de ressources de calcul. Sur la base de ces exigences, MonALISA a été choisi comme solution globale pour la surveillance de la Grille de ALICE.

Le service de MonALISA est installé sur chaque VOBox et sur chacun des serveurs centraux de la Grille de ALICE. Ces services sont utilisés pour collecter les informations sur chaque tâche active, transfert ou service, et de mettre l'information à la disposition de tout client sous forme brute ou agrégée. Par exemple, l'information détaillée est disponible pour chaque tâche, mais des valeurs telles que le nombre de tâches dans chaque état ou la puissance de calcul totale consommée par toutes les tâches en cours sur ce site, sont également disponibles.

Chaque composante de AliEn, incluant les services centraux, les services des VOBox et les tâches pilotes, est instrumentée avec ApMon et il envoie les informations au service MonALISA local qui les agrège et les envoie au service MonALISA central. Les services qui ne font pas partie de AliEn sont eux aussi surveillés par un petit ApMon externe à base de script. La page d'état global du référentiel MonALISA montré dans la Figure 75 présente l'état de tous les services s'exécutant sur un site. Il est également le point de départ pour le débogage d'un problème par les experts, qui peuvent immédiatement voir quel composant ne fonctionne pas correctement et quelle est la cause d'un échec, et sont aussi prévenus par des courriels envoyés automatiquement.

Une fonctionnalité très importante des modules de surveillance de MonALISA est la possibilité d'effectuer des tests simples qui vérifient l'état et la disponibilité des services intergiciels et les autorisations (mandataires de Grille), nécessaires pour interagir avec ces services.

Les tâches pilotes sont instrumentées avec ApMon et elles envoient des données sur l'état des tâches en cours (comme « a commencé », « en cours d'exécution », « enregistrement », etc.) et sur l'utilisation des ressources (taille de mémoire, espace de disque, temps d'unité de traitement, puissance de l'unité de traitement en SPECint⁴² ou HEP-SPEC2006⁴³). Les systèmes qui exécutent des services critiques pour AliEn sont également surveillés pour assurer une détection immédiate d'une défaillance du service ou d'autres événements critiques.

Une grande quantité de paramètres de tâches et du système de gestion de charge de travail sont disponibles en temps réel ou agrégés sur des périodes rétrospectives arbitraires du point de vue du site ou de l'utilisateur. Différentes visualisations à divers niveaux d'agrégation de l'utilisation de l'ensemble des ressources disponibles sont également accessibles : le temps d'exécution totale utilisé par ALICE comme temps écoulé, mais aussi en termes d'unités SPECint2000 x s ou HEP-SPEC06 x s, l'efficacité des tâches mesurée comme le rapport entre le temps utilisé par l'unité d'exécution et le temps écoulé, et ainsi de suite. Le trafic réseau est également surveillé (entrant et sortant) ainsi que les paramètres clés du système: utilisation de la mémoire par les tâches sur chaque site, utilisation de la mémoire virtuelle, utilisation de la mémoire système,

⁴² SPECint est une spécification de référence pour la puissance de traitement d'opérations entre entiers d'une unité. Il est maintenu par la corporation d'évaluation des spécifications de référence de la performance des unités de calcul (SPEC, en anglais *Standard Performance Evaluation Corporation*). La suite de tests de performance SPECint est la composante entière de la suite de tests SPEC. La dernière norme de SPECint est nommée CINT2006 ou SPECint2006.

⁴³ HEP-SPEC06 est la nouvelle unité de référence pour la Physique des Hautes Énergies développée pour remplacer l'obsolète unité SpecINT, qui était devenue inadéquate à représenter les performances du code de la Physique des Hautes Énergies sur les nouvelles architectures des unités de calcul. HEP-SPEC06 est basé sur un sous-ensemble de référence (all_cpp, bset) du standard industriel SPEC CPU2006 qui reproduit avec plus de fidélité que le sous-ensemble entier (SPECint) le pourcentage d'opérations en virgule flottante observées dans le code de la Physique des Hautes Énergies (environ 10 %). HEP-SPEC06 a été adopté officiellement par WLCG en Avril 2009.

ouverture des fichiers, taille des répertoires de travail, utilisation des disques et des processeurs.

L'exigence qui nous a amenés à introduire MonALISA a été de pouvoir évaluer la performance globale du système et déboguer rapidement les problèmes et les situations critiques. Au fur et à mesure de l'utilisation de MonALISA, nous avons réalisé que nous aurions pu exploiter davantage l'énorme quantité d'information détaillée dont nous disposons. Sur la base des informations reçues de la surveillance des sites, le système peut prendre diverses actions automatisées pour faire face aux événements, qui vont de l'envoi d'un message par courriel ou SMS à l'exécution d'une série plus ou moins complexe d'actions automatiques visant à restaurer la fonctionnalité du système. Un autre type d'action automatisée est le maintien de l'équilibrage de charge des services centraux, basé sur leur charge, à partir du DNS.

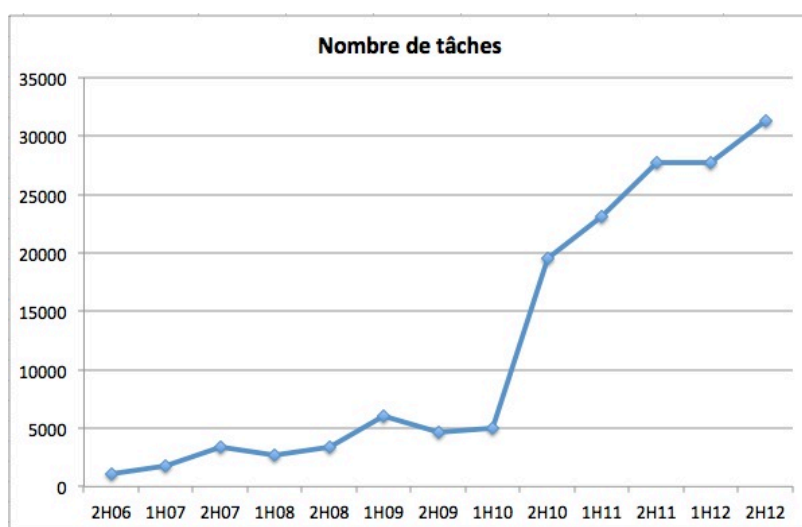


Figure 77 : Évolution du nombre moyen de tâches en exécution sur la Grille de ALICE.

L'intégration progressive de MonALISA dans AliEn nous a permis de l'utiliser aussi pour gérer les productions de données MonteCarlo. En surveillant continuellement le nombre de tâches en attente dans le système, lorsque le nombre descend en dessous d'un certain seuil, de nouvelles tâches sont automatiquement envoyées au gestionnaire des ressources de calcul de AliEn, selon un schéma de priorités prédéterminées. Les tâches qui ont échoué sont automatiquement réintroduites pour être exécutées.

Les services centraux sont continuellement surveillés, et si l'un d'eux s'arrête il est automatiquement redémarré, et une opération de vérification de l'état des serveurs est démarrée aussi, en parallèle aux notifications envoyés aux experts, de telle façon que, de leur intervention, ils auront à disposition le maximum d'informations sur le dysfonctionnement.

Bien que toutes ces mesures aient beaucoup amélioré la stabilité et la disponibilité globale du système, nous pensons que nous sommes au début de ce que nous pouvons faire avec une utilisation intelligente de l'information contenue dans MonALISA. D'un côté, nous pouvons pousser l'automatisation de la réponse aux situations critiques plus loin, de façon à minimiser l'intervention

humaine et à garantir davantage de stabilité et disponibilité du système. Mais nous pouvons aussi utiliser cette information pour optimiser la distribution des tâches et des fichiers dans la Grille pour s'adapter automatiquement aux conditions de charge de travail et de configuration du système. C'est un travail extrêmement intéressant du point de vue « intellectuel » et qui, à part les bienfaits pratiques sur l'efficacité du système informatique distribué de ALICE, pourrait aussi éclairer le fonctionnement de la Grille vue comme système complexe et distribué, qui est l'un des paradigmes émergents de notre époque.

5.2.4 L'évolution future de la Grille

En dehors de la Physique des Hautes Énergies, la Grille n'a pas eu le succès auquel elle semblait promise. Les autres sciences n'ont pas encore le besoin désespéré de ressources de calcul gigantesques qui caractérise la Physique des Hautes Énergies, et aucun modèle commercial valide n'a été élaboré à partir de la vision de la Grille.

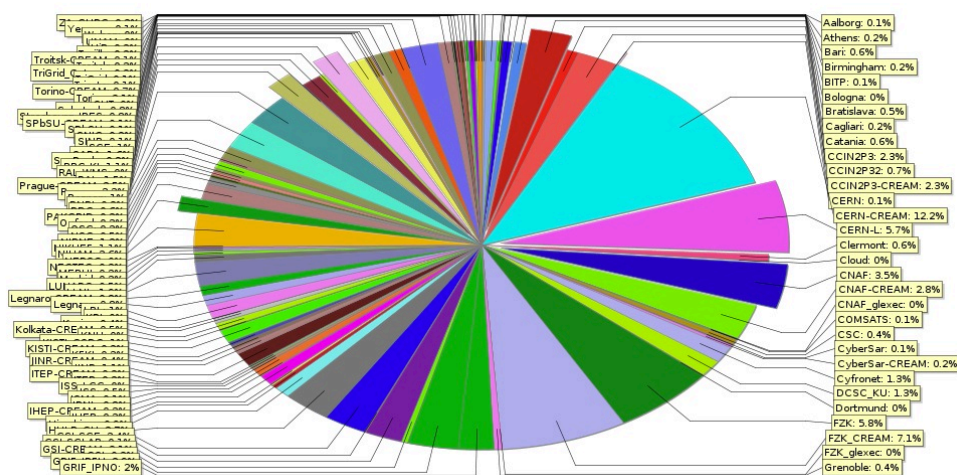


Figure 78 : Contribution des différents centres de calcul à la Grille de ALICE en termes de temps de calcul fourni.

Bon nombre des concepts et des techniques utilisés pour construire la Grille sont entrés dans l'élaboration du Nuage (en anglais *Cloud*), qui est conçu comme un service commercial privé (voir par exemple le Cloud Amazon [156]). La Physique des Hautes Énergies ne semble toutefois pas en mesure de pouvoir exploiter immédiatement le paradigme du Nuage, qui, en particulier, ne fournit pas une solution toute faite pour le problème posé par le traitement de grandes quantités de données. Le concept du Nuage consiste à offrir des services informatiques à une communauté distribuée, mais les ressources n'ont pas besoin d'être elles-mêmes distribuées et elles sont fournies à titre de « service ».

Si une organisation a accès à plusieurs services de Nuage éventuellement fournis par des fournisseurs différents dans des endroits différents, le problème de voir cet ensemble de services comme un service cohérent n'a pas encore de solution technique définitive. Ce problème reste ouvert, surtout si nous commençons à examiner les différents types de services disponibles (stockage, puissance de calcul, bande passante du réseau), et le fait qu'il devrait y avoir une couche de

services qui leur permette de travailler ensemble de manière cohérente. Une vision possible est que la Grille pourrait encore jouer un rôle comme technologie pour proposer et coordonner les services Cloud pas nécessairement distribués. Dans cette architecture « verticale », la Grille se verrait « poussée » au niveau de la mise en œuvre de l'accès universel à l'infrastructure informatique, plus ou moins distribuée, tandis que le Nuage serait l'ensemble des services et le modèle qui donnerait accès à ces ressources. Un autre élément de distinction est que le calcul de Nuage offre des services à la fois interactifs et par lots, alors que l'approche de Grille se limite jusqu'à présent à gérer un traitement par lots « classiques » de lourdes charges de calcul.

Disk storage elements																			
SE Name	AliEn SE	Size	Used	Statistics	Usage	No. of files	Type	Size	Used	Free	Usage	Version	add	get	Last OK test	Successful	Failed	Demotion	
1. Bari - SE	ALICE:Bari::SE	400 TB	290.5 TB	109.5 TB	72.63%	4,023,501	FILE	4.014 PB	3.7 PB	321.4 TB	94.1%	20100510-1509_400	Subr...		30.10.2012 14:01	10	0	0.588%	
2. Birmingham - SE	ALICE:Birmingham::SE	54.6 TB	0	54.6 TB	-	0	FILE	54.57 TB	7.425 TB	47.15 TB	13.61%	v3.1.0	Subr...		30.10.2012 14:01	10	0	0.588%	
3. BITP - SE	ALICE:BITP::SE	26.91 TB	13.05 TB	13.86 TB	48.49%	406,067	FILE	26.91 TB	15.22 TB	11.69 TB	56.56%	v3.2.2	Subr...		30.10.2012 14:01	10	0	6.765%	
4. Bologna - SE	ALICE:Bologna::SE	6.26 TB	647.2 GB	5.628 TB	10.1%	8,639	FILE	-	-	-	-	-	Subr...		30.10.2012 14:04	11	1	8.603%	
5. Bratislava - SE	ALICE:Bratislava::SE	151 TB	131 TB	20.04 TB	86.73%	2,641,805	FILE	151 TB	147 TB	3.927 TB	97.4%	v3.2.2	Subr...		30.10.2012 14:02	10	0	0.294%	
6. Catania - SE	ALICE:Catania::SE	249 TB	202.9 TB	46.13 TB	81.49%	3,321,569	FILE	994.8 TB	917 TB	77.73 TB	91.1%	v3.2.4	Subr...		30.10.2012 14:01	10	0	8.529%	
7. CCIN2P3 - SE	ALICE:CCIN2P3::SE	600.2 TB	592.6 TB	7.55 TB	98.77%	8,304,056	FILE	600.2 TB	588.7 TB	11.52 TB	98.1%	20110519-6d6ec62_600	Subr...		28.10.2012 06:03	10	0	0.000%	
8. CERN - ALICECDISK	ALICE:CERN:ALICECDISK	20.78 TB	18.66 TB	2.12 TB	88.54%	282,865	CASTOR	2.292 PB	1.444 PB	868.5 TB	63%	-	Subr...		30.10.2012 14:04	10	2	13.09%	
9. CERN - EOS	ALICE:CERN:EOS	543 TB	456.2 TB	86.82 TB	84.01%	11,474,816	FILE	2.618 PB	2.022 PB	610 TB	77.24%	-	Subr...		30.10.2012 14:00	10	2	13.09%	
10. CERN - OCDB	ALICE:CERN:OCDB	20 TB	0	20 TB	-	0	FILE	-	-	-	-	-	Subr...		30.10.2012 14:05	10	2	13.09%	
11. Clermont - SE	ALICE:Clermont::SE	179.9 TB	165.9 TB	14 TB	92.7%	2,998,798	FILE	-	-	-	-	-	Subr...		26.10.2012 10:04	0	18	17.73%	
12. CNAP - SE	ALICE:CNAP::SE	989.7 TB	999.3 TB	-	100%	12,336,499	FILE	989.7 TB	969.9 TB	19.79 TB	98%	v3.1.0	Subr...		28.10.2012 20:03	0	0	0.000%	
13. CyberSar, Cagliari - SE	ALICE:CyberSar_Cagliari::SE	53.15 TB	18.06 TB	34.49 TB	35.12%	709,440	FILE	53.15 TB	51.37 TB	1.781 TB	96.4%	v3.1.0	Subr...		30.10.2012 14:05	10	0	3.294%	
14. Cyfronet - SE	ALICE:Cyfronet::SE	10 TB	12.06 TB	-	120.6%	283,050	FILE	-	-	-	-	-	Subr...		23.06.2012 08:02	0	18	180%	
15. Cyfronet - XRD	ALICE:Cyfronet:XRD	25.52 TB	0	25.52 TB	-	0	FILE	25.52 TB	22.64 TB	2.877 TB	89.1%	v3.2.2	Subr...		30.10.2012 14:02	10	0	0.000%	
16. FZK - SE	ALICE:FZK::SE	179.9 TB	13.44 TB	166.46 TB	64.04%	13,965,302	FILE	1.063 PB	922.8 TB	165.8 TB	84.17%	20100510-1509_400	Subr...		30.10.2012 14:01	10	0	0.000%	
17. Grenoble - SE	ALICE:Grenoble::SE	82.55 TB	29.93 TB	52.58 TB	36.27%	482,201	FILE	82.51 TB	80.03 TB	1.48 TB	97.8%	20100510-1509_400	Subr...		26.10.2012 14:00	10	0	14.11%	
18. GRIF, IPNO - SE	ALICE:GRIF_IPNO::SE	223.3 TB	213.8 TB	9.542 TB	95.7%	3,514,199	FILE	223.3 TB	218.7 TB	4.583 TB	97.8%	20100510-1509_400	Subr...		25.10.2012 12:00	0	18	93.84%	
19. GRIF, INFU - DPM	ALICE:GRIF_INFU::DPM	171 TB	41 TB	130 TB	23.97%	350,561	SRM	-	-	-	-	-	Subr...		30.10.2012 14:04	10	0	0.000%	
20. GSI - SE	ALICE:GSI::SE	111 TB	290.2 TB	-	261.4%	3,584,818	FILE	-	-	-	-	-	Subr...	Last...	17.09.2012 02:02	0	18	180%	
21. GSI - SE2	ALICE:GSI:SE2	28 TB	327.8 GB	27.68 TB	1.143%	5,060	FILE	100.7 GB	92.29 GB	8.453 GB	90.8%	v3.1.0	Subr...	Last...	18.09.2012 18:04	0	18	180%	
22. HHU, GU - SE	ALICE:HHU_GU::SE	100 TB	51.19 TB	48.81 TB	51.19%	1,130,683	FILE	-	-	-	-	-	Subr...	Last...	23.10.2012 20:03	0	18	180%	
23. Hiroshima - SE	ALICE:Hiroshima::SE	118.2 TB	79.54 TB	38.66 TB	67.3%	2,167,452	FILE	-	-	-	-	-	Subr...	Last...	30.10.2012 14:00	10	0	0.000%	
24. IHEP - SE	ALICE:IHEP::SE	36.38 TB	26.17 TB	7.211 TB	80.18%	1,059,457	FILE	72.76 TB	31.67 TB	41.09 TB	43.53%	v3.2.4	Subr...		30.10.2012 14:04	10	0	0.000%	
25. IPNL - SE	ALICE:IPNL::SE	37.3 TB	54.48 TB	-	146.1%	866,741	FILE	37.3 TB	36.57 TB	748.9 GB	98.0%	20100510-1509_400	Subr...		26.10.2012 08:04	0	18	180%	
26. ISMA - SE	ALICE:ISMA::SE	9.275 TB	0	9.275 TB	-	0	FILE	9.275 TB	958.7 GB	8.339 TB	10.09%	v3.2.2	Subr...		30.10.2012 14:05	10	0	0.000%	
27. ISS - FILE	ALICE:ISS::FILE	140.5 TB	138.1 TB	2.35 TB	98.2%	3,175,992	FILE	140.5 TB	137.6 TB	2.896 TB	98.6%	v3.2.4	Subr...		26.10.2012 10:04	10	0	3.634%	
28. ITEP - SE	ALICE:ITEP::SE	140 TB	103.1 TB	36.85 TB	73.66%	2,016,368	FILE	139.9 TB	99.25 TB	40.64 TB	70.95%	20100510-1509_400	Subr...		30.10.2012 14:00	10	0	0.000%	
29. JINR - SE	ALICE:JINR::SE	308.9 TB	247.8 TB	61.11 TB	80.22%	5,597,064	FILE	308.9 TB	298.5 TB	10.45 TB	96.6%	20100510-1509_400	Subr...		30.10.2012 14:01	10	0	0.000%	
30. KFKI - SE	ALICE:KFKI::SE	73 TB	27.76 TB	45.24 TB	38.03%	425,280	FILE	72.75 TB	66.25 TB	6.509 TB	91.2%	20100510-1509_400	Subr...		30.10.2012 14:04	10	0	0.000%	
31. KISTL_GSDC - SE	ALICE:KISTL_GSDC::SE	100 TB	61.62 TB	38.38 TB	61.62%	1,424,411	FILE	101.8 TB	99.74 TB	2.032 TB	98%	20100510-1509_400	Subr...		28.10.2012 20:02	10	0	36.43%	
32. KISTL_GSDC - SE2	ALICE:KISTL_GSDC:SE2	966.8 TB	73.81 TB	893 TB	7.634%	1,914,528	FILE	966.8 TB	95.9 TB	870.9 TB	9.919%	20100510-1509_400	Subr...		30.10.2012 14:03	10	0	0.000%	
33. KISTI - SE	ALICE:KISTI::SE	49.95 TB	52.56 TB	-	105.2%	1,474,436	FILE	0	0	0	0	20100510-1509_400	Subr...		26.10.2012 10:02	0	18	18.91%	
34. Kolkata - SE	ALICE:Kolkata::SE	223.2 TB	53.01 TB	170.2 TB	23.75%	1,497,242	FILE	186.1 TB	77.31 TB	108.8 TB	41.55%	20100510-1509_400	Subr...	Last...	30.10.2012 14:06	8	4	28.82%	
35. Kosice - SE	ALICE:Kosice::SE	60 TB	55.05 TB	4.951 TB	93.42%	1,496,588	FILE	60.03 TB	58.81 TB	1.217 TB	97.7%	20100510-1509_400	Subr...		30.10.2012 14:02	10	0	0.254%	
36. LBL - SE	ALICE:LBL::SE	756.3 TB	288.7 TB	467.4 TB	38.44%	7,747,118	FILE	716.1 TB	338.3 TB	377.8 TB	47.24%	v3.0.2_099	Subr...		30.10.2012 14:01	10	0	0.000%	
37. Legnaro - SE	ALICE:Legnaro::SE	385.6 TB	312.3 TB	73.27 TB	81%	6,979,073	FILE	385.6 TB	377.4 TB	8.245 TB	97.8%	v3.1.0	Subr...		30.10.2012 14:03	10	0	7.647%	
38. LLNL - SE	ALICE:LLNL::SE	688 TB	229.9 TB	458.1 TB	33.41%	5,660,839	FILE	687.8 TB	344.7 TB	343.1 TB	50.12%	20100510-1509_400	Subr...		30.10.2012 14:00	10	0	0.000%	
39. Madrid - SE	ALICE:Madrid::SE	36.6 TB	27.66 TB	8.938 TB	75.58%	915,333	FILE	-	-	-	-	-	Subr...		30.10.2012 14:01	10	0	0.000%	
40. NDGF - DCACHE	ALICE:NDGF:DCACHE	873 TB	315.3 TB	557.7 TB	36.09%	3,010,269	SRM	-	-	-	-	-	Subr...		30.10.2012 14:03	10	0	0.000%	
41. NETCC - SE	ALICE:NETCC::SE	30 TB	13.61 TB	16.39 TB	45.04%	418,775	FILE	30 TB	18.82 TB	11.18 TB	62.73%	v3.1.0	Subr...		30.10.2012 14:05	10	0	3.294%	
42. NIHAM - FILE	ALICE:NIHAM:FILE	995 TB	523.8 TB	371.2 TB	52.53%	14,314,460	FILE	-	-	-	-	-	Subr...		05.07.2012 08:00	0	18	180%	
43. OSC - SE	ALICE:OSC::SE	30 TB	0	30 TB	-	0	FILE	-	-	-	-	-	Subr...		-	-	-	180%	
44. PNPI - SE	ALICE:PNPI::SE	40 TB	32.03 TB	7.97 TB	80.08%	712,357	FILE	39.95 TB	33.19 TB	6.765 TB	84.0%	20100510-1509_400	Subr...		30.10.2012 14:02	10	0	0.000%	
45. Poznan - SE	ALICE:Poznan::SE	197.3 TB	45.54 TB	151.8 TB	23.09%	946,259	FILE	0	0	0	0	20100510-1509_400	Subr...	Last...	20.10.2012 08:02	0	18	180%	
46. Prague - SE	ALICE:Prague::SE	136.7 TB	369.7 TB	-	269.9%	4,733,608	FILE	488.3 TB	477.9 TB	10.4 TB	97.8%	v3.2.2	Subr...		13.10.2012 18:01	0	18	180%	
47. RAL - SE	ALICE:RAL::SE	188 TB	117.2 TB	70.79 TB	62.35%	1,377,822	CASTOR	-	-	-	-	-	Subr...		30.10.2012 09:04	1	18	84.04%	
48. RARC-I - SE	ALICE:RARC-I::SE	319.8 TB	251.1 TB	68.73 TB	78.51%	4,429,631	FILE	339.1 TB	332.2 TB	6.872 TB	97.6%	20100510-1509_400	Subr...		30.10.2012 14:04	10	0	0.000%	
49. SaoPaulo - SE	ALICE:SaoPaulo::SE	51 TB	10.6 TB	40.2 TB	21.19%	262,129	FILE	-	-	-	-	-	Subr...	Last...	22.10.2012 12:01	0	18	180%	
50. SARA - DCACHE	ALICE:SARA:DCACHE	200 TB	189.5 TB	10.5 TB	94.75%	1,988,433	SRM	-	-	-	-	-	Subr...	Last...	26.08.2011 08:01	0	18	180%	
51. SNIC - DCACHE	ALICE:SNIC:DCACHE	400 TB	0	400 TB	-	0	FILE	-	-	-	-	-	Subr...	Last...	30.10.2012 14:04	10	0	0.000%	
52. SPFSU - SE	ALICE:SPFSU::SE	61.7 TB	25.88 TB	35.82 TB	41.95%	680,890	FILE	-	-	-	-	-	Subr...	Last...	30.10.2012 14:02	10	0	0.000%	
53. Strasbourg_IRES - SE	ALICE:Strasbourg_IRES::SE	52.83 TB	66.88 TB	-	126.6%	1,313,237	FILE	52.83 TB	51.81 TB	1.017 TB	98.4%	20100510-1509_400	Subr...		26.10.2012 10:00	0	18	90.23%	
54. Subatech - SE	ALICE:Subatech::SE	313.3 TB	315.5 TB	-	100.7%	6,218,195	FILE	313.3 TB	307.5 TB	5.785 TB	98.4%	v3.1.0	Subr...		26.10.2012 08:00	0	18	90.23%	
55. SUT - SE	ALICE:SUT::SE	30 TB	10.15 GB	29.95 TB	0.323%	1,094	FILE	57.8 GB	-	29.97 TB	-	v3.1.0	Subr...		30.10.2012 14:04	10	0	3.294%	
56. Torino - SE	ALICE:Torino::SE	498.1 TB	242.1 TB	256 TB	48.6%	4,387,893	FILE	-	0	0	0	20100510-1509_400	Subr...		30.10.2012 14:03	10	0	0.000%	
57. Trieste - SE	ALICE:Trieste::SE	251 TB	21.87 TB	229.1 TB	8.713%	462,549	FILE	251 TB	187.4 TB	63.67 TB	24.64%	20100510-1509_400	Subr...	Last...	26.10.2012 04:02	0	18	180%	
58. Troitsk - SE	ALICE:Troitsk::SE	94.49 TB	79.1 TB	15.39 TB	83.71%	2,315,700	FILE	94.49 TB	90.56 TB	3.935 TB	97.4%	20100510-1509_400	Subr...		30.10.2012 14:02	10	0	0.000%	
59. Trujillo - SE	ALICE:Trujillo::SE	67.64 TB	26.09 TB	41.55 TB	38.57%	465,106	FILE	-	-	-	-	-	Subr...	Last...	09.08.2012 12:03	10	2	16.85%	
60. ZA_CMS - SE	ALICE:ZA_CMS::SE	44.58 TB	0	44.58 TB	-	0	FILE	44.58 TB	114.4 GB	44.58 TB	0.251%	v3.2.4	Subr...		30.10.2012 14:02	10	2	16.85%	
Total		18,524 PB	10,86 PB	6,41 PB		190,328,274		18,524 PB	14,49 PB	4,048 PB									

quelle sera son interaction avec les nouvelles normes et modèles qui apparaissent dans le monde du e-commerce. En outre, la Grille que nous connaissons en Physique des Hautes Énergies a été adaptée aux besoins et à la sociologie de cette discipline, et peut-être même de l'organisation particulière des expériences du LHC. Encore une fois, nous devons nous demander comment cette infrastructure doit évoluer pour être durable à long terme comme une véritable plate-forme de calcul multi-sciences.

Le futur de AliEn dépendra du futur de la Grille dans la Physique des Hautes Énergies. Une des directions de développement consiste à minimiser la configuration nécessaire pour le système en développant les capacités de découverte et de configuration autonome des composants de AliEn. Comme nous l'avons vu plus haut, une étroite intégration avec MonALISA peut permettre la récupération des erreurs à travers la reconfiguration des services et leur redémarrage, de façon à minimiser la nécessité d'une intervention humaine et en réduire l'urgence, dans le sens que le système doit continuer à marcher, même de façon sub-optimale en attendant l'intervention des experts.

Tape storage elements																
SE Name	AliEn SE	Statistics			No. of files	Type	Xrootd info			Version	Functional tests		Last OK test		Last day tests	
		Size	Used	Free			Free	Usage	Usage		add	get	Successful	Failed	Demotion	factor
1. CCINP3 - TAPE	ALICE:CCINP3:TAPE	275 TB	725 TB	-	471,946	FILE	272.8 TB	238.7 TB	34.06 TB	87.51%	v20110519-dcac3_0bq		30.10.2012 14:04	12	0	0
2. CERN - CASTOR2	ALICE:CERN:CASTOR2	2,907 PB	3,611 PB	-	15,116,888	CASTOR	-	-	-	-			30.10.2012 14:03	12	0	0
3. CERN - TDAlice	ALICE:CERN:TDAlice	2,907 PB	503.5 TB	2,315 PB	17,511	FILE	608.639	2,691 PB	695.4 TB	74.77%			30.10.2012 14:02	12	0	0
4. CNAF - TAPE	ALICE:CNAF:TAPE	349 TB	555.7 TB	-	345,664	FILE	232.9 TB	209.2 TB	23.63 TB	84.66%	v3.1.0		30.10.2012 14:05	12	0	0
5. FZK - TAPE	ALICE:FZK:TAPE	1,074 PB	2,342 PB	-	1,281,472	FILE	640.4 TB	542.6 TB	97.84 TB	84.72%	20110516-1009_0bq		30.10.2012 14:02	12	0	0
6. NDGF - DCACHE_TAPE	ALICE:NDGF:DCACHE_TAPE	93.13 TB	145.6 TB	-	103,054	SRM	-	-	-	-			30.10.2012 14:02	12	0	0
7. RAL - TAPE	ALICE:RAL:TAPE	326 TB	244 TB	-	152,888	CASTOR	-	-	-	-			30.10.2012 14:02	12	0	0
8. RAL - TAPE TEST	ALICE:RAL:TAPE TEST	30.38 GB	0	-	0	CASTOR	-	-	-	-			30.10.2012 14:02	12	0	0
9. SARA - DCACHE_TAPE	ALICE:SARA:DCACHE_TAPE	492 TB	22.78 TB	469.2 TB	10,643	SRM	-	-	-	-			13.09.2011 08:04	0	10	0.544%
Total		8,185 PB	13.3 PB	2,854 PB	18,090,568		3,81 PB	2,979 PB	850.9 TB							

Figure 80 : Éléments de stockage sur bande dans la Grille de ALICE.

Un deuxième axe de développement est la possibilité pour AliEn de fonctionner de façon « opportuniste ». Une partie du personnel des centres de calcul de WLCG a été financée par les différents projets de Grille (EDG, EGEE, EMI-EGI) qui maintenant sont arrivés à échéance. Il est donc important de comprendre quelles sont les demandes « minimales » aux centres de calcul en termes de services, support et installation pour pouvoir continuer à utiliser leurs ressources, car leur personnel sera très probablement réduit. Dans le passé, nous avons été contactés plusieurs fois par des centres de calcul prêts à nous fournir des ressources de calcul, mais qui considéraient l'installation d'une VOBox comme trop contraignante. Donc, l'idée serait de réduire autant que faire se peut les demandes pour le site et d'élaborer une stratégie où les ressources d'un site peuvent aller et venir sans dommage pour le traitement des données, mais peuvent être utilisées efficacement. Évidemment cela n'est pas possible pour l'ensemble de la Grille, car il est nécessaire d'avoir des centres pour stocker les données de façon fiable pour le long terme. Néanmoins nous pourrions imaginer une situation où l'intergiciel de Grille est installé de façon stable, avec des services permanents, seulement sur les centres Tier 1. Tout le reste pourrait être constitué de ressources de calcul « opportunistes ». L'utilisation des tâches pilotes et de BitTorrent pour distribuer le logiciel est déjà un pas dans cette direction, car nous pourrions imaginer que la tâche pilote pourrait récupérer aussi le logiciel de AliEn et tourner des services temporairement.

Le problème est plus compliqué avec le stockage, car l'utilisation d'espace de disque de façon opportuniste semble être moins évidente. Bien que nous n'ayons

pas de solution complètement élaborée, une piste que nous sommes en train d'explorer est l'utilisation de la fonctionnalité de xRootd. Il faudrait pouvoir installer un serveur xRootd « au vol » et utiliser l'espace de disque local comme un cache. Chaque fichier sera donc écrit à la fois sur des éléments de stockage « volatile » ou « opportuniste » locaux ou éloignés et sur un élément de stockage « classique » de gardiennage dans un Tier 1. Seule la réplique « de gardiennage » serait enregistrée dans le catalogue, qui se limitera à cataloguer les copies sur les Tier 1s. La localisation des répliques sera gardée dans un redirecteur global xRootd dynamique, qui travaillera de concert avec le catalogue des fichiers pour fournir au courtier des tâches les informations sur l'emplacement des fichiers. La réplification des fichiers sur les éléments de stockage « temporaires » pourra être gérée par xRootd lui-même, selon la charge et l'occupation des disques. Cette fonctionnalité est présente dans xRootd, même si elle doit probablement être développée et connectée avec les informations provenant de MonALISA. Si l'un des éléments de stockage est arrêté, même sans préavis, les catalogues en mémoire de xRootd seront reconfigurés automatiquement et de nouvelles répliques créées par xRootd. En tous cas, une copie est toujours gardée dans un Tier 1 et cataloguée dans le catalogue central des fichiers. Cette ébauche de stratégie doit évidemment être développée et testée en pratique, mais les éléments existent déjà et il semble possible de les agencer pour obtenir un tel fonctionnement.

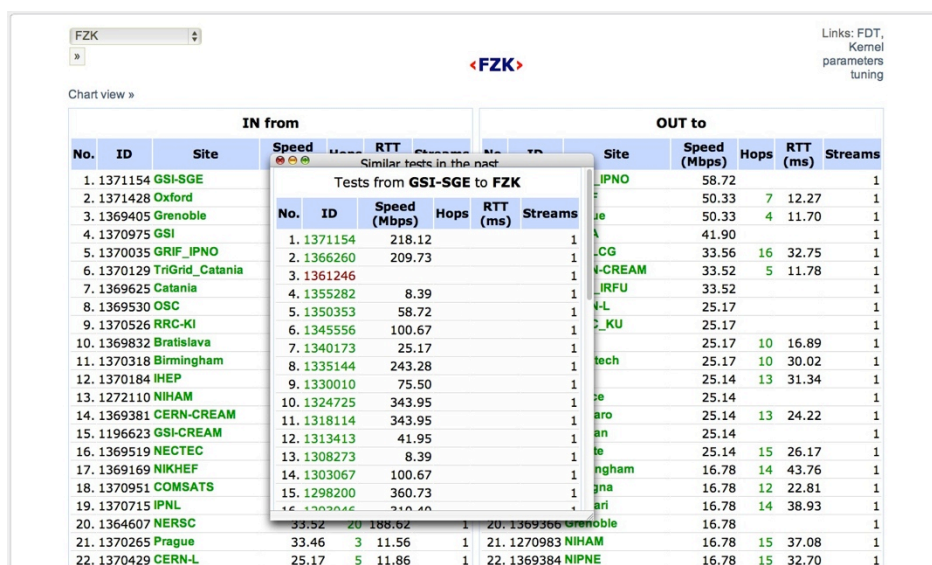


Figure 81 : Aperçu des essais de performance exécutés par MonALISA entre chaque paire de VOBoxes avec l'historique.

Minimiser la configuration manuelle et réduire les demandes de services requis pour les centres de calcul sont autant d'étapes nécessaires pour pouvoir effectuer une transition entre les services de Grille et les services de Nuage, en suivant la tendance générale du monde de la technologie de l'information. Nous pouvons voir dans la Figure 82 l'évolution de l'occurrence des références à la Grille et au Nuage sur la toile. Le nœud principal de cette transition est en effet la gestion du stockage, et comme cela est déjà arrivé pour la Grille, la Physique des Hautes Énergies doit maintenant élaborer le concept de Nuage de Données (en

anglais *Data Cloud*) pour pouvoir exploiter les opportunités qui s'offrent à nous en termes de ressources de calcul.

5.3 Utilisation de la Grille de ALICE

La Grille de ALICE est de facto l'instrument de calcul qui permet à la collaboration de traiter les données brutes provenant du détecteur et de produire les résultats publiés dans les revues scientifiques, aboutissement du travail des scientifiques.

Depuis le début de la prise de données, ALICE a récolté presque 7 Po de données brutes (voir Figure 76) pour un total de presque 5 millions de fichiers. Tous ces fichiers sont stockés au CERN et ceux correspondant à des prises de données de physique sont aussi répliqués sur un des Tier 1s. Ces fichiers contiennent les résultats de toutes les périodes de prise de données, c'est-à-dire rayons cosmiques, prises de données de calibrage, collisions proton-proton, Plomb-Plomb et proton-Plomb.

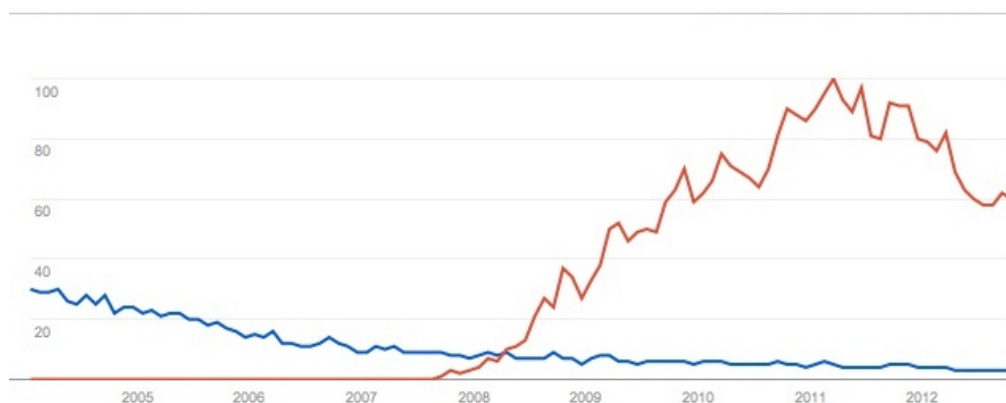


Figure 82 : Évolution des références à Grille (en bleu) et Nuage (en rouge) sur la toile depuis 2005.

La Figure 77 montre l'évolution du nombre moyen des tâches en exécution simultanée sur la Grille de ALICE. Depuis 2006 la Grille de ALICE a fourni 381,5 millions d'heures de temps de calcul, ce qui correspond à une machine travaillant pendant 43.600 années. La Grille de ALICE est constituée de six Tier 1s et environ 80 Tier 2s. La Figure 78 montre la contribution des différents centres à la Grille de ALICE. Il est important de noter que 50 % des ressources sont fournies par les centres Tier 2. C'est remarquable car cela démontre l'efficacité de la Grille de ALICE à fédérer et utiliser même des centres de taille moyenne et petite, ce qui est un des objectifs de la « vision de la Grille » formulée par ses inventeurs.

Pour ce qui est des utilisateurs, pendant la dernière année 27 % des ressources ont été utilisées pour l'analyse par les utilisateurs, 14 % pour la reconstruction des données brutes, 52 % pour la simulation et la reconstruction des données simulées et 7 % pour les trains d'analyse organisée. L'analyse organisée est maintenant fortement encouragée, car elle est plus efficace, grâce à l'utilisation de trains, que l'analyse par les utilisateurs isolés.

La Grille de ALICE dispose de 17 Po de stockage et 14 Po sont utilisés. Pour l'opération de la Grille il est très important d'avoir une marge d'espace sur disque. L'état et la fonctionnalité de chacun des 60 éléments de stockage de ALICE sont continuellement surveillés par MonALISA, comme indiqué dans la Figure 79. La Figure 80 montre la page de surveillance pour les neuf éléments de stockage sur bande dans les Tier 1 et le Tier 0 de ALICE. La pénurie d'espace sur disque nous amène à faire des nettoyages réguliers des fichiers. C'est une procédure assez compliquée, car il faut obtenir le consensus de tous les utilisateurs potentiels de ces fichiers avant de les effacer des disques, tout en gardant une copie sur bande magnétique. Pour augmenter l'efficacité des tâches d'analyse, les utilisateurs sont encouragés à utiliser plutôt les AODs que les ESDs pour leur travail. Du point de vue opérationnel, cette migration est encouragée par une réduction du nombre des répliques des ESDs de trois à deux, ce qui rend l'accès plus difficile et donc le délai d'exécution des tâches qui utilisent les ESDs plus long.

Le réseau est continuellement surveillé par MonALISA, qui effectue en continu des essais de performance entre chaque paire de VOBox. Un exemple est montré dans la Figure 81. L'optimisation de la performance du réseau est une activité continue pour le Core Offline de ALICE en collaboration avec les experts de réseau du CERN et des centres concernés.

L'opération de la Grille de ALICE est assurée par une personne travaillant à 50 % dans chaque centre de calcul en moyenne et quatre ou cinq professionnels dans le groupe Core Offline au CERN, avec le support du département d'informatique du CERN. On peut considérer cela comme une exploitation assez efficace, qui est possible uniquement grâce au niveau élevé d'automatisation de la Grille de ALICE obtenu par l'étroite collaboration du système de contrôle avec l'intergiciel de AliEn.

5.4 Conclusions et perspectives

Nous pouvons dire que la Grille a été un succès technologique, scientifique et humain. Présentée il y a quinze ans comme une « vision d'avenir », elle est maintenant l'outil de référence de la communauté des physiciens de LHC. La Grille permet un accès ubiquitaire et « démocratique » aux ressources de calcul et donc une meilleure participation des chercheurs de tous pays à l'analyse des données. Mais le résultat le plus important de la Grille, selon nous, n'est pas dans la globalisation des ressources, ni même dans la globalisation de l'accès, même si cela est une première dans le calcul scientifique, particulièrement sur cette échelle.

Le résultat le plus important est, selon nous, la possibilité qu'a donnée la Grille à des chercheurs dans des pays numériquement défavorisés⁴⁴ de participer pleinement aux développements de la physique et de l'informatique les plus avancés, avec un effet de « désenclavage » sans précédent. Nous avons vérifié plusieurs fois que l'installation d'un nœud de Grille en lui-même est un exercice qui permet de s'intégrer dans la communauté du point de vue technique et

⁴⁴ C'est-à-dire avec un accès réduit à du matériel informatique haut de gamme et à des réseaux à haute performance.

scientifique, au-delà de la quantité des ressources de calcul effectivement mises à disposition.

Une preuve de cette globalisation est le développement de AliEn lui-même, qui, grâce à l'installation des centres de la Grille de ALICE, a suscité un intérêt certain de la part de développeurs de la Grille. En particulier nous collaborons aux développements suivants :

- Utilisation opportuniste des ressources et contrôle de la mémoire par les tâches pilotes (NERSC, USA) ;
- Communications entre différentes Grilles de AliEn (Université de Glasgow, UK) ;
- Développement de l'élément de calcul CREAM (Université de Bologne, Italie) ;
- Modèle de confiance (en anglais *Trust Model*, Université de Talca, Chile et Université de Wuhan, Chine) ;
- Interface avec Oracle (GSI, Allemagne) ;
- Fonctionnalité XML, installation du logiciel (Université d'Erevan, Arménie) ;
- Optimisation de l'effacement des fichiers (Calcutta, Inde) ;
- Développement du système des quotas (KISTI centre, République de Corée) ;

Bien que la grille WLCG ait rempli son rôle en fournissant une plateforme stable pour traiter les données provenant du LHC, nous ne pouvons pas affirmer que la « vision de la Grille » ait été entièrement réalisée. Plusieurs éléments ont encore besoin d'interventions manuelles importantes pour fonctionner et nous sommes essentiellement au début du processus d'apprentissage pour optimiser l'utilisation d'un ensemble de ressources en réseau, distribuées dans le monde entier. Quelques-uns des principaux domaines qui nécessiteront attention sont les suivants:

Réseau. Les experts réseau ont préconisé depuis longtemps la nécessité de traiter le réseau comme une ressource qui doit être provisionnée et optimisée en collaboration avec les autres ressources. Le développement d'outils efficaces et des interfaces d'application pour transférer de grandes quantités de données sur réseau étendu est essentiel pour les applications intensives de données utilisées sur les grandes installations de calcul distribué. La nécessité de cette évolution a été quelque peu masquée par le fait que, au début du LHC, les réseaux de recherche ont vu un développement au-delà de toute prévision. Cependant, ce développement ne se poursuivra pas indéfiniment et une gestion plus intelligente de la ressource réseau sera bientôt nécessaire pour faire face à l'augmentation de la demande de bande passante.

L'informatique comme un service. Les concepts de Grille et de Nuage peuvent être considérés comme un prolongement de la virtualisation des ressources informatiques, déjà présente dans les systèmes d'exploitation, à l'emplacement et la répartition de ces ressources. Le but ultime est exprimé par une vision de l'informatique comme un « service » qui peut être demandé via des interfaces

appropriées et livré selon un modèle donné à différents niveaux de « qualité de service ». C'est le modèle qui sous-tend les visions de Grille et de Nuage.

Grille interactive. L'utilisation actuelle de la Grille est principalement du travail par lots, et aucune tentative sérieuse n'a été faite pour être en mesure d'utiliser la Grille de manière interactive. Cela est surprenant, car l'exigence de cette utilisation, au moins dans la Physique des Hautes Énergies, est clairement démontrée par le succès des applications interactives telles que Root et des installations d'analyse interactive. L'extension à la Grille de l'utilisation des paradigmes interactifs est également compatible avec la vision d'une Grille en mesure de répondre avec souplesse à une demande ponctuelle de ressources à travers une allocation dynamique des composants virtuels.

La gestion des données. La gestion des données sur réseau est encore relativement primitive. Les catalogues de données globaux n'ont commencé à apparaître que récemment, et il n'est pas encore clair si cette évolution est une conséquence de l'échec dans l'utilisation des catalogues distribués ou si, tout simplement, les technologies modernes des bases de données ont rendu leurs produits suffisamment puissants pour gérer une telle base de données centrale, ce qui a dispensé les développeurs de l'intergiciel de Grille d'avoir à fournir des solutions distribuées plus complexes à mettre en œuvre. Le placement adaptif et optimisé des données en est encore à ses balbutiements et l'optimisation de l'accès aux données consiste principalement en l'envoi du travail dans un endroit « proche » de données. Il n'existe pas encore de procédures globales pour l'optimisation du placement des données pour un type donné de charge de travail. Le traitement des métadonnées est encore assez primitif et il n'existe aucune norme qui puisse être mise à profit. Récemment, il y a eu plusieurs tentatives de déplacement de la frontière entre catalogage « statique » de ressources et systèmes dynamiques qui peuvent trouver des ressources par interactions poste-à-poste (en anglais *peer-to-peer*), mais encore beaucoup de travail reste à faire dans ce domaine.

Intégration de la surveillance. Bien que de très bons systèmes de surveillance soient disponibles, il n'existe pas de solution universellement acceptée. L'utilisation des informations de surveillance pour une optimisation adaptative de la charge de travail pour la Grille et le Nuage a été tentée seulement dans des domaines très limités. Les données de surveillance sont utilisées plus pour repérer des dysfonctionnements que dans le but d'améliorer la performance globale de la Grille. Également, le suivi en temps réel des ressources du réseau (topologie, utilisation de chaque segment principal) peut être grandement amélioré, et cette information est cruciale pour le développement de services appropriés pour une gestion efficace des ressources du réseau.

Simulation de la Grille. Cela peut paraître surprenant, mais il n'y a pas de simulation de la Grille. Comme nous l'avons déjà remarqué, à une époque où la simulation est considérée comme la « troisième voie de la connaissance en plus de l'expérience et de la théorie », il semble étrange qu'un système dans lequel il n'y a pas de problèmes « de principe » pour décrire les composants de base et pour lesquels d'importantes données expérimentales sont présentes sous la forme de données de surveillance, n'ait pas encore été correctement simulé. Étant un système hautement non linéaire, une simulation prédictive serait un

outil précieux pour améliorer la compréhension de la Grille et permettre l'exploration de l'espace des paramètres sans être limité à l'expérimentation avec les systèmes existants.

Nous pouvons conclure en disant que l'aventure humaine et technologique de la Grille et du Nuage est déjà assez riche et foisonnante, mais que probablement nous ne sommes qu'au début de l'histoire. La Physique des Hautes Énergies restera vraisemblablement un environnement privilégié pour observer et influencer cette évolution.

6 AliRoot en action

Après avoir décrit le programme de physique de ALICE, son détecteur et l'infrastructure de calcul, nous proposons un survol rapide des principaux résultats de physique obtenus par elle. Rappelons les objectifs principaux du programme de physique de ALICE : 1) étendre le Modèle Standard de la physique des particules à un système complexe, dynamique et de taille finie ; 2) comprendre comment les propriétés macroscopiques de la matière émergent des lois microscopiques fondamentales de la physique des particules ; 3) étudier les propriétés du plasma de quarks et gluons, qui est l'état de la matière entre la transition de phase électrofaible ($T \sim 100$ GeV) et la transition de phase hadronique ($T \sim 170$ MeV).

Pour guider la description de quelques-uns des résultats parmi les plus significatifs obtenus par l'expérience ALICE, nous commencerons par donner une brève description de ce que nous pourrions appeler le Modèle Standard de la physique des ions lourds [157, 158]. Il s'appuie sur deux théories ou modèles, décrivant l'une les propriétés statiques de la matière nucléaire à haute température et l'autre ses propriétés dynamiques.

La Chromodynamique Quantique, la théorie de l'interaction forte, ne s'applique qu'à un système statique et permet de calculer les propriétés thermodynamiques, les propriétés dynamiques restant hors de portée de la théorie. Nous en avons décrit les principaux résultats dans le Chapitre 1.

Les théories décrivant la dynamique de la collision d'ions lourds considèrent une évolution de la collision en trois phases successives :

1. La transformation des champs de couleur, présents dans les noyaux en interaction, en matière QCD (c'est-à-dire constituée de quarks et de gluons) en cours de thermalisation. Les gluons, constituants dominants de la fonction de structure du noyau aux très petites valeurs de x (fraction de l'impulsion du noyau portée par les gluons) pertinentes aux énergies du RHIC (10^{-2}) et du LHC (10^{-3}), dans un régime de saturation (CGC, un condensat de verre de couleur, a été suggéré pour décrire cet état), sont libérés et suivent une évolution non linéaire pour former un nouvel état appelé glasma dans le cadre du modèle CGC. La dynamique de cette phase reste la moins bien maîtrisée par la théorie.
2. La thermalisation rapide du glasma conduit au QGP thermalisé ($T \approx 2-3 T_c$) dont l'expansion explosive est décrite par les équations relativistes de l'hydrodynamique. Ces équations déterminent les aspects collectifs mesurables de l'état final à partir des propriétés du fluide (équation d'état, viscosité, vitesse du son) calculables *ab initio* par la QCD sur réseau (LQCD⁴⁵) par exemple, et de conditions aux limites (géométrie, gradient de pression, fluctuations) de l'état initial.

⁴⁵ La théorie de jauge sur réseau (LQCD), une approche non perturbative pour résoudre les équations de QCD, simule la fonction de partition de QCD sur un réseau d'espace-temps à partir des principes premiers de la théorie QCD.

3. En dessous de la température critique, la condensation du QGP en hadrons conduit vers l'état final dans un état de gel chimique, dont la dynamique est décrite par les équations de transport de Boltzmann jusqu'au gel cinématique de l'état final. Le phénomène de hadronisation ayant lieu à l'échelle non perturbative n'est pas calculable et doit faire appel à des modèles phénoménologiques de fragmentation des partons.

Signalons une approche nouvelle au succès grandissant qui permet de décrire les propriétés dynamiques d'un fluide quantique non visqueux et en forte interaction. Il a été démontré que la théorie des cordes dans un espace Anti-deSitter est duale avec la théorie conforme des champs et permet une description exacte des systèmes fortement couplés. Le formalisme de cette approche, appelée AdS/CFT, identifie la théorie quantique des champs à couplage fort, non solvable, avec la théorie gravitationnelle classique à couplage faible dans cinq dimensions qui, elle, est solvable. Il a ainsi été prédit qu'il existait pour un fluide quantique relativiste une valeur limite inférieure au rapport viscosité (η) sur densité d'entropie (S), telle que $4\pi\eta/S \geq 1$ dans un système d'unité où $\hbar = k_B = 1$ [159].

Le scénario de la dynamique des collisions d'ions lourds a été validé par les données expérimentales accumulées ces dix dernières années par les expériences du RHIC et plus récemment par celles du LHC. L'ensemble de ces mêmes données converge de façon cohérente vers la prédiction de LQCD, à savoir qu'aux densités d'énergie atteintes, bien au-delà de la valeur critique de la théorie, la matière forme un état partonique thermalisé. Des mesures détaillées visant une caractérisation quantitative de cet état indiquent que ses propriétés sont celles d'un liquide (les constituants sont fortement couplés), que sa viscosité est faible, conduisant à une valeur η/S proche de la valeur limite prédite par AdS/CFT (on parle dès lors de liquide parfait), et que la matière QCD est opaque aux charges de couleur et transparente aux bosons de jauge tels photon et Z .

Durant la première période de collisions Plomb–Plomb à $\sqrt{s_{NN}} = 2,76$ TeV, d'à peine trois semaines fin 2010, ALICE a inspecté une luminosité intégrée de l'ordre de $10 \mu\text{b}^{-1}$ et accumulé de l'ordre de 10^8 événements de biais minimum. Une nouvelle période, fin 2011, avec le même système et à la même énergie, a permis d'inspecter une luminosité quinze fois supérieure et d'accumuler des événements rares triés par des déclenchements sélectifs (événements à 1 ou 2 muons, événements incluant des particules de grande impulsion transverse, événements de collisions centrales et semi-centrales).

Les résultats de l'analyse des données Plomb–Plomb enregistrées par l'expérience ALICE et des données proton–proton ont permis, dans un premier temps, de consolider les modèles décrivant l'évolution de la collision établis à partir des résultats du RHIC, et ensuite d'ouvrir de nouvelles pistes pour caractériser le QGP, notamment en exploitant les observables multi-différentielles, plus facilement accessibles aux énergies du LHC, impliquant les saveurs lourdes, les quarkonia ou encore les jets.

Les différentes analyses se sont appuyées à la fois sur les performances uniques du système de détection ALICE, à savoir la trajectographie de haute précision des particules chargées sur une large gamme d'impulsions transversales (d'une centaine de MeV/c jusqu'à une centaine de GeV/c) et l'identification des

particules à la fois trace par trace et par analyse de leur topologie de désintégration.

6.1 Propriétés globales de la collision

Les particules observées dans l'état final d'impulsion transverse inférieure à quelques GeV/c sont les témoins des propriétés globales du système formé dans la collision. L'état hadronique final, caractérisé par la multiplicité des hadrons, le spectre en impulsion des particules et l'abondance des divers types de hadrons, est le témoin de la dynamique de la collision. Le spectre de photons directs joue un rôle particulier, car il reflète l'état du système au moment de leur production.

6.1.1 Multiplicité

La densité des particules chargées mesurée à rapidité centrale dans les collisions Plomb-Plomb à $\sqrt{s_{NN}} = 2,76$ TeV est $dN/d\eta = 1600 \pm 76$ [160]. À partir de cette mesure, la densité d'énergie au moment de la formation du QGP en équilibre thermique peut être évaluée à partir d'une formule proposée par Bjorken [161] :

$$\varepsilon \geq \frac{dE_T/d\eta}{\tau_0 \pi R^2} = \frac{3}{2} \langle E_T/N \rangle \frac{dN_{ch}/d\eta}{\tau_0 \pi R^2},$$

où τ_0 représente le temps à partir duquel le système est thermalisé, R est le rayon nucléaire, et $E_T/N \approx 1$ GeV est l'énergie transverse moyenne par nucléon. La valeur de la multiplicité mesurée au LHC implique que la densité d'énergie initiale ($\tau_0 = 1$ fm/c) est d'environ 15 GeV/fm³ [162], soit environ trois fois supérieure à la valeur déduite des mesures dans les collisions Or-Or à $\sqrt{s_{NN}} = 200$ GeV de RHIC [163, 164, 165, 166].

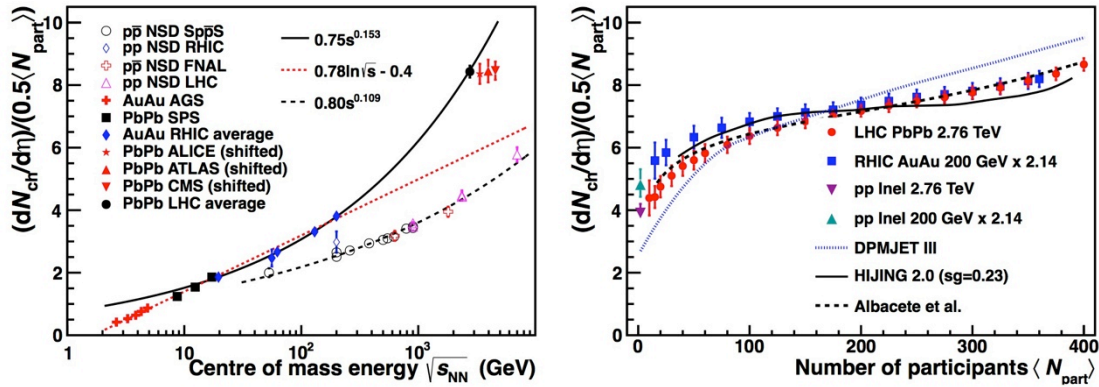


Figure 83 : Densité de particules chargées par unité de pseudorapidité et par paire de nucléons ($0,5 \langle N_{part} \rangle$) mesurée dans les collisions Plomb-Plomb et proton-proton en fonction de l'énergie du centre de masse (à gauche) et en fonction du nombre de nucléons participants (à droite), comparée avec les prédictions théoriques pour les collisions Plomb-Plomb à 2,67 TeV. Les données de RHIC ont été normalisées à celles du LHC à $N_{part} = 350$, en utilisant un facteur multiplicatif de 2,14. Les valeurs à 2,76 TeV sont une combinaison statistique des résultats obtenus par trois détecteurs du LHC.

La mesure de la multiplicité permet également de contraindre les modèles de l'état initial. La distribution de la multiplicité des particules chargées par paire de

participants [167, 168, 169]⁴⁶ mesurée dans les collisions centrales noyau-noyau [170, 171, 172]⁴⁷ est représentée dans la Figure 83 en fonction de l'énergie du centre de masse. Avec l'ajout des résultats du LHC, l'évolution de la production des particules est compatible avec une loi de puissance $\approx s^{0,15}$. Il est intéressant de noter que les données proton-proton semblent également bien décrites par une loi de puissance, mais avec une dépendance à l'énergie moins forte ($\approx s^{0,11}$).

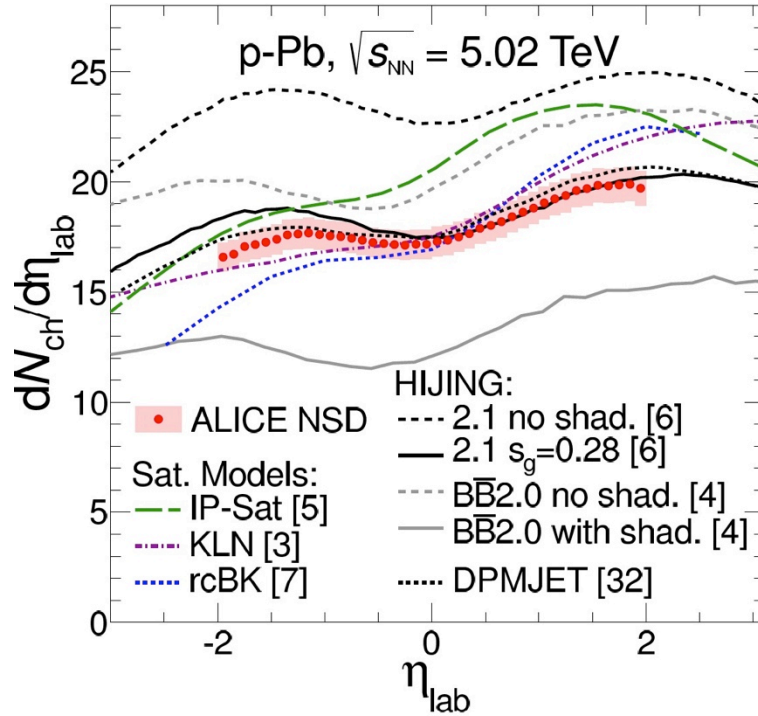


Figure 84 : Densité de particules chargées en fonction de la pseudorapacité mesurée dans les collisions proton-Plomb à $\sqrt{s_{NN}} = 5,02$ TeV, comparée aux prédictions théoriques [173, 174, 175, 176, 177].

La dépendance de la production de particules par rapport à la centralité mesurée au LHC et à RHIC (Figure 83) montre une remarquable similitude dans la forme des deux distributions. Pour les collisions périphériques, cependant, les deux distributions se séparent et tendent vers les valeurs respectives mesurées dans les collisions proton-proton ($N_{part} = 2$). Le fait que la forme des distributions varie peu entre les énergies du RHIC et du LHC est surprenant. En effet, on pourrait s'attendre à ce qu'une contribution significative à la production des particules des processus durs augmente avec le nombre de collisions binaires. Cela devrait conduire à une dépendance plus forte à la centralité, comme prévu par le modèle partonique à deux composantes (molle et dure) implémenté dans

⁴⁶ Le nombre de nucléons participants (N_{part}) ou le nombre de collisions nucléon-nucléon (N_{coll}) contenus dans le volume de la zone de recouvrement des deux noyaux et qui donc participent à la collision est calculé à partir d'une simulation MonteCarlo avec la méthode de Glauber.

⁴⁷ À noter que même à centralité égale, le nombre moyen de participants et de collisions sont différents aux diverses énergies en raison de la différence de taille du noyau (Plomb ou Or) et de l'augmentation de la section efficace proton-proton.

DPMJET [178]. Cependant, une forte atténuation de la fonction d'onde des hadrons (en anglais *shadowing*) à certaines valeurs de x pourrait tempérer la dépendance de la multiplicité à la centralité. Cet effet est responsable du plateau prévu par le modèle à deux composantes de HIJING [179] qui s'accorde mieux avec les données. Les modèles de saturation basés sur le condensat vitreux de couleur (voir page 42) [180], comme par exemple [181], prédisent naturellement une modification nucléaire de la fonction d'onde des hadrons, ainsi que la valeur de la multiplicité mesurée au LHC (Figure 83).

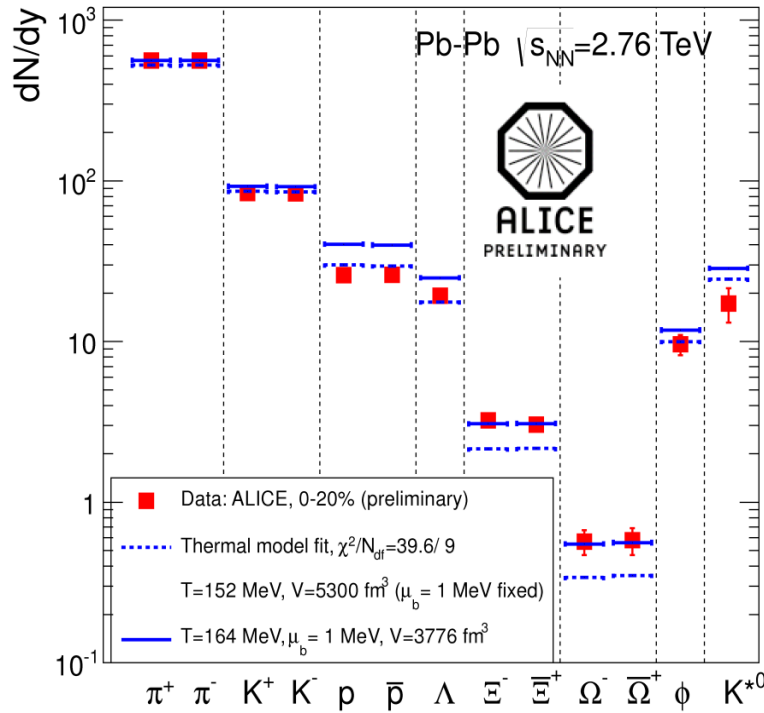


Figure 85 : Production de particules dans les collisions Plomb-Plomb à $\sqrt{s_{NN}}=2.76$ TeV comparée aux prédictions d'un modèle statistique pour deux températures, l'une ($T=152$ MeV, $\mu_B=1$ MeV) résultant d'un ajustement aux données (ligne pointillée) et l'autre d'un ajustement ($T=164$ MeV, $\mu_B=1$ MeV-fixé) excluant les baryons (proton et Λ) (ligne pleine).

Le cadre cohérent qui semble se dégager de ces mesures a été quelque peu altéré par les mesures de multiplicité dans les collisions proton-Plomb effectuées au LHC à $\sqrt{s_{NN}}=5.02$ TeV [182]. Ces collisions offrent la possibilité d'explorer la fonction de structure des hadrons dans le noyau à très petites valeurs de x . La distribution de la densité de particules chargées en fonction de la pseudorapacité (Figure 84) mesurée présente l'asymétrie due à la nature différente des deux projectiles. Les modèles incluant processus durs et mous ainsi que les fonctions de structure des hadrons ajustées aux mesures avec les collisions deuteron-Or au RHIC [183, 184] (DPMJET et HIJING) reproduisent le mieux les données, tandis que les modèles de saturation semblent avoir plus de mal à reproduire la forme et la valeur absolue des mesures.

6.1.2 Abondance des particules identifiées

Les taux relatifs de production de particules identifiées (π , K , p , Λ , Ξ , Ω , etc.) ont été comparés (Figure 85) aux prédictions d'un modèle statistique [185, 186] où les particules sont créées en équilibre thermique sous le contrôle de deux paramètres assimilés à une température (T) et au potentiel chimique baryonique (μ_B) et sous la contrainte des lois de conservation. Un paramètre supplémentaire μ_S permet de décrire la surproduction, par rapport aux prédictions du modèle statistique, de particules étranges observée expérimentalement.

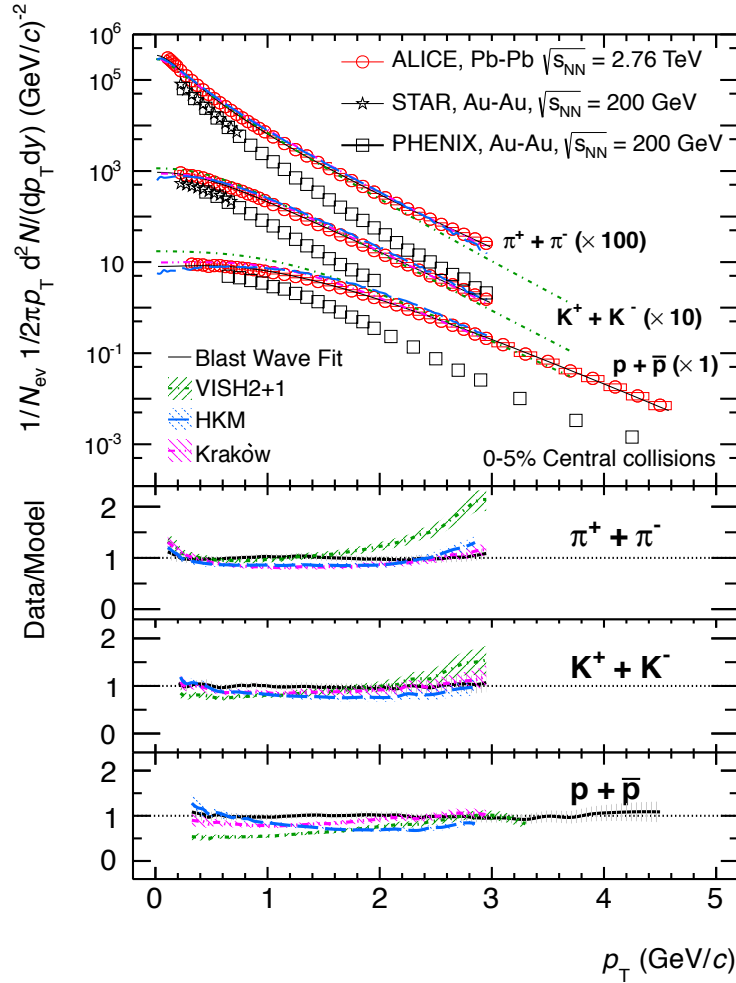


Figure 86 : Spectres en impulsion transverse mesurée pour diverses espèces de particules mesurés à rapidité centrale dans les collisions Plomb-Plomb à $\sqrt{s_{NN}}=2,76$ TeV (ALICE) et Or-Or à $\sqrt{s_{NN}}=200$ GeV (RHIC). Les lignes pleines sont le résultat d'un ajustement aux données ALICE d'une fonction « Blastwave » avec $\langle\beta_t\rangle=0,65c$ et $T=95$ MeV.

De cette comparaison, il est déduit que la température du gel chimique (instant à partir duquel le nombre relatif des diverses particules reste inchangé) est voisine (152 MeV) de la température critique ($T\sim T_c$) ; la valeur de μ_B (1 MeV) est voisine de zéro, traduisant la production en quantité égale de particules et d'antiparticules. Il est cependant surprenant que la valeur obtenue pour la température soit inférieure à la température chimique déduite (164 MeV), à l'aide du même modèle statistique, à partir des données Or-Or à $\sqrt{s_{NN}}=200$ GeV

du RHIC. En répétant le calcul à une valeur fixée de $\mu_B=1$ MeV, et en excluant de l'ajustement la production des baryons (p, et Λ), la même température du gel chimique que celle déduite au RHIC est déduite.

Se pose donc la question de savoir pourquoi le taux de production des baryons est si mal reproduit par le modèle statistique. Une explication possible pourrait être liée aux processus inélastiques qui, du fait de leur importante section efficace, changent la composition des particules même après le gel chimique. Le processus candidat, c'est l'annihilation baryon-antibaryon et effectivement il a été mis en évidence dans les spectres de proton et les mesures de femtoscopie [187].

6.1.3 Spectre des particules

Après le gel chimique, les hadrons produits continuent à diffuser en modifiant leur cinématique et, comme nous l'avons vu, peut-être même leur composition. Ce processus se poursuit jusqu'à ce que les hadrons soient assez distants les uns des autres et cessent d'interagir, et donc leur cinématique est figée à la température du gel cinématique.

Les distributions d'impulsion des hadrons de différentes espèces reflètent cette condition de température ainsi que l'expansion radiale collective de la collision ou boule de feu (Figure 86).

La vitesse de l'écoulement radial et la température du gel cinématique peuvent être estimées en ajustant à tous les spectres simultanément une fonction de type « Blastwave » [188]. En comparant les résultats de cet ajustement aux données du LHC et à ceux du RHIC [189], il est observé que, dans les collisions les plus centrales, la vitesse moyenne d'écoulement augmente de manière significative au LHC, atteignant environ $0,65c$, et que la température du gel cinématique est, aux deux énergies, voisine de 95 MeV. Il est à noter que les calculs qui incluent des interactions élastiques et inélastiques dans l'évolution du système après le gel chimique et jusqu'au gel cinématique reproduisent le mieux les données, notamment le spectre des pions à haute impulsion (interactions élastiques) et les spectres des protons à basse impulsion (interactions inélastiques, en particulier annihilation proton-antiproton). Cette dernière observation pourrait donc aussi expliquer le déficit du taux de production des baryons par rapport aux prédictions du modèle statistique, comme noté plus haut.

Les observables construites à partir de corrélations entre hadrons donnent accès à de nouvelles propriétés globales et collectives caractérisant la collision. Le volume du système au moment du gel cinématique et sa durée de vie sont déterminés à partir de mesures interférométriques (corrélations Hanbury-Brown et Twiss) avec des hadrons identiques. Un volume et une durée de vie de respectivement 5000 fm^3 (plus de cinq fois le volume d'un noyau de Plomb) et $10 \text{ fm}/c$ ont été mesurés pour des collisions centrales Pb-Pb, soit 2 fois et 1,3 fois les valeurs obtenues à RHIC.

6.1.4 L'état initial de la collision

Les photons directs⁴⁸ sont produits tout au long de l'évolution de la collision et, du fait de leur caractère électromagnétique, sont les témoins directs des conditions dans lesquelles ils ont été créés. Dans le spectre de photons directs (Figure 87) mesuré dans les collisions Plomb-Plomb à $\sqrt{s_{NN}}=2,76$ TeV deux composantes apparaissent distinctement. La première à haute énergie présente la dépendance en loi de puissance typique d'un spectre de photons directs prompts produits dans la phase initiale de la collision par processus pQCD durs. La deuxième composante à basse énergie présente une forme exponentielle caractéristique d'un rayonnement thermique de type corps noir émis pendant la phase partonique et la phase hadronique de la collision.

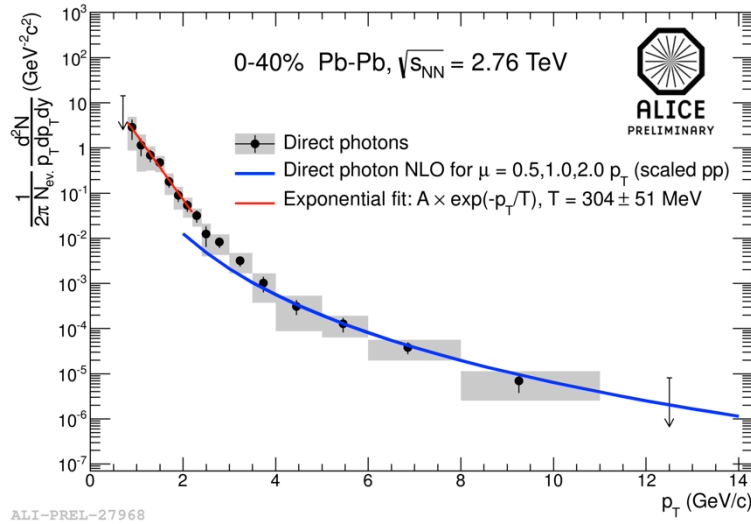


Figure 87 : Spectre des photons directs mesuré dans les collisions centrales (0-40 %) Plomb-Plomb à $\sqrt{s_{NN}}=2,76$ TeV. Les lignes continues représentent le résultat d'un ajustement d'une loi de puissance à la partie haute énergie et d'une fonction exponentielle à la partie basse énergie.

La pente de la partie exponentielle peut être interprétée comme l'inverse d'une température effective moyennée sur la durée et l'intensité du rayonnement. Une valeur de 304 MeV est déduite du spectre. La température initiale du système équilibré (qu'il faudra évaluer exactement à l'aide de calculs hydrodynamiques) est supérieure d'environ un facteur 2 à cette valeur et donc très supérieure à la valeur de la température critique de la transition de phase hadronique.

6.2 Propriétés du QGP

Après avoir considéré les propriétés globales des événements, nous décrirons les résultats relatifs à quelques-unes des observables, qui peuvent nous donner des informations sur les propriétés du plasma de quarks et gluons.

⁴⁸ Par définition, sont appelés photons directs tous les photons qui ne proviennent pas de la décroissance d'autres particules.

6.2.1 Écoulement collectif des particules

La distribution azimutale des hadrons est décrite par une série de Fourier dont les coefficients (v_n) caractérisent l'anisotropie de l'écoulement collectif par rapport à des plans de symétrie. Le coefficient v_2 , ou écoulement elliptique, reflète l'asymétrie géométrique dipolaire initiale (forme d'une ellipse) de la zone de réaction, qui induit une asymétrie identique du gradient de pression dans le QGP (gradient plus fort dans la direction du petit axe de l'ellipse) imprimée dans la forme de l'écoulement collectif des hadrons dans l'état final. La capacité du système à transférer l'asymétrie géométrique initiale en une asymétrie dans l'espace des impulsions est une propriété fondamentale du QGP caractérisée par la viscosité de cisaillement, le plus souvent exprimée en unités de η/s (viscosité de cisaillement η sur la densité d'entropie s).

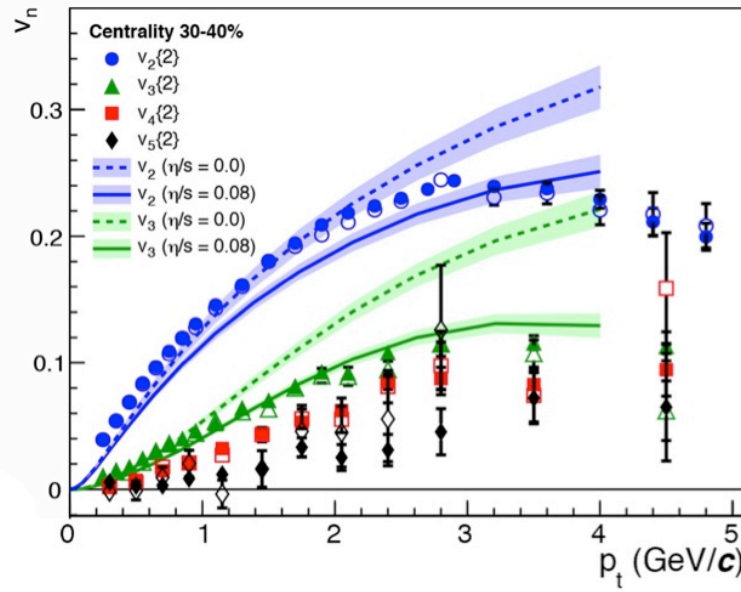


Figure 88: Spectres de l'écoulement elliptique en fonction de l'impulsion transversale pour des espèces de particules de masse différente produites dans les collisions Plomb-Plomb à $\sqrt{s_{NN}}=2,76$ TeV et de centralité intermédiaire.

La valeur de v_2 et sa dépendance à l'impulsion transverse mesurées par ALICE par la technique des corrélations à deux particules sont identiques aux résultats de RHIC, traduisant la similarité du fluide d'un point de vue hydrodynamique aux énergies RHIC et LHC. Les harmoniques d'ordre supérieur ont pour origine la forme très irrégulière de la zone de recouvrement, qui fluctue d'un événement à l'autre. Ces fluctuations résultent du caractère aléatoire des collisions nucléon-nucléon individuelles dans la voie d'entrée de la collision. Cette géométrie très irrégulière de l'état initial donne naissance à la même irrégularité des gradients de pression, et ce d'autant plus facilement que la viscosité du milieu est faible (la viscosité décrit comment l'impulsion se propage dans le milieu). ALICE a déterminé les harmoniques de l'écoulement collectif jusqu'à l'ordre six. Par comparaison avec des calculs hydrodynamiques, la valeur de la viscosité ($4\pi\eta/s \leq 2$) suggérée par les mesures se rapproche de la valeur limite pour un fluide quantique renforçant l'idée du QGP comme un liquide parfait.

Cependant, cette valeur calculée est fortement corrélée aux choix des conditions initiales. Pour s'affranchir d'un choix arbitraire et déduire des mesures une valeur précise de la viscosité (valeur fondamentale du QGP calculable ab initio, rappelons-le), de nouvelles mesures à la fois plus précises et plus contraignantes sont nécessaires.

Un test très important pour valider l'interprétation de l'asymétrie azimutale dans la production des particules en termes d'écoulement collectif est la dépendance en masse de v_2 mesurée pour diverses espèces de particules (π , K, p, Λ) (Figure 89) [190, 191]. Rappelons que l'écoulement radial tend à égaliser la vitesse des particules, et non leur impulsion, et qu'il «pousse» les particules lourdes vers les valeurs élevées de p_t [192].

Le modèle hydrodynamique [193, 194] décrit correctement les données pour toutes les espèces de particules jusqu'à une impulsion transverse d'environ 2 GeV/c. Il prédit aussi que l'effet de masse est plus marqué au LHC qu'au RHIC comme conséquence de l'augmentation de l'écoulement radial.

Aux grandes impulsions transverses ($p_t > 10$ GeV/c), l'anisotropie est due au mécanisme de perte d'énergie des jets, qui reflète également l'anisotropie géométrique initiale puisque la perte d'énergie est proportionnelle à la distance parcourue dans le milieu.

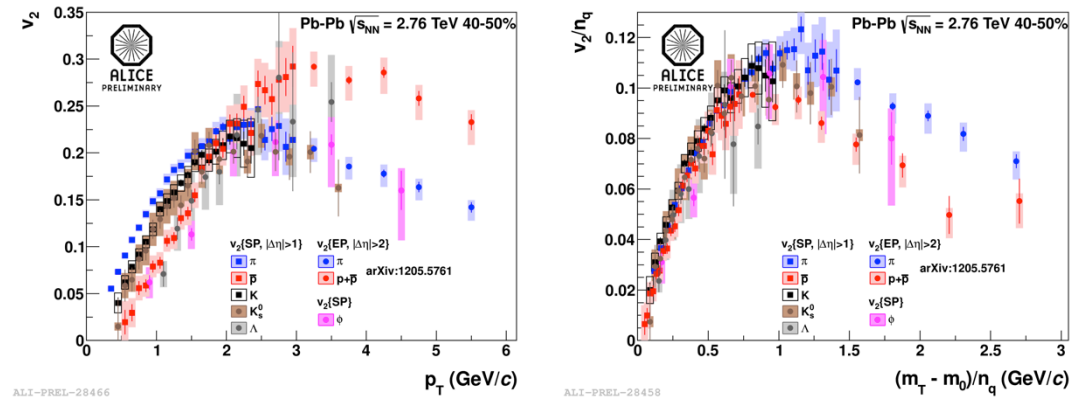


Figure 89 : Spectres de l'écoulement elliptique pour différentes espèces de particules produites dans les collisions Plomb-Plomb à $\sqrt{s_{NN}}=2,76$ TeV et de centralité intermédiaire. À gauche, écoulement elliptique en fonction de l'impulsion transverse. A droite, écoulement elliptique en fonction de l'énergie cinétique transverse, normalisé au contenu en quarks de valence des particules. Les valeurs de v_2 sont calculées avec la méthode du plan de l'événement (EP dans la figure) et des cumulants à deux particules avec intervalle de rapidité (SP dans la figure) [195].

La région intermédiaire en impulsion est moins bien comprise. Il a été observé que dans cette région, tous les baryons d'un côté et tous les mésons d'un autre côté suivent une tendance similaire. Cette déviation par rapport au comportement hydrodynamique est liée à l'anomalie des baryons, un phénomène déjà mis en évidence à RHIC, et est observée de façon beaucoup plus marquée dans une région en impulsion transverse dite intermédiaire ($1,5 < p_t < 6$ GeV/c). A une valeur donnée de l'impulsion transverse, le rapport baryons (p, Λ) sur mésons (π , K) produits dans les collisions Pb-Pb est beaucoup

plus important (un facteur 3) que le même rapport mesuré dans les collisions p-p. Cet excès est d'abord dû à l'écoulement radial qui amplifie l'impulsion des baryons plus fortement que celle des mésons (l'écoulement radial tend à évaluer la vitesse des hadrons et non leur impulsion). Cependant, au-delà d'environ 3 GeV/c, la distribution de l'excès indique que la dynamique des hadrons se découple progressivement de l'écoulement hydrodynamique. Cette observation indiquerait l'apparition d'un nouveau phénomène de hadronisation, la formation de hadrons par coalescence ou recombinaison des quarks constituant le QGP [196, 197, 198, 199]. Ce mécanisme confère une fraction plus importante de l'impulsion collective des quarks du QGP aux baryons constitués de trois quarks qu'aux mésons constitués d'un quark et d'un antiquark. Une description quantitative de ce mécanisme n'est cependant pas encore acquise et nécessitera des contraintes expérimentales supplémentaires.

Les quarks lourds, charmés et beaux, du fait de leur masse élevée sont essentiellement produits dans la phase initiale de la collision par processus durs. On s'attend à ce que ces quarks interagissent fortement avec le milieu, avec la possibilité de thermaliser avant de fragmenter. Dans ce cas, les hadrons de saveurs lourdes devraient hériter de l'anisotropie azimutale moyenne, même si en raison de leur grande masse, les quarks charmés et beaux sont moins sensibles à l'expansion collective: il faut des interactions fréquentes avec un couplage fort pour qu'ils acquièrent un v_2 non nul.

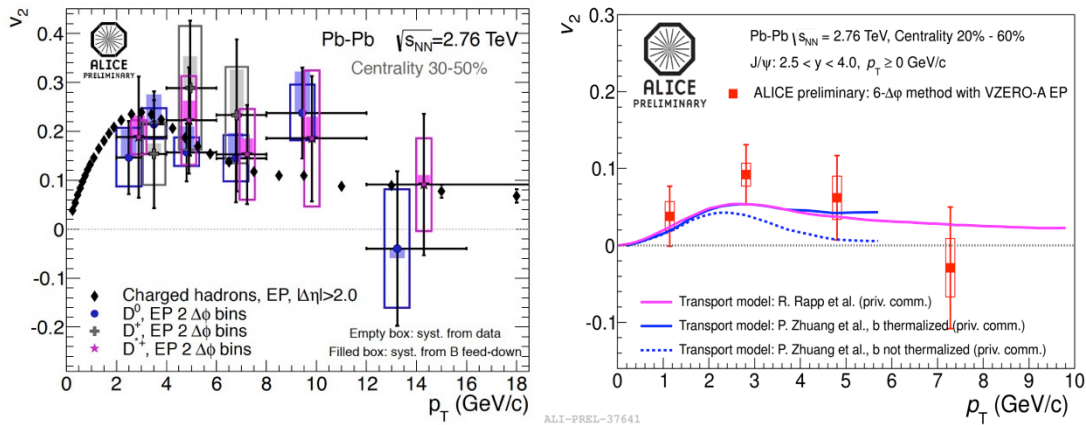


Figure 90 : (à gauche) Coefficient de l'écoulement elliptique v_2 des mésons D^0 , D^+ et D^* mesuré dans les collisions Plomb-Plomb à $\sqrt{s_{NN}} = 2,76$ TeV dans la classe de centralité 30-50 % comparé au v_2 des hadrons chargés. (à droite) v_2 de J/ψ en fonction de l'impulsion transverse.

Les premières mesures [200, 201] indiquent (voir Figure 90) qu'il existe un écoulement elliptique non nul pour le charme ouvert et caché. Nous pouvons donc avancer l'hypothèse que les quarks charmés participent à la dynamique du QGP. Ces résultats doivent être combinés avec la distribution de R_{AA} dont nous parlerons ensuite pour avoir une image plus complète de la dynamique du QGP.

6.2.2 Suppression des jets

Les sondes dures de grande impulsion transverse ou de grande masse sont produites dans des processus durs et leur temps de formation ($t \sim 1/p_t$, $t \sim 1/m$)

est très bref comparé au temps de formation du QGP. Le taux de production de ces sondes et leurs caractéristiques sont alors potentiellement modifiés par des interactions dans l'état final du processus dur lorsqu'elles traversent le QGP. L'étude de ces modifications permet d'explorer l'interaction d'objets colorés avec le milieu coloré du QGP ou encore de tester l'état déconfiné du QGP. Parmi les sondes dures, ALICE a tout particulièrement étudié les hadrons de grande impulsion transverse issus de la fragmentation de quarks et gluons durs, les hadrons charmés et beaux ainsi que le J/ψ (état $c\bar{c}$).

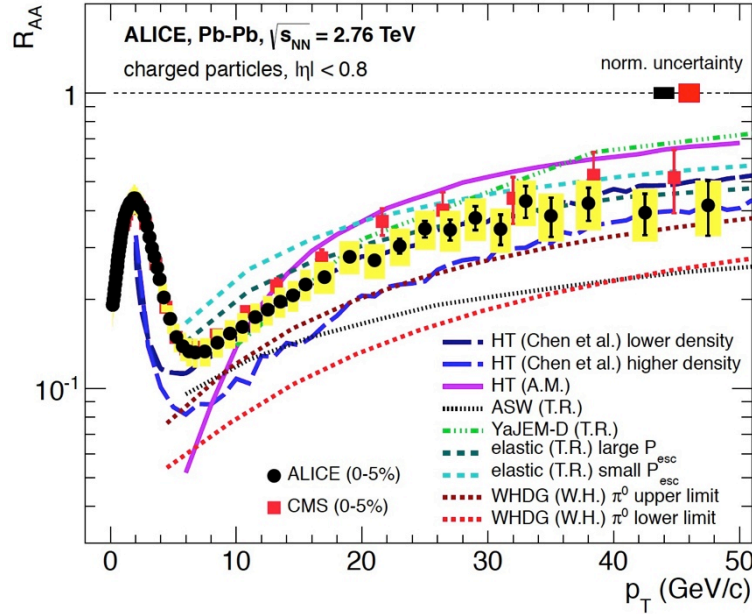


Figure 91 : Facteur de modification nucléaire R_{AA} pour les particules chargées mesurées par ALICE dans les collisions Plomb-Plomb centrales (0-5 %) en comparaison avec les résultats de CMS et différents modèles théoriques. Voir la référence [202] pour les détails.

Le facteur de modification nucléaire R_{AA} est défini par comparaison au taux de production d'un type de particules dans les collisions Plomb-Plomb et proton-proton normalisé par le nombre de collisions nucléon-nucléon :

$$R_{AA}(p_t) = \frac{d^2 N_{AA} / dp_t d\eta}{\langle T_{AA} \rangle d^2 \sigma_{NN} / dp_t d\eta},$$

où N_{AA} et σ_{NN} représentent le nombre de particules produites dans les collisions noyau-noyau et la section efficace de la même production dans les collisions nucléon-nucléon. La fonction de recouvrement nucléaire $\langle T_{AA} \rangle$ est le rapport entre le nombre de collisions binaires et la section efficace inélastique proton-proton $\sigma_{pp} = (64 \pm 5) \text{ mb}$ à $\sqrt{s_{NN}} = 2,76 \text{ TeV}$. En l'absence d'effets nucléaires, le facteur R_{AA} est égal à 1 par construction. Le facteur de modification nucléaire est le résultat combiné de la fonction de structure nucléaire qui décrit la fonction d'onde hadronique dans le noyau, et de l'effet du milieu sur la propagation et la fragmentation des partons durs traversant le milieu, c'est-à-dire un effet de l'état final (après production du parton dur).

Le facteur de modification nucléaire a été mesuré par ALICE dans les collisions centrales pour les hadrons chargés jusqu'à de très grandes valeurs d'impulsion transverse (Figure 91) [202]. La distribution présente aux grandes impulsions transverses ($<10\text{GeV}/c$), où la production par fragmentation de partons durs est dominante, la forme caractéristique attendue pour le phénomène du jet quenching. La suppression tend à diminuer vers les hautes valeurs de p_t du fait de la diminution de la valeur relative d'énergie perdue. Cette observation confirme qualitativement les résultats de RHIC, mais quantitativement l'effet est plus marqué au LHC, traduisant une densité partonique plus importante au LHC comparée à celle atteinte à RHIC. Afin de préciser la nature du mécanisme de perte d'énergie, par rayonnement de gluons et par collisions élastiques dans le cadre d'une approche perturbative de QCD, ou par des modes hydrodynamiques dans le cadre d'une théorie de champ à couplage fort où la notion de quasi particules est absente, les valeurs de R_{AA} pour diverses particules identifiées ont été comparées entre elles, notamment pour les protons, issus majoritairement de la fragmentation de gluons, et les mésons charmés D, issus de la fragmentation de quarks lourds charmés (Figure 92) [203, 204].

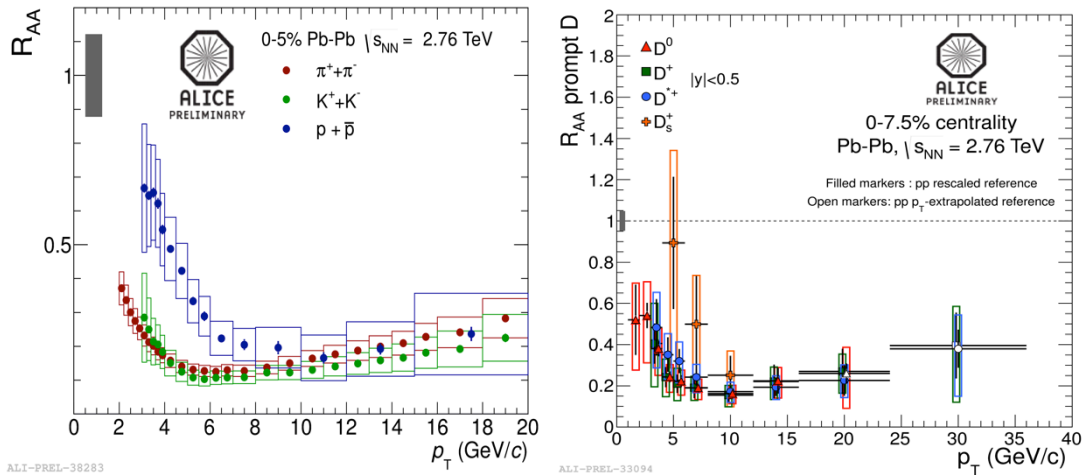


Figure 92 : (gauche) R_{AA} en fonction de p_t pour $\pi^\pm$, $K^\pm$, p et $\bar{p}$ pour les collisions Plomb-Plomb à centralité 0-5 %. (droite) R_{AA} en fonction de p_t pour D^0 , D^+ , D^{*+} et D_s^+ pour la classe de centralité 0-7,5 %.

La hiérarchie de charge de couleur et de masse ($R_{AA}^g < R_{AA}^q < R_{AA}^Q$) attendue pour le mécanisme de perte d'énergie par rayonnement de gluons n'est pas établie sans ambiguïté, sans être vraiment exclue. Ces résultats préliminaires suggèrent une contribution du mécanisme de perte d'énergie par collisions élastiques. En l'absence de données contraignantes supplémentaires, la détermination expérimentale du paramètre $\hat{q}$ (coefficient de transport identifié au taux de dispersion de l'impulsion transverse d'un parton traversant le QGP ; ce coefficient est proportionnel à la densité de gluons) caractéristique du QGP et calculable ab initio reste hors de portée. Une autre question ouverte est de savoir comment est dispersée l'énergie injectée par le parton dans le QGP ou, en d'autres termes, quelle est la réponse du QGP à un dépôt d'énergie : cette énergie est-elle rapidement thermalisée dans le milieu ou reste-t-elle dans le jet et se manifeste-t-elle par une modification de la fonction de fragmentation ? Des mesures multi-différentielles, nécessitant des statistiques plus importantes que

celles actuellement à notre disposition, pourront apporter des éléments de réponse.

Une importante clarification est en train d'être apportée par l'analyse des collisions proton-Plomb. Comme nous pouvons le voir dans la Figure 93 [205], nous n'observons aucune suppression pour une impulsion transverse supérieure à 2 GeV/c. Cette mesure indique que la forte suppression de la production de hadrons à haute p_t observée au LHC dans les collisions Plomb-Plomb n'est pas due à un effet de l'état initial, mais est bel et bien l'empreinte de l'atténuation des jets QCD dans la matière chaude.

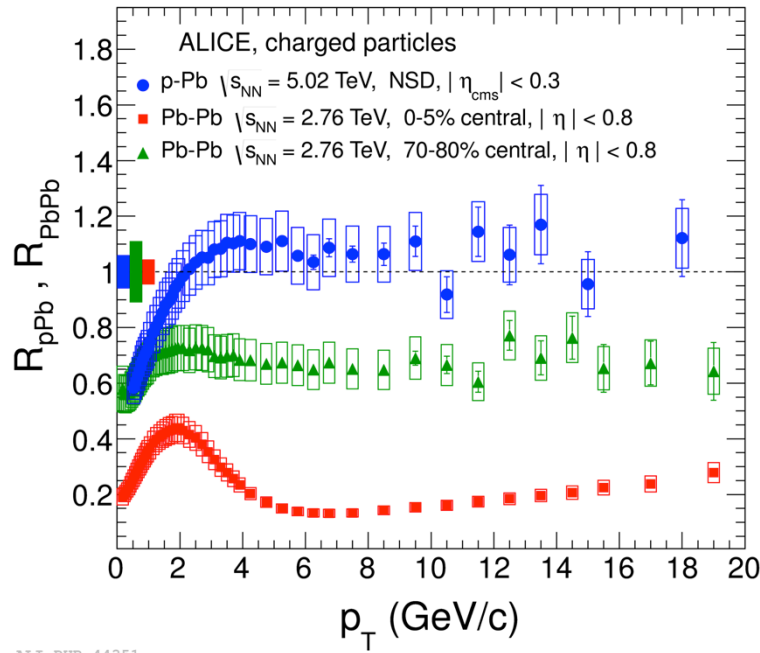


Figure 93 : Facteur de modification nucléaire des particules chargées en fonction de l'impulsion transverse pour les collisions proton-Plomb à $\sqrt{s_{NN}}=5,02$ TeV. Les données sont comparées à des mesures pour les collisions Plomb-Plomb à $\sqrt{s_{NN}}=2,76$ TeV dans deux classes de centralité différentes : 0-5 % and 70-80 %.

Plus de données et une comparaison quantitative avec des modèles seront nécessaires pour expliquer la petite différence qui, avec les statistiques actuelles, n'est pas très significative, entre la suppression des mésons de quarks légers et les mésons de quarks lourds.

6.3 La saga du J/ψ

L'écrantage de couleur (phénomène analogue à l'écrantage Debye de la charge électrique dans un plasma), une propriété intrinsèque du QGP due à la propriété de liberté asymptotique de QCD, peut être testé en mesurant le taux de survie des états liés des quarkonia dans le QGP. L'objectif de telles mesures est d'établir une échelle d'écrantage de couleur à partir des différents quarkonia, les états le plus faiblement liés étant le plus facilement écrantés, donc le plus fortement supprimés. Des effets dus d'un côté à la structure dans la voie d'entrée, et d'un autre côté à des mécanismes dans la voie de sortie sont cependant à prendre en

compte afin d'évaluer correctement le rôle de l'écrantage de couleur dans la production des quarkonia. La mesure de R_{AA} pour les mésons J/ψ a fourni de précieuses informations sur les divers mécanismes en jeu. Aux impulsions transverses élevées (Figure 94 à droite), la suppression observée suit la phénoménologie établie pour les hadrons légers, en particulier une suppression dans les collisions centrales au moins aussi forte que celle mesurée à RHIC.

La production de J/ψ de basses impulsions transverses est, quant à elle, beaucoup moins supprimée, suggérant une régénération de J/ψ par recombinaison de quarks charmés constituant le QGP (Figure 94 à gauche).

Une interprétation plus définitive de ces observations nécessite cependant des mesures supplémentaires pour identifier et quantifier les effets de la voie d'entrée sur la production de quarks charmés. Ainsi, les effets d'ombrage ("shadowing") et de saturation des gluons caractérisant la structure des noyaux en collision à des valeurs de x beaucoup plus petites que celles en jeu aux énergies de RHIC, et à ce jour inexplorés pourraient modifier cette production très différemment aux énergies RHIC et LHC. L'absorption dans l'état final par la matière nucléaire froide pourrait, elle aussi, modifier différemment la production de J/ψ . L'ensemble de ces effets sera mieux compris grâce à l'analyse des données proton-Plomb disponibles dès la fin de cette année.

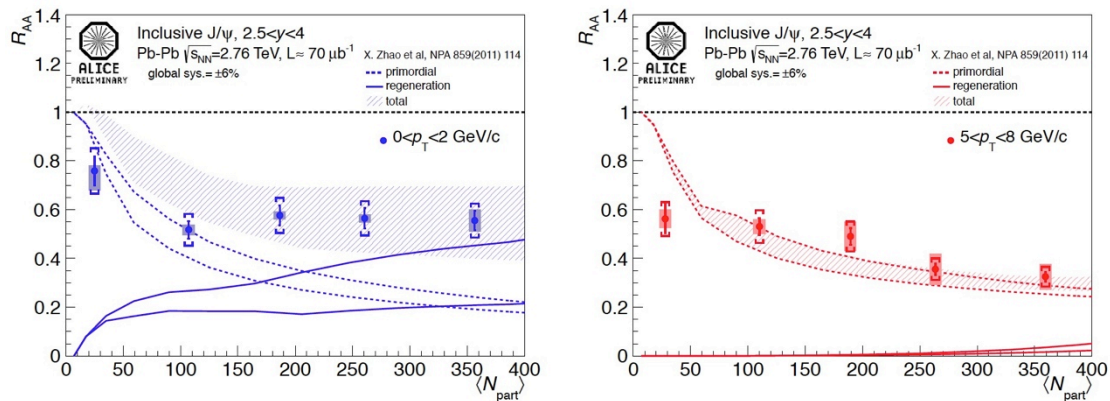


Figure 94 : R_{AA} en fonction du nombre des participants pour deux intervalles d'impulsion transversale ($0 < p_t < 2$ et $5 < p_t < 8$ GeV/c), superposée aux résultats d'un modèle de transport partonique. Les contributions des composantes primordiales (en pointillés) et de régénération (ligne continue) sont représentées, de même que leur somme. La zone comprise entre les courbes représente une estimation de l'incertitude théorique.

En conclusion de ces premières prises de données pour les collisions Plomb-Plomb à 2,76 TeV, les résultats obtenus par ALICE ont permis de consolider la compréhension de la dynamique des collisions d'ions lourds établie par les résultats des expériences de RHIC durant ces dix dernières années. L'augmentation conséquente de l'énergie, associée aux remarquables performances de ALICE en termes d'identification de particules et de trajectographie jusqu'à très petites impulsions transverses, permet de considérer des degrés de liberté supplémentaires, le charme et la beauté par exemple, intervenant dans la thermalisation du QGP. Les observables

multidimensionnelles ouvrent la voie vers des mesures de précision de propriétés du QGP, telles la viscosité, qui découlent directement de la nature intrinsèque de l'interaction forte. Les sondes externes ou dures produites abondamment aux énergies LHC, renforcées par de nouvelles observables potentielles, stimulent une meilleure compréhension des propriétés de transport de charges de couleur par un milieu QCD coloré.

Restent cependant un certain nombre de questions ouvertes qui trouveront, pour certaines, des réponses avec les données supplémentaires (statistique accrue pour la production d'événements rares et pour les analyses multidimensionnelles; collisions p-Pb pour identifier les divers effets sur les observables dus à la structure partonique de la voie d'entrée) jusqu'à la luminosité intégrée de 1nb^{-1} prévue dans le programme scientifique originel de ALICE. Cet objectif pourra être atteint d'ici l'arrêt du LHC prévu en 2017. Pour aborder les questions restantes, une augmentation de la luminosité intégrée par au moins un ordre de grandeur est indispensable, ce qui implique une modification importante du système de détection et un fonctionnement se poursuivant au-delà de 2018. Parmi les questions ouvertes, citons: à quelle température la charge de couleur est-elle écrantée? Quelle est la valeur de la viscosité et de la vitesse du son dans le QGP? Comment les partons perdent-ils leur énergie en traversant le QGP? Comment l'énergie injectée dans le QGP se dissipe-t-elle? La notion de quasi-particules comme constituants du QGP est-elle applicable? Comment le QGP peut-il thermaliser sur des échelles de temps aussi rapides? Quelle est la nature de l'état initial précurseur du QGP? La réponse à ces questions apportera un éclairage nouveau sur QCD.

7 Conclusions

Dans ce travail, nous avons présenté brièvement un des plus grands instruments scientifiques jamais réalisés. Nous avons énoncé les buts scientifiques qui ont motivé sa réalisation, nous avons donné des détails techniques sur la constitution des sous-détecteurs qui le composent et nous avons décrit les résultats qui ont été obtenus pendant les deux premières années d'opération, avant la fermeture de l'accélérateur pour des travaux d'amélioration. Au début de 2013, il y aura un mois de collisions proton-Plomb qui sont très attendues pour pouvoir faire la lumière sur l'interaction entre effets de l'état final et fonctions de modification des nucléons. Comme nous l'avons vu, les collisions proton-Plomb enregistrées en 2012 pendant une brève période de prise de données ont déjà fourni des résultats assez intrigants.

Si la possible (probable) découverte du boson de Higgs se confirme, les expériences ATLAS et CMS auront mis la dernière brique dans l'édifice du modèle standard. Mais tel qu'il est maintenant, cet édifice semble avoir des limites indépassables pour répondre aux demandes qui restent en suspens sur l'origine de la masse et de l'énergie manquante et sur les liens avec la relativité. Une partie de la réponse viendra-t-elle de l'étude de la matière primordiale générée dans les collisions d'ions lourds ? Il est sûrement trop tôt pour le dire, mais en tous cas nous ne pouvons pas l'exclure. En quelque sorte l'achèvement du modèle standard, si cela se confirme, nous libère d'une tâche et nous permet de partir à la recherche des réponses qui nous font encore défaut.

La partie centrale et plus importante de ce travail de thèse est la description du système informatique qui a été conçu, réalisé et déployé au cours des quatorze dernières années par l'équipe dirigée par l'auteur en tant que chef de groupe et de projet. Ce système informatique est à la mesure de l'appareil scientifique pour traiter les données duquel il a été conçu. Nous pensons qu'encore une fois l'outil informatique développé a été à la hauteur du défi technique et scientifique posé par le détecteur et son ambitieux programme de physique. La quantité et la qualité des résultats produits en témoignent.

Le calcul pour la Physique des Hautes Énergies est désormais une discipline bien établie, avec une longue histoire et un rôle de premier plan dans les nombreuses et importantes découvertes scientifiques faites par cette science depuis la deuxième guerre mondiale. Néanmoins, relativement peu d'ouvrages spécifiques lui sont dédiés, hormis les comptes rendus des conférences sur le sujet et les articles dans les revues spécialisées.

En particulier, nous avons senti le manque d'une littérature qui relie le développement du calcul dans la Physique des Hautes Énergies avec le développement général de la science de l'information, en dépit de faits majeurs tels que l'invention par le CERN de la Toile et le déploiement de la première et plus importante Grille de calcul mondiale. Avec ce travail, nous avons porté un regard que nous réalisons être assez personnel sur cet aspect à partir de notre expérience de plus d'une décennie au cœur du projet de calcul de ALICE.

Les défis qui attendent le calcul pour l'expérience ALICE, et le calcul pour la Physique des Hautes Énergies en général, sont encore plus formidables que ceux que nous avons affrontés. Les nouvelles architectures matérielles à parallélisme élevé doivent être exploitées avec grande efficacité pour pouvoir disposer de la puissance de calcul nécessaire pour réussir des programmes de recherche de plus en plus ambitieux dans des budgets qui s'annoncent assez serrés. Les études préliminaires que nous avons menées nous montrent que cela demandera une réécriture massive de notre logiciel, dans une situation où l'évolution technologique s'accélérera davantage avec l'apparition de nouvelles architectures matérielles et de nouveaux langages de programmation. La réécriture de nos programmes pose de gros défis techniques, car si nous avons un grand parallélisme naturel intrinsèque à nos problèmes (tous les événements sont indépendants, toutes les particules se propagent dans le détecteur de façon indépendante), son exploitation a longtemps défié nos programmeurs. L'absence d'un « noyau computationnel » à optimiser, la grande hétérogénéité des objets que nous traitons (géométrie des détecteurs, différentes particules, gammes d'énergie, etc.) et le mélange d'opérations entières et en virgule flottante et de décisions logiques rendent les techniques d'optimisation utilisées dans d'autres domaines difficiles à adopter.

Mais même si nous arrivions à optimiser notre code sur les nouvelles architectures parallèles au prix d'une réécriture substantielle et de l'utilisation de langages et techniques spécifiques, nous devrions nous poser le problème de la gestion du cycle de vie d'un tel logiciel, et de son acceptation par les physiciens, qui ne peuvent pas tous devenir des experts de programmation parallèle. Déjà le passage du FORTRAN au C++ a eu des effets négatifs collatéraux que nous ne sommes pas encore capables de gérer complètement. Le passage possible à des langages spécifiques pour un certain matériel ou architecture système pourrait être d'autant plus problématique.

La Grille a été la cheville ouvrière du calcul de LHC, mais le futur semble appartenir au Nuage. La différence entre les deux est en partie sémantique et en partie réelle. Du point de vue technique, l'utilisation d'une structure à Nuage pose des problèmes considérables, surtout pour la gestion des données. Du point de vue économique, l'évaluation de la rentabilité d'un possible passage, partiel ou total, à l'utilisation de Nuages commerciales telles que Amazon est compliqué par la disparité des modèles de financement et de gestion, technique et politique, du calcul pour la Physique des Hautes Énergies dans les différents pays. Nous avons commencé seulement maintenant à explorer l'impact d'une possible transition.

En parallèle, le monde de la Technologie de l'Information s'est approprié le concept de « grandes données » (en anglais *Big Data*) et il est essentiel pour nous de nouer des liens étroits avec les centres qui pensent le traitement des bases de données extra-larges (ce qu'en anglais on indique par *Exascale Computing*) et qui sont en train de développer les concepts et les logiciels pour leur traitement. Si d'une part nous avons beaucoup d'expérience et aussi de produits à offrir, de l'autre nous devons nous assurer que les nouvelles solutions qui sont en train d'émerger n'ignorent pas nos besoins et nos demandes. Dans les technologies qui évoluent avec grande rapidité, comme c'est le cas pour la technologie de

l'information, c'est souvent le cas que les pionniers d'hier deviennent les « dépassés » de demain par un effet de « saute-mouton » technologique (en anglais *leapfrogging*) qui est bien connu, mais très difficile à éviter, surtout quand il y a une lourde infrastructure « de production » à maintenir active.

Comment ce processus sera géré et par qui, c'est aussi une question qui mérite d'être considérée. Le CERN a été jusqu'à ici le centre de cristallisation des développements du calcul pour la Physique des Hautes Énergies. Ce rôle lui a été disputé à plusieurs reprises dans le passé et même il a été ressenti comme la source d'une forme d'« autorité » qui a exclu des développements parfois prometteurs seulement parce qu'ils n'étaient pas dans l'orthodoxie décidée à et par le CERN, au point de parler d'« impérialisme informatique » du CERN. La qualité indéniable de plusieurs produits sortis du CERN (mais, il faut le souligner, pas de tous les produits), le support offert et le modèle « open source » adopté, qui a permis à toute la communauté de contribuer, ont été des facteurs prépondérants dans la création de cette situation.

L'adoption consensuelle et mondiale de Root et l'entrée en fonctionnement de LHC, qui est exploité par plus de la moitié des physiciens des Hautes Énergies du monde, a estompé ce débat, mais seulement de façon temporaire. Les nouvelles machines linéaires qui sont maintenant à l'étude par la communauté de la Physique des Hautes Énergies seront très probablement le résultat d'efforts plus globaux et partagés que le LEP, le Tevatron ou même LHC, où déjà des contributions substantielles à la machine même ont été fournies par des états non membres du CERN. Nous pouvons donc nous demander quel sera le nouveau modèle de développement, distribution et support du logiciel, comment seront agencés les moyens de calcul et où sera et quelle forme aura le centre décisionnel dans le futur.

Tout cela reste à imaginer et à réaliser. Dans la Physique des Haute Énergies, nous avons une solide expérience, mais aussi un lourd bagage de traditions et d'habitudes qui nous rendent l'évolution parfois difficile à imaginer. Nous avons aussi une tradition de fermeture sur nous-mêmes qui a des raisons objectives, car la plus grande partie des problèmes que nous avons à résoudre sont spécifiques à notre communauté, mais que, face à l'augmentation de la dimension et de la complexité de ces problèmes, il faudrait savoir dépasser pour joindre nos forces avec d'autres disciplines et avec les centres de développement de la nouvelle technologie de l'information.

En tous cas, le futur s'annonce riche en surprises, du point de vue aussi bien scientifique que technologique, et porteur d'opportunités de développement. Nombreuses sont les questions qui attendent une réponse et c'est sûrement un privilège de faire partie de cette belle histoire scientifique et humaine.

8 Glossaire

Acronyme	Anglais	Français	Explication
ACL	Access control list	Liste de contrôle d'accès	Liste des permissions relatives à un élément informatique (habituellement un fichier) définissant quelle action est permise à quel utilisateur.
	Alice week	Semaine de rencontre des membres de l'expérience ALICE	
ADC	Analogue to Digital Converter	Numériseur	Circuit électronique capable de transformer un signal analogique en signal numérique.
AF	Analysis facility	Ferme d'analyse	Grappe d'ordinateurs et de serveurs de stockage utilisée pour l'analyse des données.
AOD	Analysis object data	Données des Objets d'Analyse	
	Analysis tag	Étiquette d'analyse	Version du code partiellement validée et publiée sur la Grille pour l'analyse des données.
API	Application programming interface	Interface de programmation de l'application	Interface de programmation qui permet à un utilisateur d'accéder à la fonctionnalité d'un programme.
ASIC	Application specific integrated circuit	Circuit intégré développé pour une application spécifique	
APD	Avalanche photo diode	Photodiode à avalanche	Dispositif électronique semi-conducteur qui exploite l'effet photo-électrique pour convertir la lumière en signal électrique par un mécanisme d'avalanche d'électrons.
	Back pressure	Contre-pression	Signal émis par un circuit électronique vers une source de données qui ne limite le flux à un taux que le circuit peut élaborer en temps réel.
	Background independent theory	Théorie indépendante de l'environnement	Théorie qui ne suppose pas la préexistence du temps et de l'espace.
	Bandwidth	Largeur de bande	Capacité d'une connexion réseau généralement mesurée en bits par seconde.
	Baryon free zone	Zone sans baryons	Zone centrale d'une collision dont le nombre baryonique net est proche de zéro.
	Batch jobs	Tâches par lot	Séquences d'instructions et programmes exécutés par un système informatique de façon asynchrone avec la requête d'exécution.

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
	Big data	Grandes données	
	Big endian	Gros-boutiste	Architecture d'ordinateur où l'octet le plus significatif d'un mot est stocké en premier, à l'adresse d'octet inférieur.
	Bjorken-x	x de Bjorken	Rapport entre l'impulsion échangée dans une collision de particules chargées et le double du produit de l'énergie perdue par la particule diffusée et de la masse du centre de diffusion.
	Bremsstrahlung	Rayonnement de freinage	Rayonnement émis par une particule chargée subissant une décélération.
	Build	Assemblage (code)	Traduction d'un programme en langage de programmation vers le langage binaire et son assemblage dans des modules exécutables.
	Bump bonding	Soudure par point	Procédé de connexion de dispositifs semi-conducteurs à des circuits externes qui n'utilise pas des fils mais des gouttes de soudure déposées sur les damiers de la puce au cours de l'étape finale de traitement.
	Busy	Occupé (d'un détecteur)	État d'un détecteur où, pendant l'acquisition d'un signal, il est incapable d'en enregistrer un autre.
	Calibration run	Période de prises de données dédiée au calibrage	
CPU	Central processing unit	Processeur central	Dispositif principal d'un ordinateur pour l'élaboration des données.
CTP	Central trigger processor	Processeur central de déclenchement	Dispositif pour la collection et le traitement des signaux de déclenchement provenant des sous-détecteurs et pour la génération du global signal de déclenchement.
	CERN program library	Bibliothèque des Programmes du CERN	
	Certification authority	Autorité de Certification	Entité qui a l'autorité de valider les certificats électroniques des utilisateurs d'un système informatique distribué.
CCIU	Channel Concentration and Interface Unit	Unité d'interface et de concentration des canaux	Carte électronique du détecteur TOF qui récolte les signaux et, après formatage, les envoie au système d'acquisition des données.
CIU	Channel interface unit	Carte d'interface de canal	Carte électronique pour le détecteur V0 qui contient les modules pour la numérisation des signaux des photomultiplicateurs.
CSP	Charge sensitive	Amplificateur de	Circuit électronique qui génère un

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
CGC	preamplifier	charge	signal proportionnel à la quantité de charge accumulée.
	Chemical freeze-out	Gel chimique	Moment de la collision auquel la population des différentes espèces de particules se fige.
	Cloud computing	Nuage	Système distribué offrant des services de calcul indépendamment de la location des ressources et du demandeur.
	Cluster	Grappes de signaux définissant les points de passage des particules	
	Cluster (of detectors)	Grappe de détecteurs	Un ensemble de détecteurs qui sont lus ensemble pendant la prise des données.
	Coding conventions	Conventions de codage	Ensemble de règles syntactiques pour l'écriture d'un programme informatique.
	Collaboration board	Conseil de collaboration	
	Colour glass condensate	Condensat vitreux de couleurs	Type extrême de matière formé par les gluons à l'intérieur d'un noyau voyageant à une vitesse proche de celle de la lumière, apparaissant à un observateur stationnaire comme un « mur gluonique » de grande densité.
	Colour screening	Écrantage de couleur	Effet par lequel une charge de couleur est masquée par les charges de couleur du vide.
	CMRW	Mémoire à colonne pour lecture et écriture	Mémoire rapide à accès facilité.
CE	Computing board	Comité de calcul	
	Computing element	Élément de calcul	Élément d'une Grille de calcul qui abstrait la fonction d'exécution des programmes.
CSG	Computing technical design report	Rapport de Conception Technique du Calcul	
	Condition data	Données des conditions expérimentales	Paramètres (entre autres de calibrage et d'alignement) spécifiques pour une période de prise de données.
	Constructive solid geometry	Géométrie constructive des solides	Description des objets solides réalisée via la combinaison booléenne ou la division d'un certain nombre de solides de base.
	CPU-bound	Limité par la disponibilité de ressources de calcul plutôt que d'entrée et	

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français sortie	Explication
	Crash	Plantage (d'un programme)	Arrêt d'un programme suite à une condition d'erreur non recouvrable.
D-RORC	DAQ readout Receiver Card	Carte électronique de réception et lecture	Carte électronique du système d'acquisition de ALICE qui permet aux ordinateurs LDC de lire les signaux provenant du détecteur.
DAQ	Data acquisition system	Système d'acquisition des données	Système pour récolter les signaux fournis par un détecteur, les assembler en événements et les stocker dans des fichiers.
	Data centric	Orienté données	Système informatique qui a pour but principal l'élaboration et l'analyse de grandes quantités de données.
	Data challenges	Défi des données	Exercice à grande échelle de la Grille de calcul de LHC pour en vérifier la préparation à la gestion des données.
	Data cloud	Nuage de données	Nuage de calcul capable de gérer et élaborer de grandes quantités de données.
	Data driven	Piloté par les données	Programme qui est piloté par les données en entrée.
DRM	Data readout module	Module de lecture des données	
	Data set	Ensemble de données	
DIS	Deep inelastic scattering	Diffusion profondément inélastique ou diffusion très dures	Diffusion dans laquelle une grande quantité d'énergie est transférée de la particule projectile à la particule cible.
DCS	Detector control system	Système de contrôle du détecteur	Système pour le contrôle et la surveillance des dispositifs physiques du détecteur indispensables pour son fonctionnement.
DDL	Detector data link	Connexion à fibre optique	
	Detector security system	Système de sûreté du détecteur	
	Digit	Signal numérisé ou Valeur numérisé	Signal électronique analogique transformé en signal numérique.
DMA	Direct memory access	Accès direct à la mémoire	Quand un périphérique peut accéder directement aux données stockées dans la mémoire centrale d'un ordinateur.
DAG	Directed acyclic graph	Graphe orienté acyclique	Un ensemble de sommets (ou nœuds) connectés par des arcs orientés qui ne forment pas des cycles.
DCC	Disoriented chiral condensate	Condensat chiral désorienté	État hypothétique produit dans la phase initiale de non-équilibre des collisions d'ions lourds, et constitué par plusieurs domaines avec une valeur propre non nulle du champ de

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
	Downscaling	Réduction (du taux d'événements)	pion, qui ensuite se désintègrent dans le vide QCD normal en rayonnant des pions mous.
	Electromagnetic calorimeter	Calorimètre électromagnétique	Instrument pour mesurer la quantité d'énergie d'une particule, optimisé pour la détection des photons et des leptons.
	Electronic logbook	Cahier de bord électronique	Logiciel pour enregistrer et présenter l'historique du processus de prise de données d'une expérience.
	Event building network	Réseau d'assemblage des événements	Réseau du système d'acquisition des données de ALICE sur lequel sont échangés les événement partiels entre concentrateurs de données locaux et globaux.
EDM	Event destination manager	Contrôleur de la destination des événements	Programme faisant partie du système d'acquisition des données de ALICE qui indique à quel concentrateur global les concentrateurs locaux de données doivent envoyer leurs événements partiels à assembler.
ESD	Event summary data	Données de Récapitulation de l'Événement	
ECS	Experiment control system	Système de contrôle de l'expérience	Système pour la coordination des activités du système de contrôle du détecteur, du système d'acquisition des données, du système de déclenchement et du déclencheur de haut niveau.
	Fake	Faux positif	Particule faussement identifiée.
	Fast-OR	OU logique rapide	Circuit électronique dans le détecteur SPD de ALICE qui fournit un signal si au moins un des pixels a enregistré le passage d'une particule.
	Feedback	Rétroaction	Processus dans lequel des informations du passé ou du présent influencent le même phénomène dans le présent ou l'avenir.
	Fermi sea	Mer de Fermi	État fondamentale pour des fermions libres a température nulle.
FPGA	Field-programmable gate array	Matrice de portes électroniques programmables	Circuit réalisant un réseau logique programmable composé de cellules logiques élémentaires librement assemblables.
	File replica catalogue	Catalogue des répliques des fichiers	Catalogue contenant les informations sur les fichiers d'un système informatique distribué, comme le nom et la location des différentes copies et les droits d'accès.

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
	Fill	Remplissage (de l'accélérateur)	Injection des trains de particules dans un accélérateur jusqu'au nombre de trains souhaité.
	Finite state machine	Automate fini	Système constitué d'états et de transitions dirigé par un mot en entrée qui le fait passer d'état en état, suivant les transitions, à la lecture de chaque lettre.
	Fit	Ajustement	Recherche des valeurs des paramètres d'une fonction qui reproduisent au mieux les données.
	Flow	Écoulement	Modulation angulaire de la répartition dans l'espace des particules produites par une collision.
FEC	Front end card	Carte électronique frontale	Carte électronique pour le détecteur TPC qui contient les modules pour l'amplification, l'échantillonnage et la numérisation des signaux détectés.
FED	Front end device	Dispositif d'entrée	
FEE	Front end electronics	Circuit électronique frontal	Carte électronique du calorimètre électromagnétique de ALICE pour façonner, échantillonner à 10 MHz et numériser les signaux provenant de 32 éléments de détection.
FERO	Front end readout unit	Unité de lecture d'entrée	Circuit électronique du système d'acquisition des données de ALICE qui reçoit les données des détecteurs et les envoie aux concentrateurs locaux des données.
	Full detector simulation	Simulation de la réponse du détecteur complet	Programme qui simule la réponse d'un détecteur aux particules qui le traversent.
	Garbage collector	Éboueur de mémoire	Système pour détecter la mémoire vive non utilisée par un programme en exécution et la restituer au programme pour être utilisée dynamiquement.
GCS	Gas control system	Système de contrôle des gaz	
	General purpose detector	Détecteur universel	Détecteur pour un large éventail de types de particule de différentes énergies.
GDC	Global data concentrator	Concentrateur global de données	Ordinateur faisant partie du système d'acquisition des données de ALICE qui reçoit les événements partiels des concentrateurs locaux de données, les assemble en événements complets et les écrit dans des fichiers.
GTU	Global tracking unit	Unité globale de trajectographie	Circuit électronique du détecteur TRD qui produit le signal local de déclenchement et l'envoie au Processeur Central de Déclenchement.

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
GUID	Globally unique identifier	Identifiant global unique	Identifiant unique pour un composant logiciel (normalement un fichier).
	Glueball	Boule de gluons	Etat hypothétique composé uniquement de gluons, sans quarks de valence, interagissant via leur charge de couleur.
	Grid	Grille informatique	Système informatique distribué permettant l'accès à des ressources de calcul hétérogènes et de haut niveau.
	Hardware	Matériel	Ensemble des composants matériels d'un ordinateur, à distinguer des programmes qui ne gèrent que le fonctionnement.
	Heartbeat	Battement de cœur	Signal périodique envoyé par un programme pour confirmer qu'il marche.
HCP	High ceremony method	Procédé de haute cérémonie	Procédé hautement formel et structuré de conception et d'écriture des programmes.
HLT	High level trigger	Déclencheur de haut niveau	Dernier niveau de filtrage du déclencheur basé sur une ferme d'ordinateurs qui permet l'application d'algorithmes sophistiqués allant jusqu'à la reconstruction complète des événements.
	I/O	Entrée et sortie	
	I/O bound	Limité par la disponibilité de ressources d'entrée et sortie plutôt que de calcul	
IS	Information system	Système d'information	Système pour le stockage et la gestion des informations sur l'état d'une Grille et de ses centres de calcul.
	Interaction record	Liste d'interaction	Liste avec le comptage de toutes les configurations du déclencheur rencontrées pendant une session de prise de données.
	Job agent	Agent de tâche	Tâche par lots qui vérifie l'environnement d'exécution dans un centre de calcul avant de récupérer une charge de travail à partir d'une liste centrale.
	Job broker	Courtier des tâches	Logiciel qui, dans une Grille, choisi à quel centre de calcul envoyer une tâche par lots afin d'optimiser l'utilisation des ressources.
JDL	Job description language	Langage décrivant le flux de travail d'une tâche par lots	
	Job matching	Appariement de tâche	Processus par lequel une tâche par

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
	Job optimiser	Service d'optimisation des tâches	lots est assignée au centre de calcul le plus apte à l'exécuter.
	Job token	Identifiant d'une tâche ou Jeton de tâche	Système qui permet d'évaluer les caractéristiques d'une tâche par lots afin de déterminer le centre de calcul le plus apte à l'exécuter.
	Kinematic freeze-out	Gel cinématique	Identifiant unique d'une tâche par lots.
	Kink	Coude	Moment de la collision auquel les caractéristiques cinématiques des particules se figent.
	Lattice QCD	QCD sur réseau	Désintégration d'une particule chargée en une autre particule chargée détectée et des particules neutres invisibles.
	Leading edge	Front de montée	Technique non-perturbative pour résoudre les équations de la chromodynamique quantique de façon approximative en utilisant une théorie de jauge formulée sur une grille de points dans l'espace et le temps.
NLO	Next to leading order	Après l'ordre principal	Partie croissante d'un signal électrique en fonction du temps.
	Leapfrogging	Saute-mouton	Approximation numérique tenant compte du terme d'une expansion en série après l'ordre principale.
L0	Level 0 trigger	Déclenchement de niveau 0	
LCB	LHC computing board	Comité du calcul au LHC	Premier niveau de filtrage du déclencheur des données du détecteur.
LED	Light emission diodes	Diode à émission de lumière	
	Little endian	Petit-boutiste	Composant électronique semi-conducteur qui émet de la lumière au passage du courant.
LDC	Local data concentrator	Concentrateur local de données	Architecture d'ordinateur où l'octet moins significatif d'un mot est stocké en premier, à l'adresse d'octet inférieur.
LTU	Local trigger unit	Unité locale de déclenchement	Ordinateur faisant partie du système d'acquisition des données de ALICE qui reçoit les données numérisées des sous-détecteurs et les assemble en événements partiels.
LFN	Logical file name	Nom logique d'un fichier	Dispositif pour la génération du signal de déclenchement d'un détecteur.
LUS	Look up services	Service de recherche	Nom d'un fichier dans un espace de noms abstrait et global, géré par le catalogue des fichiers.
			Système qui permet la recherche

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français et découverte	Explication dynamique de services dans un réseau informatique.
LCP	Low ceremony processes	Procédés de petite cérémonie	Procédé informel et flexible de conception et d'écriture des programmes.
	Management board	Comité d'administration	
	Mass storage	Stockage de masse	Dispositif pour stocker et gérer des grandes quantités de données.
	Master job	Tâche maître	Tâche par lots contrôlant d'autres tâches par lots.
	Master-worker	Maître – esclave	Architecture logicielle où un élément de programme fournit du travail à d'autres éléments de programme.
	Material budget	Budget matériel	Epaisseur d'un matériau, mesurée en pourcentage de longueur de radiation.
	Memorandum of Understanding	Protocole d'entente	
	Memory leak	Fuite de mémoire	Perte de l'accès à une partie de la mémoire d'un programme, qui de fait devient inutilisable.
	Metadata	Métadonnées	Informations relatives à un paquet de données.
	Middleware	Intergiciel	Logiciel pour l'exploitation et la surveillance de la Grille.
MIP	Minimum ionizing particle	Particule au minimum d'ionisation	Particule voyageant dans la matière avec un $\beta\gamma$ autour de 3 où la perte d'énergie est minime.
MWPC	Monitoring	Surveillance	
	Multi wire proportional chamber	Chambre proportionnelle à fils multiples.	Détecteur composé d'un conteneur avec des fils chargés positivement par rapport aux parois et rempli d'un gaz approprié. Quand une particule chargée traverse le gaz, les électrons d'ionisation, sous l'effet du champ électrique, produisent une avalanche qui est détectée sous la forme d'un courant électrique dans le fil et qui est corrélée à l'énergie de la particule.
MCM	Multi-chip modules	Modules multi-puce	
MRPC	Multi-gap Resistive Plate Camber	Chambre résistive à interstices multiples	Chambre à plaques résistives avec des plaques additionnelles à potentiel flottant à l'intérieur de la chambre.
nfs	Network file system	Système de fichiers en réseau	Système permettant le partage de fichiers entre plusieurs ordinateurs sur un réseau local.
	Obfuscated	Obscurci (code)	Code écrit de façon à rendre difficile la compréhension de sa fonctionnalité.
OO	Object Oriented	Orientée Objet	Style de programmation qui s'effectue

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français (programmation)	Explication
	Offline	Hors-ligne	à travers la définition et la manipulation d'objets, c'est-à-dire des unités de programmation contenant à la fois des données et les fonctions qui agissent sur ces données.
OADB	Offline analysis data base	Base de données hors-ligne pour l'analyse	Élaboration des données asynchrone avec la prise des données. Système logiciel permettant le stockage et la récupération des paramètres pour l'analyse des données.
OCDB	Offline calibration data base	Base de données de calibrage hors-ligne	Système logiciel permettant le stockage et la récupération des données relatives au calibrage des détecteurs.
	Offline computing	Système de traitement hors-ligne	Système de traitement des données asynchrone avec la prise des données.
	On-line	En-ligne	De façon synchrone avec la prise des données.
	Online computing	Traitement en-ligne	Traitement des données synchrone avec la prise des données.
OGSA	Open grid service architecture	Architecture Ouverte des Services de Grille	Proposition d'une définition standardisée des services informatiques pour une Grille de calcul.
OGSI	Open grid service infrastructure	Infrastructure Ouverte des Services de Grille	Proposition d'une définition standardisée des agencements (architectures) des services informatiques pour une Grille de calcul.
	Open source	Code source libre	Logiciel distribué avec le code source avec une licence spécifique.
	Operating system	Système d'exploitation	Programme qui gère le fonctionnement d'un ordinateur et permet d'accéder à ses fonctionnalités.
	Overlaps	Superpositions de volumes solides	
	Package manager	Gestionnaire de paquets de logiciel	Programme pour la récupération et l'installation des paquets de logiciel.
	Pair programming	Programmation en couple	Activité d'écriture d'un programme par deux personnes travaillant devant le même écran.
PCA	Partition control agent	Agent de contrôle des partitions	Programme qui gère les partitions logiques du détecteur ALICE pour l'acquisition des données.
	Payload	Charge de travail	Description des opérations à effectuer par une tâche par lots.
P2P	Peer-to-peer	Poste-à-poste	Réseau informatique dans lequel chaque ordinateur peut agir en tant que client ou serveur pour les autres,

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
PC	Personal computers	Ordinateur personnel	permettant un accès partagé aux ressources (fichiers, périphériques et capteurs) sans un serveur central. Ordinateur de petite dimension qui peut être utilisé à la maison ou au bureau.
PFN	Physical file name	Nom physique d'un fichier	Nom réel d'une réplique (copie) d'un fichier sur un élément de stockage.
	Physics board	Comité pour la Physique	
	Physics working group	Groupe de travail de physique	
	Pile-up	Empilement	Détection de plusieurs collisions superposées, provenant du même train de particules ou de plusieurs trains différents.
	Pilot job	Tâche pilote	Voir <i>job agent</i>
POSIX	Plugin	Gestionnaire d'extension qui peut être inséré dans un programme	
	Portable Operating System Interface for UNIX	Interface portable au système d'exploitation UNIX	
	Postdocs	Jeunes docteurs de recherche	
	Power grid	Réseau de distribution électrique	
	Program librarian	Responsable de la bibliothèque des programmes	
	Proxy certificate	Certificat de procuration	Certificat électronique qui permet à un programme ou à un individu d'effectuer certaines opérations au nom du donneur de procuration.
	Proxy service	Service mandataire	Service qui effectue des opérations pour le compte d'une autre personne ou entité.
	Publisher-subscriber mechanism	Mécanisme de souscription-publication	Mécanisme de programmation en réseau où les composantes s'« abonnent » aux données dont ils ont besoin et « publient » sur le réseau les données qu'ils produisent.
QCD	Quantum chromo dynamics	Chromodynamique quantique	Théorie de l'interaction forte.
QGP	Quark gluon plasma	Plasma de quarks et gluons	État de la matière à haute température et densité constitué de quarks et gluons libres.
	Quenching	Étouffement	Résultat de l'interaction d'un jet de

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
RAW	Raw data	Données brutes	particules avec le milieu. Données non traitées tels qu'ils sont produites par le détecteur.
	Readout	Lecture des signaux des détecteurs	
RCU	Readout control unit	Unité de contrôle de lecture	Carte électronique du détecteur TPC qui récolte les signaux et, après formatage, les envoie au système d'acquisition des données.
ROI	Region of interest	Signaux autour des traces	Signaux contenus dans un cône ou cylindre autour de ceux identifiés comme laissés par une trajectoire reconstruite.
	Relativistic rise	Montée relativiste	Augmentation de la perte d'énergie pour des valeurs de $\beta\gamma$ supérieurs à 3.
	Release	Publication	Mise à disposition des utilisateurs d'une nouvelle version d'un programme.
	Replica	Réplique	Instance physique d'un fichier catalogué dans un espace abstrait de noms, géré par le catalogue des fichiers.
RTAG	Requirement technical assessment group	Groupe d'évaluation des exigences techniques	
RPC	Resistive-plate chamber	Chambre à plaques résistives	Chambre proportionnelle à fils multiples où les parois sont isolantes et la cathode est une couche de métal appliquée à l'extérieur des parois.
RTT	Round trip time	Temps d'allée et retour	Temps que met un signal pour aller d'un point à l'autre d'un réseau et revenir au point de départ.
	Run	Période de prise de données	Laps de temps pendant lequel les détecteurs sont actifs et enregistrent des données de façon ininterrompue.
	Sandbox	Bac-à-sable	Répertoire où des fichiers peuvent être récupérés ou créés afin d'être manipulés sans altérer les copies de référence.
	Scaler	Compteur	
	Scaling law	Invariance d'échelle	Relation entre entités physiques qui ne change pas avec l'énergie ou l'impulsion.
	Schema evolution	Évolution du schéma	Changement de la structure des données d'une classe C++, qui doit être pris en compte lors d'une opération d'entrée et sortie.
	Scoring	Accumulation des quantités à mesurer (détecteur)	Enregistrement des signaux créés par des particules dans les éléments actifs d'un détecteur pendant une

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
SOAP	Seed	Graine ou Semence	Point de passage (cluster) à partir duquel commence la reconstruction de la trajectoire d'une particule.
	Setup	Mise en route de l'expérience	
	Shadowing	Atténuation de la fonction d'onde des hadrons	Modification de la fonction d'onde des hadrons due à leur interaction réciproque à l'intérieur d'un noyau.
	Shared libraries	Librairies partagées	Fichier contenant des programmes au format binaire qui peuvent être partagés en mémoire par plusieurs programmes.
	Shear viscosity	Viscosité de cisaillement	Mesure de la résistance d'un fluide à la déformation par un effort tendant à l'entailler ou couper par glissement (cisaillement).
	Shuttle	Navette	Programme dans le système d'acquisition des données de ALICE chargé de collecter et stocker les informations de calibrage générées en ligne.
	Silicon strip detector	Détecteur en silicium à pistes	Détecteur composé de deux couches de pistes de silicium formant un angle entre elles.
	Simple object access protocol	Protocole simple d'accès aux objets	Protocole pour échanger des informations structurées dans la mise en œuvre des services de la Toile à l'aide de messages texte.
	Slats	Lames (pour le détecteur MUON)	
	Software engineering	Génie logiciel	Application systématique de méthodes pour la conception, le développement, le test et la documentation de logiciels, afin d'en rationaliser la production, le suivi et la qualité.
	Spectrometer	Spectromètre	Détecteur qui mesure les propriétés cinématiques des particules.
	Stack	Liste de particules en mémoire	
	Standard data taking year	Année standard de prise de données	Année de prise des données selon les prévisions du calendrier de la machine LHC.
	Statefull	Avec état propre	Système qui possède des paramètres internes pouvant avoir différentes valeurs définissant son état.
	Stateless	Sans état propre	Système qui ne possède pas de paramètres internes pouvant avoir différentes valeurs permettant de définir son état.

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
	Storage element	Élément de stockage	Élément d'une Grille de calcul qui abstrait la fonction de stockage des données.
SCADA	Supervisory controls and data acquisition	Superviseur de l'acquisition des données et du contrôle	Système de contrôle industriel contrôlé par ordinateur pour la surveillance et le contrôle des processus industriels physiques.
SMP	Symmetric multi-processing	Multiprocesseurs symétriques	Ordinateur composé de plusieurs unités de calcul identiques.
	Tag file	Fichier des étiquettes	Collection de résumés synthétiques des caractéristiques des événements.
PDS	Tape	Bande magnétique	Support pour stocker de grandes quantités de données de façon séquentielle et relativement économique, basé sur l'état de magnétisation d'une bande recouverte d'oxyde de fer.
	Technical board	Comité technique	
	Technical design report	Rapport de Conception Technique	
	Templates	Modèles	Technique de programmation générique en C++ qui permet d'utiliser la même expression pour des classes différentes.
TDS	Temporary data storage	Serveur de disque temporaire	Dispositif qui permet le stockage et l'accès à des données pendant un certain laps de temps.
	Test beam	Faisceau de test	Faisceau de particules utilisé pour tester les sous-détecteurs pendant leur développement et avant de les installer.
	Thread	Filière de calcul	Une séquence d'instructions qui peuvent être exécutées en parallèle avec d'autres.
TOF	Time of flight	Temps de vol	Temps que met une particule pour aller du point d'origine à un point donné dans le détecteur.
TOT	Time over threshold	Temps total au-dessus du seuil	Temps pendant lequel un signal électronique reste supérieur à un certain seuil.
TPC	Time projection chamber	Chambre à projection temporelle	Détecteur tridimensionnel de particules chargées composé d'une chambre proportionnelle à fils multiples à l'extrémité d'un volume gazeux dans lequel dérivent les électrons d'ionisation sous l'effet d'un champ électrique.
TTC	Timing, Trigger and Control	Temporisation, déclenchement et contrôle	Dispositif électronique qui permet aux détecteurs de recevoir et interpréter les signaux provenant du processeur

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
	Toolkit	Boîte à outils	central de déclenchement. Ensemble de composants logiciels qui peuvent être utilisés indépendamment les uns des autres.
TRD	Transition radiation detector	Détecteur de la radiation de transition	Détecteur de la radiation émise par une particule chargée lorsque elle traverse l'interface entre deux matériaux.
	Trigger	Déclencheur	Système logiciel et matériel qui décide quels événements seront enregistrés et lesquels seront écartés.
TRU	Trigger Region Unit card	Circuit de déclenchement régional	Carte électronique qui, à partir des signaux enregistrés par un module du calorimètre électromagnétique de ALICE, génère un signal de déclenchement.
	Trust model	Modèle de confiance	Système pour établir l'authenticité de la connexion entre une clé cryptographique et son propriétaire.
UPC	Ultra peripheral collision	Collision ultrapériphérique	Collision d'ions lourds à très grand paramètre d'impact où les ions interagissent par des collisions photon-ion et photon-photon.
	Unit test	Test unitaire	Test portant sur une partie isolée d'un programme.
ufs	UNIX file system	Système de fichiers UNIX	Partie du système d'exploitation UNIX pour l'accès et la gestion des fichiers.
	Very large databases	Bases de données extra-larges	Système logiciel permettant le stockage et la récupération de très grandes quantités de données.
VMC	Virtual MonteCarlo	Interface MonteCarlo Virtuel	
VO	Virtual organisation	Organisation virtuelle	Ensemble d'individus identifiés par des certificats électroniques partageant des ressources communes dans une Grille ou un Nuage de calcul.
VOBox	Virtual organisation box	Boîte de l'organisation virtuelle	Ordinateur contenant tous les services nécessaires pour connecter un centre de calcul à la Grille.
	Wakeup signal	Signal de réveil	Signal qui active un dispositif électronique.
	Warning	Avertissement	Message émis par un programme informatique pour signaler une situation d'erreur n'empêchant pas la continuation de l'exécution.
	Wavelength shifting fiber	Fibre optique à décalage de longueur d'onde	Fibre optique photo-fluorescente qui absorbe les photons et émet des photons de fréquence plus basse qui sont plus facilement détectables.
	Web	Toile	Système hypertextuel permettant de visiter des informations stockées sur

Traitement des données de l'expérience ALICE

Acronyme	Anglais	Français	Explication
	Web services resource framework	Infrastructure des ressources des services de la Toile	des serveurs reliés par le réseau Internet. Une famille de spécifications pour les services de la Toile publiées par l'organisation pour la promotion de normes de l'information structurée (Organization for the Advancement of Structured Information Standards, OASIS).
WAN	Wide area networks	Réseaux à grande distance	Réseau informatique pour la transmission des données sur de grandes distances.
WMS	Workload management system	Gestionnaire de charge de travail	Système pour répartir la charge de travail par lots dans un système informatique distribué de Grille.
	Wrapped	Emballé (code)	Envelopper la fonctionnalité et les données d'un programme par un autre programme qui gère l'accès aux données et à la fonctionnalité du code originaire, via une nouvelle interface.
	Zombie	Tâche mort vivant	Tâche qui ne répond plus aux signaux mais qui continue à être présent dans le système de gestion des tâches par lots.

9 Références

- 1 C. G. Jung, Psychologie et alchimie, Buchet Chastel (2 janvier 2004)
- 2 Jung, C. G., (1934–1954). The Archetypes and The Collective Unconscious. (1981 2nd ed. Collected Works Vol.9 Part 1), Princeton, N.J.: Bollingen. ISBN 0-691-01833-2.
Jung, C. G. (1955–56). From "The Conjunction," *Mysterium Coniunctionis*, Collected Works, XIV, New Jersey: Princeton University Press.
- 3 M. Gell-Mann. Symmetries of Baryons and Mesons. *Phys. Rev.* 125, 1067–1084 (1962)
- 4 <http://www.slac.stanford.edu/>
- 5 Voir par exemple :
The Trouble With Physics: The Rise of String Theory, the Fall of a Science, and What Comes Next. Houghton Mifflin, 2006. ISBN 978-0-618-55105-7
et aussi la controverse Smolin – Susskind relaté dans
http://www.edge.org/3rd_culture/smolin_susskind04/smolin_susskind.html
- 6 Ariew, Roger (1976). Ockham's Razor: A Historical and Philosophical Analysis of Ockham's Principle of Parsimony. Champaign-Urbana, University of Illinois.
- 7 Voir par exemple L. Smolin, *The case for background independence*, 2005, arXiv:hep-th/0507235v1
- 8 G. Veneziano. Construction of a crossing-symmetric, Regge-behaved amplitude for linearly rising trajectories. *Il Nuovo Cimento A* 1 Settembre 1968, Volume 57, Issue 1, pp 190-197.
Voir par exemple J.H. Schwarz. Introduction to Superstring Theory. arXiv:hep-ex/0008017v1 9 Aug 2000
- 9 Voir le matériel qui se trouve à
<http://www.alainconnes.org/en/downloads.php>
- 10 G. Aad et al., Combined search for the Standard Model Higgs boson using up to 4.9 fb⁻¹ of pp collision data at $\sqrt{s} = 7$ TeV with the ATLAS detector at the LHC, *Physics Letters B*, 710 (1) 2012. 49-66
S. Chatrchyan et al., Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC, *Physics Letters B*, 716 (1) 2012, 30–61
- 11 F. Englert and R. Brout, « Broken Symmetry and the Mass of Gauge Vector Mesons », *Physical Review Letters*, 13, 1964, 321-323
P. W. Higgs, « Broken Symmetries and the Masses of Gauge Bosons », *Physical Review Letters*, 13, 1964, 508-509
G. S. Guralnik, C. R. Hagen, and T. W. B. Kibble, « Global Conservation Laws and Massless Particles », dans *Physical Review Letters*, vol. 13, 1964, p. 585-587 [lien DOI [archive]]

- 12 Kurt Gottfried, Victor F. Weisskopf, « Concepts of Particle Physics: Volume II » Oxford University Press, USA (November 13, 1986), pp. 339, ISBN: 0195033930
- 13 L. De Broglie. Recherches sur la théorie des quanta (Researches on the quantum theory), Thesis, Paris, 1924, Ann. de Physique (10) 3, 22 (1925)
- 14 <http://lh.web.cern.ch/lhc/>
- 15 T.D.Lee, Rev. Modern Phys., Vol 47, (1975) 267
- 16 Voir par exemple <http://en.wikipedia.org/wiki/Emergence> et les references citées.
- 17 D.J. Gross, F. Wilczek (1973). "Ultraviolet behavior of non-abelian gauge theories". Physical Review Letters 30 (26): 1343–1346. Bibcode 1973PhRvL..30.1343G. doi:10.1103/PhysRevLett.30.1343.
H.D. Politzer (1973). "Reliable perturbative results for strong interactions". Physical Review Letters 30 (26): 1346–1349. Bibcode 1973PhRvL..30.1346P. doi:10.1103/PhysRevLett.30.1346.
- 18 See for instance R. Gupta, Introduction to Lattice QCD, Lectures given at the LXVIII Les Houches Summer School "Probing the Standard Model of Particle Interactions", July 28-Sept 5, 1997. 150 pages, arXiv:hep-lat/9807028v1
- 19 J.D. Bjorken (1969). "Asymptotic Sum Rules at Infinite Momentum". Physical Review 179: 1547–1553. doi:10.1103/PhysRev.179.1547
- 20 Karsch F., Laermann E. and Peikert A. 2000 Phys. Lett. B **478** 447
- 21 Wetzorkel et al 2002 Nucl. Phys. Proc. Suppl. **106** 510
- 22 K. Aamodt *et al.*, *Charged-Particle Multiplicity Density at Midrapidity in Central Plomb-Plomb Collisions at $\sqrt{s_{NN}}=2.76$ TeV*, *Phys. Rev. Lett.* **105**, 252301 (2010)
- 23 Braun-Munzinger P and Stachel J 1996 *Nucl.Phys.A* **606** 320
Heinz U 1998 *Nucl. Phys. A* **638** 357c
Heinz U 1999 *J. Phys. G: Nucl. Part. Phys.* **25** 263
Heinz U 1999 *Nucl. Phys. A* **661** 140c
Stock R 1999 *Phys. Lett. B* **456** 277
Stock R 1999 *Prog. Part. Nucl. Phys.* **42** 295
- 24 K. Aamodt *et al.*, Two-pion Bose–Einstein correlations in central Pb–Pb collisions at $\sqrt{s_{NN}} = 2.76$ TeV, *Physics Letters B*, 696 (4) 2011, 328–337
- 25 Bialas A and Koch V 1999 *Phys. Lett. B* **456** 1
- 26 Kienzle W *et al* [TOTEM Collaboration] 1999 *TOTEM, Total Cross Section, Elastic Scattering and Diffraction Dissociation at the LHC*, Technical Proposal CERN-LHCC-99-007, LHCC-P-5
- 27 Baltz A J, Chasman C and White S N 1998 *Nucl. Instrum. Methods A* **417** 1

- 28 The ALICE Collaboration, Measurement of inelastic, single- and double-diffraction cross sections in proton–proton collisions at the LHC with ALICE, <http://arxiv.org/pdf/1208.4968v1.pdf>
- 29 ALICE Collaboration 1995 *Technical Proposal* CERN/LHCC/95–71
 ALICE Collaboration 1996 *Technical Proposal Addendum 1*, CERN/LHCC/96–32
 ALICE Collaboration 1999 *Technical Proposal Addendum 2*, CERN/LHCC/99–13
 ALICE Collaboration 1998 *Technical Design Report of the High-Momentum Particle Identification Detector* CERN/LHCC/1998–19
 ALICE Collaboration 1999 *Technical Design Report of the Photon Spectrometer* CERN/LHCC/1999–04
 ALICE Collaboration 1999 *Technical Design Report of the Zero-Degree Calorimeter* CERN/LHCC/1999–05
 ALICE Collaboration 1999 *Technical Design Report of the Inner Tracking System* CERN/LHCC/1999–12
 ALICE Collaboration 1999 *Technical Design Report of the Forward Muon Spectrometer* CERN/LHCC/1999–22
 ALICE Collaboration 2000 *Technical Design Report of the Forward Muon Spectrometer Addendum-1*, CERN/LHCC/2000–46
 ALICE Collaboration 1999 *Technical Design Report of the Photon Multiplicity Detector* CERN/LHCC/1999–32
 ALICE Collaboration 2003 *Technical Design Report for the Photon Multiplicity Detector Addendum-1*, CERN/LHCC 2003–038
 ALICE Collaboration 2000 *Technical Design Report of the Time-Projection Chamber* CERN/LHCC/2000–01
 ALICE Collaboration 2000 *Technical Design Report of the Time-Of-Flight Detector* CERN/LHCC/2000–12 Addendum CERN/LHCC/2002–16
 ALICE Collaboration 2001 *Technical Design Report of the Transition-Radiation Detector* CERN/LHCC/2001–21
 ALICE Collaboration 2004 *Technical Design Report of the Forward Detectors* CERN/LHCC/2004–025
 ALICE Collaboration 2008 *Technical Design Report of the Electromagnetic Calorimeter* CERN-LHCC-2008-014
 ALICE Collaboration 2003 *Technical Design Report of the Trigger, Data Acquisition, High Level Trigger and Control System* CERN/LHCC/2003–062
- 30 Y. Hori et al., Charge dependent mixed harmonic azimuthal correlations in Pb-Pb collisions at $\sqrt{s_{NN}} = 2.76$ TeV, Proceedings of the Quark Matter 2012 International Conference, Washington, August 13-18, 2012.
- 31 <http://tools.ietf.org/html/rfc1122>
- 32 CERN ALICE DAQ Group 2002 ALICE Internal Note 2002-36
- 33 ALICE Technical Design Report of the Computing, CERN-LHCC-2005-018, ALICE TDR 012, 15 June 2005
- 34 <http://aliweb.cern.ch/Offline>
- 35 In Proc. Symp. On Advanced Computer Programs for Digital Computers, sponsored by ONR. Republished in Annals of the History of Computing, Oct.

- 1983, pp. 350-361. Reprinted at ICSE'87, Monterey, California, USA, March 30-April 2, 1987, June 1956.
- 36 In P.Naur and B.Randell, editors, Software Engineering: Reports of a conference sponsored by the NATO Science Committee, Garmish, Germany, 7-11 October, 1968.
- 37 <http://www.yourdon.com>
- 38 Peter P. Chen. The entity-relationship model - toward a unified view of data. ACM Trans. Database Syst., 1(1):9-36, 1976.
- 39 Grady Booch. Object-Oriented Analysis and Design with Applications. Benjamin-Cummings, Redwood City, Calif., 2nd edition, 1994.
- 40 J. Rumbaugh, M. Blah, W. Premerlani, F. Eddy, and W. Lorensen. Object-Oriented Modeling and Design. Prentice Hall, Englewood Cliffs, 1991.
- 41 Sally Shlaer and Stephen J Mellor. Object lifecycles modeling the world in states. Yourdon Press computing series. Yourdon, Englewood Cliffs, NJ, 1992.
- 42 Guide to the software engineering standards, issue 1. Eur. Space Agency, Noordwijk, 1991.
- 43 <http://www.uml.org>
- 44 <http://adamo.web.cern.ch>
- 45 <http://www.ilogix.com>
- 46 <http://sdt.web.cern.ch>
- 47 <http://omtools.com>
- 48 <http://www.aonix.com/stp.html>
- 49 <http://www.rational.com>
- 50 <http://cernlib.web.cern.ch/cernlib>
- 51 <http://paw.web.cern.ch/paw>
- 52 <http://wwwasdoc.web.cern.ch/wwwasdoc/pdfdir/geant.pdf>
- 53 <http://Root.cern.ch>
- 54 Voir par exemple DiBona, C, Ockman, S, Open Sources. Voices from the Open Source Revolution. O'Reilly Media, January 1999
- 55 Voir <http://cernlib.web.cern.ch/cernlib/>
- 56 <http://geant4.cern.ch/>
- 57 <http://wwwasd.web.cern.ch/wwwasd/lhc++/indexold.html>
- 58 Beck, K. Extreme Programming Explained. Addison-Wesley, Boston, 2000.
- 59 <http://www.doxygen.org>

- 60 P. Tonella, S.L. Abebe, Code Quality from the Programmer's Perspective, In Proceedings of Science, XII Advanced Computing and Analysis Techniques in Physics Research, Erice, Italy, 2008.
- 61 Booch, G, Rumbaugh, J, Jacobson, I (2000). Le guide de l'utilisateur UML (ISBN 2-212-09103-6)
- 62 IEC 60559:1989, Binary Floating-Point Arithmetic for Microprocessor Systems (anciennement IEC 559:1989)
- 63 P. Buncic , C. Aguado Sánchez , J. Blomer , A. Harutyunyan and M. Mudrinic, A practical approach to virtualization in HEP, Eur. Phys. J. Plus (2011) 126: 13, DOI: 10.1140/epjp/i2011-11013-1
- 64 <http://agilemanifesto.org>
- 65 Kent Beck. Extreme Programming Explained. Addison-Wesley, Boston, 2000.
- 66 <http://www.opensource.org>
<http://www.sourceforge.com>
<http://www.gnu.org>
- 67 <http://aliweb.cern.ch/Offline/>
- 68 <http://aliweb.cern.ch/Offline>
- 69 <http://subversion.apache.org/>
Pilato, M., Collins-Sussman, B., Fitzpatrick, B., Version Control with Subversion. O'Reilly Media; Second Edition edition (September 30, 2008)
- 70 <http://www.dwheeler.com/sloccount/>
- 71 Barry Boehm. Software engineering economics. Englewood Cliffs, NJ:Prentice-Hall, 1981. ISBN 0-13-822122-7
- 72 <http://aliweb.cern.ch/Offline/AliRoot/Coding-Conventions.html>
- 73 P. Tonella, S.L. Abebe, Code Quality from the Programmer's Perspective, In Proceedings of Science, XII Advanced Computing and Analysis Techniques in Physics Research, Erice, Italy, 2008.
- 74 <http://www.coverity.com/>
- 75 <http://www.cmake.org/>
<http://www.cmake.fr/>
- 76 <https://savannah.cern.ch/>
- 77 <http://gna.org/projects/savane/>
- 78 <http://valgrind.org/>
- 79 <http://code.google.com/p/google-perftools/>
- 80 Brad J. Cox, Andrew J. Novobilski (1986). Object-Oriented Programming: An Evolutionary Approach, ISBN 0-201-54834-8.
Bertrand Meyer (2000). Conception et programmation orientées objet, ISBN 2-212-09111-7.

- Grady Booch, James Rumbaugh, Ivar Jacobson (2000). Le guide de l'utilisateur UML, ISBN 2-212-09103-6
- 81 Brun, R, Carminati, F, Galli Carminati, G (Eds.). From the Web to the Grid and beyond. Computing Paradigms Driven by High Energy Physics. Springer 2011, ISBN 978-3-642-23156-8
- 82 <http://na49info.web.cern.ch/na49info/>
- 83 <http://Root.cern.ch/>
- 84 Bjarne Stroustrup, The Design and Evolution of C++, Addison-Wesley Professional; 1 edition (April 8, 1994)
- 85 <http://www-public.slac.stanford.edu/babar/>
- 86 <http://www-zeuthen.desy.de/CHEP97/paper/502.ps>
- 87 <http://www-zeuthen.desy.de/CHEP97/chep97.htm>
- 88 <http://Root.cern.ch>
- 89 <http://llvm.org>
- 90 *"The FLUKA code: Description and benchmarking"* G. Battistoni, S. Muraro, P.R. Sala, F. Cerutti, A. Ferrari, S. Roesler, A. Fassò, J. Ranft, Proceedings of the Hadronic Shower Simulation Workshop 2006, Fermilab 6--8 September 2006, M. Albrow, R. Raja eds., AIP Conference Proceeding 896, 31-49, (2007)
"FLUKA: a multi-particle transport code" A. Ferrari, P.R. Sala, A. Fassò, and J. Ranft, CERN-2005-10 (2005), INFN/TC_05/11, SLAC-R-773
- 91 http://en.wikipedia.org/wiki/Constructive_solid_geometry
- 92 Hřivnáčová I et al 2003 Proc. of Computing in High Energy and Nuclear Physics (La Jolla) pp THJT006
Carminati et al 2004 Proc. of Computing in High Energy and Nuclear Physics (Interlaken) pp 433
- 93 http://Root.cern.ch/Root/html/doc/GEOM_GEOM_Index.html
- 94 ALICE-INT-2009-001
- 95 F. Riggi et al., ALICE-INT-2001-26.
- 96 P. Billoir; NIM A225 (1984) 352
P. Billoir et al.; NIM A241 (1985) 115,
R. Fruhwirth; NIM A262 (1987) 444,
P.Billoir;. CPC (1989) 390
- 97 ALICE Collaboration, Technical Design Report of the Time-Projection Chamber, CERN/LHCC/2000-01.
- 98 ALICE Physics Performance Report CERN / LHCC 2005-049, Volume 2 (5 Décembre 2005)].
- 99 M. Aguilar-Benitez *et al.*, Z. Phys. **C50** (1991) 405–426.

- 100 G. O. V. de Barros, Bo Fenton-Olsen, Peter Jacobsb, Mateusz Płoskoń, Data-driven analysis methods for the measurement of reconstructed jets in heavy ion collisions at RHIC and LHC Nuclear Physics A 00 (2012) 1–5
- 101 S. Bagnasco, L. Betev, P. Buncic, F. Carminati, C. Cirstoiu, C. Grigoras, A. Hayrapetyan, A. Harutyunyan, A.J. Peters, P. Saiz, *AliEn: ALICE Environment on the GRID*, Proc. Conf. for Computing in High-Energy and Nuclear Physics (CHEP), Victoria, Canada, 2-9 Sep 2007
- 102 M. Ballintijn, G. Roland, R. Brun and F. Rademakers, *The PROOF distributed parallel analysis framework based on Root*, Proc. Conf. for Computing in High-Energy and Nuclear Physics (CHEP), La Jolla, California, 24-28 Mar 2003
- 103 M. Tadel, EVE - Event Visualization Environment of the ROOT framework, Proceedings of XII Advanced Computing and Analysis Techniques in Physics Research November 3-7, 2008, Erice, Italy
- 104 M. Tadel, Raw-data Display and Visual Reconstruction Validation in ALICE, International Conference on Computing in High Energy and Nuclear Physics (CHEP'07), Journal of Physics: Conference Series, 119 (2008) 032036, doi:10.1088/1742-6596/119/3/032036
- 105 M. Tadel and A. Mrak-Tadel, XV Int. Conf. on Comp. in High Energy and Nucl. Phys. 2006, 1 (Mumbai: Macmillan) pp 398–401.
- 106 M. Tadel, 2007 The New Generation of OpenGL Support in ROOT, J.Phys.Conf.Ser. 119:042028, 2008. <http://dx.doi.org/10.1088/1742-6596/119/4/042028>
- 107 <http://www.opengl.org>
- 108 M. Tadel GLED – an Implementation of a Hierarchic Server–Client Model, (Advances in Computation: Theory and Practice vol 16) ed Pan Y and Yang L (New York: Nova Science Publishers) pp 21–37. See also <http://www.gled.org/>
- 109 D Kovalskyi et al., Fireworks: A Physics Event Display for CMS, 17th International Conference on Computing in High Energy and Nuclear Physics (CHEP09), Journal of Physics: Conference Series 219 (2010) 032014 doi:10.1088/1742-6596/219/3/032014
- 110 I. Antcheva, R. Brun, C. Hof and F. Rademakers, The Graphics Editor in ROOT, Nucl.Inst.&Meth.Phys.Res. 1 559 pp 17–21.
- 111 H. Drevermann, D. Kuhn and B. Nilsson, Event Display: Can We See What We Want to See?, Presented at CERN School of Computing '95, Arles, France. <http://ipt.web.cern.ch/IPT/Papers/CSC95/EDisplay/>
- 112 « The Metacomputer: One from Many », NCSA (1995) <http://archive.ncsa.uiuc.edu/Cyberia/MetaComp/MetaHome.html>
- 113 Overview of the I-WAY: Wide-Area Visual Supercomputing – Thomas A Defanti, Ian Foster, Michael E Papka, Rick Stevens, Tim Kuhfuss – The International Journal of Supercomputer Applications and High Performance

- Computing, Vol. 10, No. 2/3. (Summer - Fall 1996), pp. 123- 131 – MIT Press.
- 114 <http://www.globus.org>
- 115 « Globus: a Metacomputing Infrastructure Toolkit », I. Foster and C. Kesselman (1997).
<ftp://ftp.globus.org/pub/globus/papers/globus.pdf>
- 116 « The Grid: Blueprint for a New Computing Infrastructure »
<http://portal.acm.org/citation.cfm?id=289914>
- 117 Ian Foster, Carl Kesselman and Steven Tuecke, « The Anatomy of the Grid: Enabling Scalable Virtual Organizations », Lecture Notes in Computer Science **2150** (2001) 1
- 118 Randy Butler, Von Welch, Douglas Engert, Ian Foster, Steven Tuecke, John Volmer and Carl Kesselman, « A National-Scale Authentication Infrastructure », Computer **33 12** (2000) 60-66
- 119 http://press.web.cern.ch/press/PressReleases/Releases1994/PR16.94E_LHC-Council.html
- 120 B.Carpenter et al., The Muscle Report: The Computing Needs Of The Lep Experiments. Jan 1988, CERN-DD/88/1
- 121 Report of the Steering Group of the LHC Computing Review – S. Bethke (Chair), M. Calvetti, H.F. Hoffmann, D. Jacobs, M. Kasemann, D. Linglin – CERN/LHCC/2001-004, CERN/RRB-D 2001-3, 22 February 2001.
- 122 <http://monarc.web.cern.ch/MONARC/>
- 123 <http://chep2000.pd.infn.it>
- 124 <http://eu-datagrid.web.cern.ch/eu-datagrid/>
- 125 The Particle Physics Data Grid – <http://www.ppdg.net/>
 Grid Physics Network – GriPhyN Project - <http://www.griphyn.org/>.
 The Virtual Data Toolkit – <http://www.lsc-group.phys.uwm.edu/vdt/>
 NorduGrid – <http://www.nordugrid.org>
 INFN Grid - <http://www.italiangrid.it>
 GridPP - <http://www.gridpp.ac.uk>
- 126 <http://alien.cern.ch>
- 127 <https://twiki.cern.ch/twiki/bin/view/Atlas/PanDA>
- 128 A.~Tsaregorodtsev *et al.*, « DIRAC - Distributed infrastructure with Remote Agent Control », Proceedings of 2003 Conference for Computing in High-Energy and Nuclear Physics (CHEP 03), La Jolla, California, 24-28 Mar 2003, pp TUAT006, [arXiv:cs/0306060].
- 129 F.Carminati et al, *LHC requirements for GRID middleware*, présentation à CHEP 2003, International Conference on Computing in High Energy and Nuclear Physics, March 24-28, La Jolla, California, USA, voir aussi

- <http://www.slideworld.com/slideshows.aspx/LHC-requirements-for-GRID-middleware-ppt-1259837>
- 130 *Common Use Cases for a HEP Common Application Layer (HEPCAL)*, – F.Carminati (editor) – May 2002 – CERN-LHC-SC2-20-2002
Common Use Cases for a HEP Common Application Layer (HEPCAL II), – F.Carminati (editor) – June 2003 – CERN-LHC-SC2-32-2003
<http://lcg.web.cern.ch/LCG/SC2/cern-sc2/cern-sc2-2003/CERN-LCG-2003-32.doc>
- 131 <http://www.eu-egee.org>
- 132 <http://www.egi.eu/community/collaborations/EMI.html>
- 133 <http://www.opensciencegrid.org>
- 134 Voir pas exemple http://www.w3schools.com/soap/soap_intro.asp
J. Snell, D. Tidwell, P. Kulchenko. *Programming Web Services With SOAP*, O'Reilly Media; 1 edition (December 15, 2001)
- 135 P. Buncic, A. J. Peters and P. Saiz, *The AliEn system, status and perspectives*, In the Proceedings of 2003 Conference for Computing in High-Energy and Nuclear Physics (CHEP 03), La Jolla, California, 24-28 Mar 2003, pp MOAT004, [arXiv:cs/0306067].
- 136 ARDA: Architectural Roadmap towards Distributed Analysis
<http://lcg.web.cern.ch/LCG/SC2/cern-sc2/cern-sc2-2003/CERN-LCG-2003-33.pdf>
- 137 EGEE Middleware Architecture and Planning, (Release 1)
<https://edms.cern.ch/document/476451/>
- 138 <http://monalisa.caltech.edu/monalisa.htm>
- 139 <http://xRootd.slac.stanford.edu>
- 140 <http://tools.ietf.org/html/rfc2459>
<http://tools.ietf.org/html/rfc5280>
- 141 S. Ponce et al., Status and evolution of CASTOR (Cern Advanced STORage), proceedings of the Computing in High Energy and Nuclear Physics (CHEP) 2012 conference
- 142 T. Mkrtchyan et al., dCache: implementing a high-end NFSv4.1 service using a Java NIO framework, proceedings of the Computing in High Energy and Nuclear Physics (CHEP) 2012 conference
- 143 R. Brito da Rocha et al., DPM: Future-proof storage, proceedings of the Computing in High Energy and Nuclear Physics (CHEP) 2012 conference
- 144 <http://projects.fnal.gov/SAMGrid>
- 145 <http://www-ccf.fnal.gov/enstore/>
- 146 <https://sdm.lbl.gov/srm-wg/>
- 147 A.Fella et al., A Comparison of Data-Access Platforms for BaBar and ALICE analysis Computing Model at the Italian Tier1, Journal of Physics:

- Conference Series 219 (2010) 072003 doi:10.1088/1742-6596/219/7/072003
F.Furano et al., Scalla/xRootd WAN globalization tools: where we are.
Journal of Physics: Conference Series 219 (2010) 072005
doi:10.1088/1742-6596/219/7/072005
- 148 C.Grigoras et al., Optimization of Grid Resources Utilization: QoS- aware client to storage connection in AliEn, 13th International Workshop on Advanced Computing and Analysis Techniques in Physics Research - ACAT 2010 Jaipur, India February 22–27 2010
- 149 <http://trends.google.com>
- 150 <http://research.cs.wisc.edu/condor/classad/>
- 151 <http://www.bittorrent.com>
- 152 I.Legrand et al., MonALISA: An agent based, dynamic service system to monitor, control and optimize distributed systems, Computer Physics Communications 180 (2009) 2472–2498
- 153 I.Legrand et al., Monitoring and Control of Large Systems with MonALISA, Queue - Data Queue Homepage archive, 7 (6) 2009, 40, ACM New York, NY, USA, doi>10.1145/1563821.1577839
- 154 <http://monalisa.caltech.edu>
- 155 I.C. Legrand, H.B. Newman, R. Voicu, C. Cirstoiu, C. Grigoras, M. Toarta, C. Dobre, MonALISA: An agent based, dynamic service system to monitor, control and optimize grid based applications, in: CHEP04, Interlaken, Switzerland, 2004.
- 156 What is Amazon Elastic Compute Cloud Amazon EC2 :
<http://aws.amazon.com/ec2/faqs>
- 157 F. Antinori, A. Billmeier and J. Zaraneek, AIP Conf. Proc. 631:294 (2003).
- 158 Müller B, Nagle JL, Ann.Rev.Nucl.Part.Sci. 56:93 (2006), nucl-th/0602029.
- 159 Kovtun P, Son D, Starinets A, Phys.Rev.Lett. 94:111601 (2005), hep-th/0405231.
- 160 ALICE Collaboration, Aamodt K, et al., Phys.Rev.Lett. 105:252301 (2010), 1011.3916.
- 161 Bjorken J, Phys.Rev. D27:140 (1983).
- 162 CMS Collaboration, Krajczar K, J.Phys.G 38:124041 (2011), CMS-PAS-HIN-11-003.
- 163 PHOBOS Collaboration, Back BB, et al., Nucl. Phys. A757:28 (2005), nucl-ex/0410022.
- 164 BRAHMS Collaboration, Arsene I, et al., Nucl. Phys. A757:1 (2005), nucl-ex/0410020.
- 165 PHENIX Collaboration, Adcox K, et al., Nucl. Phys. A757:184 (2005), nucl-ex/0410003.

- 166 STAR Collaboration, Adams J, et al., Nucl. Phys. A757:102 (2005), nucl-ex/0501009.
- 167 Czyż W, Maximon LC. Ann. Phys. 52:59 (1969)
- 168 Glauber RJ. In Lectures in Theoretical Physics, ed.WEBrittin, LG Dunham, 1:315. New York: Interscience (1959)
- 169 Bialas A, Bleszyński M, Czyż W. Acta Physiol. Pol. B 8:389 (1977)
- 170 PHENIX Collaboration, Adler S, et al., Phys.Rev. C71:034908 (2005), nucl-ex/0409015.
- 171 STAR Collaboration, Abelev B, et al., Phys.Rev. C79:034909 (2009), 0808.2041.
- 172 PHOBOS Collaboration, Alver B, et al., Phys.Rev. C83:024913 (2011), 1011.1940.
- 173 A. Dumitru, D. E. Kharzeev, E. M. Levin, and Y. Nara, "Gluon Saturation in pA Collisions at the LHC: KLN Model Predictions For Hadron Multiplicities," Phys.Rev. C85 (2012) 044920, arXiv:1111.3031 [hep-ph].
- 174 G. Barnafoldi, J. Barrette, M. Gyulassy, P. Levai, and V. Topor Pop, "Predictions for p+Pb at 4.4A TeV to Test Initial State Nuclear Shadowing at energies available at the CERN Large Hadron Collider," Phys.Rev. C85 (2012) 024903, arXiv:1111.3646 [nucl-th].
- 175 P. Tribedy and R. Venugopalan, "QCD saturation at the LHC: comparisons of models to p+p and A+A data and predictions for p+Pb collisions," Phys.Lett. B710 (2012) 125–133, arXiv:1112.2445 [hep-ph].
- 176 R. Xu, W.-T. Deng, and X.-N. Wang, "Nuclear modification of high-pT hadron spectra in p+A collisions at LHC," arXiv:1204.1998 [nucl-th].
- 177 J. L. Albacete, A. Dumitru, H. Fujii, and Y. Nara, "CGC predictions for p+Pb collisions at the LHC," arXiv:1209.2001 [hep-ph].
- 178 Bopp FW, Engel R, Ranft J, Roesler S, (2007), 0706.3875.
- 179 Deng WT, Wang XN, Xu R, Phys.Lett. B701:133 (2011), 1011.5907.
- 180 Gelis F, Iancu E, Jalilian-Marian J, Venugopalan R, Ann.Rev.Nucl.Part.Sci. 60:463 (2010), 1002.0333.
- 181 Albacete JL, Dumitru A, (2010), 1011.5161.
- 182 ALICE Collaboration, Aamodt K, et al., Pseudorapidity density of charged particles in p–Pb collisions at $\sqrt{s_{NN}} = 5.02$ TeV, arXiv:1210.3615v1 [nucl-ex] 12 Oct 2012
- 183 PHOBOS Collaboration, B. Back et al., "Pseudorapidity distribution of charged particles in d + Au collisions at $\sqrt{s_{NN}} = 200$ GeV," Phys.Rev.Lett. 93 (2004) 082301, arXiv:nucl-ex/0311009 [nucl-ex].
- 184 PHOBOS Collaboration, B. Alver et al., "Phobos results on charged particle multiplicity and pseudorapidity distributions in Au+Au, Cu+Cu, d+Au, and

- p+p collisions at ultra-relativistic energies," Phys.Rev. C83 (2011) 024913, arXiv:1011.1940 [nucl-ex].
- 185 Braun-Munzinger P, Redlich K, Stachel J, (2003), nucl-th/0304013, To appear in Quark Gluon Plasma 3, eds. R.C. Hwa and Xin-Nian Wang, World Scientific Publishing.
 - 186 Becattini F, (2009), 0901.3643.
 - 187 Salzwedel J, Two baryons correlations in heavy-ion collisions at the LHC, Proceedings of the Quark Matter 2012 International Conference, Washington, August 13-18, 2012.
 - 188 Schnedermann E, Sollfrank J, Heinz UW, Phys.Rev. C48:2462 (1993), nucl-th/9307020.
 - 189 STAR Collaboration, Adams J, et al., Nucl. Phys. A757:102 (2005), nucl-ex/0501009.
 - 190 ALICE Collaboration, Krzewicki M, et al., J.Phys.G 38:124047 (2011), 1107.0080.
 - 191 ALICE Collaboration, Yin Z, et al., (2012), 1202.0365.
 - 192 P. Huovinen, P. Kolb, U. W. Heinz, P. Ruuskanen, S. Voloshin. Phys.Lett. B503:58 (2001), hep-ph/0101136.
 - 193 Shen C, Heinz U, Huovinen P, Song H, Phys.Rev. C84:044903 (2011), 1105.3226.
 - 194 Heinz UW, Shen C, Song H, (2011), 1108.5323.
 - 195 Voloshin SA, Poskanzer AM, Snellings R, Landolt-Boernstein: Relativistic Heavy Ion Physics volume 1/23 (Springer-Verlag, 2010), pp. 5–54, 0809.2949.
 - 196 Voloshin S, Nucl.Phys. A715:379 (2003), nucl-ex/0210014.
 - 197 Fries R, Müller B, Nonaka C, Bass S, Phys.Rev.Lett. 90:202303 (2003), nucl-th/0301087.
 - 198 Fries R, Müller B, Nonaka C, Bass S, Phys.Rev. C68:044902 (2003), nucl-th/0306027.
 - 199 Greco V, Ko C, Levai P, Phys.Rev. C68:034904 (2003), nucl-th/0305024.
 - 200 LaPointe S, Open heavy-flavour measurements in pp and Pb-Pb collisions with ALICE at the LHC, arXiv:1209.6198v1 [nucl-ex] 27 Sep 2012
 - 201 Massacrier L, J/ψ elliptic flow measurement in Pb-Pb collisions at $\sqrt{s_{NN}} = 2.76$ TeV at forward rapidity with the ALICE experiment, Nuclear Physics A 00 (2012) 1–4, arXiv:1208.5401v2 [nucl-ex] 21 Sep 2012
 - 202 The ALICE Collaboration, Abelev B et al., Centrality Dependence of Charged Particle Production at Large Transverse Momentum in Pb–Pb Collisions at $\sqrt{s_{NN}} = 2.76$ TeV, arXiv:1208.2711v1 [hep-ex] 13 Aug 2012

- 203 ALICE Collaboration, Ortiz Velasquez A et al., Production of pions, kaons and protons at high p_T in $\sqrt{s_{NN}} = 2.76$ TeV Pb-Pb collisions, arXiv:1210.6995v1 [hep-ex] 25 Oct 2012
- 204 ALICE Collaboration, Grelli A, D meson nuclear modification factors in Pb-Pb collisions at $\sqrt{s_{NN}} = 2.76$ TeV with the ALICE detector, arXiv:1210.7332v1 [hep-ex] 27 Oct 2012
- 205 The ALICE Collaboration, Transverse Momentum Distribution and Nuclear Modification Factor of Charged Particles in p-Pb Collisions at $\sqrt{s_{NN}} = 5.02$ TeV, arXiv:1210.4520v1 [nucl-ex] 16 Oct 2012