



**HAL**  
open science

# Etude d'un système de filtrage en ligne pour le détecteur ATLAS

Eric Fede

► **To cite this version:**

Eric Fede. Etude d'un système de filtrage en ligne pour le détecteur ATLAS. Physique des Hautes Energies - Expérience [hep-ex]. Université de la Méditerranée - Aix-Marseille II, 2001. Français. NNT: . tel-00001424

**HAL Id: tel-00001424**

**<https://theses.hal.science/tel-00001424>**

Submitted on 14 Jun 2002

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



CPPM-T-2001-01  
N° d'ordre :

UNIVERSITÉ DE LA MEDITERANÉE  
AIX-MARSEILLE II  
FACULTÉ DES SCIENCES DE LUMINY  
163 Avenue de Luminy  
13288 MARSEILLE Cedex09

## THÈSE DE DOCTORAT

Spécialité : Physique des Particules, Physique Mathématique et Modélisation

présentée par

**Éric FEDE**

en vue d'obtenir le grade de docteur de l'Université de la Méditerranée

### ÉTUDE D'UN SYSTÈME DE FILTRAGE EN LIGNE POUR LE DÉTECTEUR ATLAS

Soutenue le 26 février 2001

Mr Manuel DELFINO-REZNICEK Rapporteur  
Mr Philippe DEVINS  
Mr Christian MICHAU  
Mr Sylvain TISSERANT  
Mr François TOUCHARD

Président de jury  
Directeur de thèse



## Remerciements

Avant tout je tiens à préciser que ces dernières années durant lesquelles j'ai effectué ma thèse au Centre de Physique des Particules de Marseille, m'ont énormément apporté tant du point de vue humain, que professionnel. C'est pourquoi je voudrais commencer par remercier l'ensemble du personnel du Centre de Physique des Particules de Marseille pour leur disponibilité et leur confiance.

Je remercie particulièrement Mr Elie Aslanides, directeur du laboratoire, de m'avoir accueilli au sein de son laboratoire et de m'avoir fourni les moyens de mener à bien cette thèse.

J'adresse mes remerciements aux membres du jury, Mr Christian Michau, Mr Sylvain Tisserant et Mr Philippe Devins pour l'attention qu'ils ont bien voulu apporter à ma thèse, ainsi qu'à Mr Manuel Delfino-Reznick et Mr Yves Robert qui ont bien voulu en être les rapporteurs.

Je voudrai également remercier l'ensemble des personnes composant le groupe "temps réel", à commencer par Mr Francois Etienne pour la confiance qu'il m'a accordé, mais aussi Christophe Meessen, Pierre Yves Duval et Auguste Le Van Suu pour leurs précieux conseils. Je n'oublie pas Mme Zuxian Qian qui a accepté de partager son bureau avec moi durant cette période et qui m'a, à de nombreuses reprises, judicieusement conseillé. Merci Chris Bee pour tout et notamment pour le merveilleux séjour en Russie que nous avons passé (merci Natacha).

Enfin je veux remercier profondément Mr Francois Touchard pour sa disponibilité et la patience qu'il a eu à mon égard. J'ai souvent été pour lui la cause d'un surplus de travail, surtout lors de la rédaction de ce mémoire.

Je remercie également l'ensemble des collaborateurs impliqués dans le système d'acquisition et déclenchement d'ATLAS avec qui j'ai eu la chance de travailler. Enfin je remercie Mr Devins, Mr Dufour et de façon plus générale ESC Informatique sans qui cette thèse n'aurait pu être réalisée.

Je remercie également ma famille pour son soutien et puis pour tout le reste et enfin merci à Anne-Sophie, Stéphane, Yann, Laurent, Fanny, Laurent, JYF, etc .....

Et puis merci aussi à tout les bibis, mouroins et autres esches qui ont été sacrifiés sur l'autel de la pêche à la ligne.

---



*Ça pite un peu le roustagaou?*





## Table des matieres

|  |    |
|--|----|
| CHAPITRE 1.....                                  | 13 |
| LES PARTICULES AUJOURD’HUI.....                  | 13 |
| 1.1 LE MODELE STANDARD.....                      | 14 |
| 1.1.1 LES DIFFERENTES FAMILLES.....              | 15 |
| 1.2 LES LIMITES DU MODELE STANDARD.....          | 19 |
| 1.3 AU DELA DU MODELE STANDARD.....              | 20 |
| 1.3.1 LES MODELES SUPERSYMETRIQUES (SUSY).....   | 20 |
| 1.3.2 LE MODELE MSSM.....                        | 21 |
| 1.3.3 LE MODELE SUGRA.....                       | 22 |
| 1.4 LES RECHERCHES ACTUELLES.....                | 22 |
| 1.4.1 LE BOSON DE HIGGS.....                     | 22 |
| 1.4.2 LES SUPERPARTICULES.....                   | 25 |
| 1.4.3 LES ETUDES MENEES.....                     | 26 |
| 1.5 LES SIGNATURES PHYSIQUES.....                | 27 |
| 1.5.1 LES DIFFERENTES SIGNATURES.....            | 27 |
| 1.5.2 LES PERFORMANCES A ATTEINDRE .....         | 29 |
| 1.6 RESUME.....                                  | 31 |
| CHAPITRE 2.....                                  | 33 |
| LE LARGE HADRONS COLLIDER ET SES DETECTEURS..... | 33 |
| 2.1 LES BESOINS.....                             | 34 |
| 2.1.1 LA LUMINOSITE.....                         | 34 |
| 2.1.2 L’ENERGIE.....                             | 34 |
| 2.2 LES ACCELERATEURS.....                       | 35 |
| 2.2.1 LES COLLISIONNEURS.....                    | 36 |
| 2.2.2 LE TEVATRON.....                           | 37 |
| 2.2.3 LE LEP.....                                | 38 |
| 2.3 LE LHC: LARGE HADRON COLLIDER.....           | 38 |
| 2.3.1 GENERALITES.....                           | 39 |
| 2.3.1.1 SITUATION GEOGRAPHIQUE.....              | 39 |
| 2.3.1.2 LA COLLABORATION.....                    | 40 |
| 2.3.2 LES CARACTERISTIQUES.....                  | 40 |



|   |    |
|---|----|
| 2.3.2.1 L'ENERGIE.....                                      | 40 |
| 2.3.2.2 LA LUMINOSITE.....                                  | 41 |
| 2.3.2.3 LES FAISCEAUX.....                                  | 41 |
| 2.3.2.4 LES AIMANTS DE COURBURE.....                        | 42 |
| 2.4 LES EXPERIENCES AUTOUR DU LHC.....                      | 44 |
| 2.4.1 ALICE : A LARGE ION COLLIDER EXPERIMENT.....          | 44 |
| 2.4.1.1 : LES OBJECTIFS D'ALICE.....                        | 44 |
| 2.4.2 LHCb : A LARGE HADRON COLLIDER BEAUTY EXPERIMENT..... | 46 |
| 2.4.3 CMS : COMPACT MUON SOLENOID.....                      | 47 |
| 2.5 ATLAS : A TOROÏDAL LHC APPARATUS.....                   | 48 |
| 2.5.1 CARACTERISTIQUES GENERALES.....                       | 49 |
| 2.5.2 DETECTEUR INTERNE OU DE TRACE.....                    | 50 |
| 2.5.3 LES CALORIMETRES.....                                 | 53 |
| 2.5.4 LES CHAMBRES A MUONS.....                             | 56 |
| 2.6 RESUME.....   | 58 |
| CHAPITRE 3.....   | 59 |
| ACQUISITION ET DECLENCHEMENT.....                           | 59 |
| 3.1 DECLENCHEMENT .....                                     | 61 |
| 3.1.1 LES OBJECTIFS DES TRIGGERS.....                       | 61 |
| 3.1.2 LES CONTRAINTES SUR LES TRIGGERS.....                 | 62 |
| 3.1.3 VUE GLOBALE DES TRIGGERS.....                         | 62 |
| 3.1.4 LVL1.....   | 63 |
| 3.1.5 LVL2.....   | 65 |
| 3.2 ACQUISITION DES DONNEES .....                           | 66 |
| 3.2.1 OBJECTIFS.....  | 66 |
| 3.2.2 CONTRAINTES.....                                      | 67 |
| 3.2.3 ACQUISITION DE BAS NIVEAU.....                        | 68 |
| 3.2.4 ACQUISITION DE HAUT NIVEAU.....                       | 70 |
| 3.2.5 ACQUISITION GLOBALE.....                              | 72 |
| 3.3 LES LOGICIELS DE SUPPORT (BACK END SOFTWARE).....       | 74 |
| 3.4 LE DCS : DETECTOR CONTROL SYSTEM.....                   | 76 |
| 3.5 RESUME.....   | 79 |
| CHAPITRE 4.....   | 81 |
| LE FILTRE D'EVENTEMENTS.....                                | 81 |

|  |     |
|--|-----|
| 4.1 GENERALITES.....                                   | 81  |
| 4.1.1 FONCTIONALITES ATTENDUES.....                    | 82  |
| 4.1.1.1 SELECTIONNER LES EVENEMENTS.....               | 82  |
| 4.1.1.2 MONITORAGE DE L'EXPERIENCE.....                | 83  |
| 4.1.1.3 FONCTIONS DE CALIBRATION ET D'ALIGNEMENT.....  | 83  |
| 4.2 FACTORISATION FONCTIONNELLE.....                   | 84  |
| 4.2.1 FONCTIONS SYSTEMES.....                          | 84  |
| 4.2.1.1 GESTION DU FLOT DE DONNEES (DATAFLOW).....     | 84  |
| 4.2.1.2 SUPERVISION DES TACHES.....                    | 85  |
| 4.2.2 FONCTIONS LIEES AUX TRAITEMENTS DES DONNEES..... | 86  |
| 4.2.2.1 TACHES D'ANALYSES.....                         | 86  |
| 4.2.2.2 MONITORAGE DES APPLICATIONS.....               | 87  |
| 4.2.2.3 PUISSANCE DE CALCUL.....                       | 87  |
| 4.3 CONTRAINTES TECHNIQUES.....                        | 88  |
| 4.3.1 CONTRAINTE GENERALES.....                        | 88  |
| 4.3.1.1 LA ROBUSTESSE.....                             | 88  |
| 4.3.1.2 LA FLEXIBILITE.....                            | 89  |
| 4.3.1.3 LA SECURITE DES DONNEES.....                   | 89  |
| 4.3.1.4 L'EVOLUTIBILITE.....                           | 90  |
| 4.3.2 CONTRAINTES LIEES AU TRAITEMENT DES DONNEES..... | 90  |
| 4.3.2.1 L'ENVIRONNEMENT DE RECONSTRUCTION.....         | 91  |
| 4.3.2.2 LA RECONSTRUCTION.....                         | 92  |
| 4.3.2.3 LA SELECTION.....                              | 95  |
| 4.4 LES UNITES DE CALCUL .....                         | 95  |
| 4.4.1 LES MONO-PROCESSEURS.....                        | 96  |
| 4.4.2 LES SYTEMES MULTI-PROCESSEURS.....               | 97  |
| 4.5 CAHIER DES CHARGES ET PROTOTYPAGE.....             | 97  |
| 4.5.1 CAHIER DES CHARGES.....                          | 97  |
| 4.5.2 PROTOTYPES.....                                  | 98  |
| 4.5.2.1 PROTOTYPE PC.....                              | 99  |
| 4.5.2.2 PROTOTYPE MULTIPROCESSEUR.....                 | 100 |
| 4.6 RESUME.....  | 101 |
| CHAPITRE 5.....  | 103 |
| LE PROTOTYPE PC.....                                   | 103 |
| 5.1 FINALITE DU PROTOTYPE.....                         | 103 |
| 5.2 LE FLOT DES DONNEES.....                           | 104 |

|         |  |     |
|---------|--|-----|
| 5.2.1   | LE DESIGN.....                             | 104 |
| 5.2.1.1 | LES MODELES DE DISTRIBUTION.....           | 104 |
| 5.2.2   | LES COMPOSANTS.....                        | 107 |
| 5.2.3   | ADEQUATION DES SOLUTIONS APPORTEES.....    | 112 |
| 5.2.4   | LES COMMUNICATIONS.....                    | 113 |
| 5.2.4.1 | LE MATERIEL ET COUCHE DE BAS NIVEAU.....   | 113 |
| 5.2.4.2 | LES PROTOCOLES DE HAUT NIVEAU.....         | 115 |
| 5.2.5   | LES TECHNOLOGIES RETENUES.....             | 115 |
| 5.2.6   | L'IMPLEMENTATION .....                     | 117 |
| 5.3     | SUPERVISION : CONTROLE ET MONITORAGE.....  | 118 |
| 5.3.1   | CONTROLE DU FLOT DES DONNÉES .....         | 118 |
| 5.3.2   | MONITORAGE DU FLOT DES DONNÉES .....       | 119 |
| 5.3.3   | CONSTRAINTES SUR LE SUPERVISEUR .....      | 119 |
| 5.3.4   | SOLUTIONS POSSIBLES.....                   | 120 |
| 5.3.5   | TECHNOLOGIES RETENUES.....                 | 121 |
| 5.3.5.1 | BASE DE DONNÉES DE CONFIGURATION.....      | 121 |
| 5.3.5.2 | LES AGENTS MOBILES.....                    | 122 |
| 5.3.5.3 | FONCTIONNEMENT.....                        | 123 |
| 5.3.5.4 | ADEQUATION DES AGENTS MOBILES.....         | 126 |
| 5.3.6   | IMPLEMENTATION.....                        | 126 |
| 5.3.6.1 | INTERFACES UTILISATEURS.....               | 126 |
| 5.3.6.2 | ADEQUATION DES SOLUTIONS APPORTEES.....    | 130 |
| 5.4     | TESTS ET MESURES.....                      | 130 |
| 5.4.1   | CONDITIONS DES TESTS.....                  | 130 |
| 5.4.2   | TESTS SUR LE FLOT DES DONNÉES.....         | 131 |
| 5.4.2.1 | TESTS DES FONCTIONS.....                   | 131 |
| 5.4.2.2 | MESURE DE PERFORMANCES.....                | 132 |
| 5.4.2.3 | CONCLUSIONS DES TESTS DE DATAFLOW.....     | 136 |
| 5.4.3   | TESTS SUR LA SUPERVISION.....              | 137 |
| 5.4.3.1 | CONCLUSIONS DES TESTS DE SUPERVISION.....  | 138 |
| 5.4.4   | INTEGRATION.....                           | 138 |
| 5.4.5   | TEST D'UN PROGRAMME DE RECONSTRUCTION..... | 138 |
| 5.4.6   | RECOMMANDATIONS.....                       | 142 |
| 5.5     | RESUME.....                                | 143 |
|         | CHAPITRE 6.....                            | 145 |
|         | PERSPECTIVES ET CONCLUSIONS.....           | 145 |
| 6.1     | CONCLUSIONS.....                           |     |

|  |     |
|--|-----|
| .....  | 145 |
| 6.2 PERSPECTIVES.....                              | 147 |
| 6.2.1 LES GRILLES.....                             | 148 |
| 6.2.2 GRILLES ET PHYSIQUE DES HAUTES ENERGIES..... | 149 |
| 6.2.3 GRILLES ET FILTRE D'EVENEMENT.....           | 150 |
| ANNEXE 1: SPECINT95.....                           | 153 |
| ANNEXE 2 : TRIGGERS PHYSIQUES.....                 | 155 |
| GLOSSAIRE.....                                     | 157 |





# CHAPITRE 1

## LES PARTICULES AUJOURD'HUI

La physique des particules, telle qu'on la définit actuellement, est une branche de la physique relativement récente mais qui a très vite évolué au cours de son existence. Son lien de parenté avec la physique nucléaire, domaine où des moyens très importants ont été engagés, n'est pas étranger à cette rapide évolution.

Les expériences « sur la table » qui prévalaient il y a 70 ans et qui ont posé les bases de cette discipline, ne sont plus aujourd'hui de rigueur. Cependant ces expériences et celles qui ont suivies ont permis de définir des modèles théoriques, qui se sont affinés au cours du temps et des expériences pour aboutir de nos jours à ce qu'on appelle le Modèle Standard. Ce modèle est l'aboutissement des recherches qui ont permis de repousser la limite de ce que croyait être l'état fondamental de la matière à savoir l'atome au début de ce siècle, jusqu'à ce qu'on croit l'être aujourd'hui : les quarks et les leptons.

Le changement d'échelle n'a pas été seulement réalisé au niveau des expériences elles-mêmes mais aussi au niveau des moyens humains mis en jeu. Les premières expériences du début du siècle ayant trait aux particules étaient l'oeuvre de quelques physiciens travaillant pour la plupart en solitaires. Aujourd'hui les expériences qui tournent autour de cette physique sont réalisées par des collaborations internationales regroupant plusieurs centaines de physiciens et d'ingénieurs.

## 1.1 LE MODELE STANDARD

La physique des particules, comme tout autre domaine de la physique, évolue par interactions entre théories et résultats expérimentaux, l'aspect théorique prédisant des comportements que l'on va chercher à mettre en évidence par l'expérience, et l'expérience fournissant les contraintes dans lesquelles devront être construits les modèles théoriques.

A l'heure actuelle, il existe un modèle théorique dit « Modèle Standard » qui explique de façon satisfaisante la quasi-totalité des phénomènes observés. Ce Modèle décrit trois des quatre interactions fondamentales qui régissent les phénomènes physiques, à savoir l'interaction électromagnétique, l'interaction faible et l'interaction forte. Seule la gravitation n'est pas intégrée dans ce Modèle.

Les caractéristiques des particules découvertes jusqu'à présent sont parfaitement expliquées par ce modèle, et même une partie d'entre elles ont été prédites à la suite de développement de ce modèle théorique, prédictions confirmées par la suite par l'expérience.

Le modèle standard n'est pas né de l'esprit d'une personne ou d'un groupe de travail, mais il est plutôt une construction qui a mis plusieurs dizaines d'années à s'édifier et à laquelle plusieurs centaines de physiciens ont participé.

Il est le résultat de la concaténation de la théorie dite QED (quantum electrodynamics), de la théorie électrofaible, de la théorie de Higgs et finalement de QCD (quantum chromodynamics). C'est l'amalgame de toutes ces idées qui a donné corps au Modèle Standard.

Le Modèle Standard est basé sur des symétries de jauge. L'ensemble des groupes nécessaire et suffisant pour laisser invariant le Lagrangien du modèle se trouve être :

$$SU(3)_C \otimes SU(2)_L \otimes U(1)_Y$$

où C, L et Y sont les nombres quantiques de couleur, d'isospin faible et d'hypercharge.

### 1.1.1 LES DIFFERENTES FAMILLES

Le modèle différencie les particules qui constituent la matière de celles qui transmettent les interactions. De plus la zoologie de toutes ces particules doit être enrichie d'une autre gamme de particules que sont les antiparticules. En effet à chaque particule est associée une antiparticule (toute aussi réelle que la particule), ce qui multiplie par quasiment deux le nombre de particules. Une particule et une anti-particules ont la même masse mais possèdent des nombres quantique opposés. En fait certaines particules sont leurs propre anti-particule, ce qui explique que ce facteur ne soit pas vraiment égal à deux.

#### *x Les particules qui forment la matière*

La matière, telle que nous la connaissons, est constituée d'un petit nombre de particules, que l'on sépare en deux types : les leptons et les quarks.

Les leptons au nombre de 6 (plus 6 anti-leptons) forment la famille dont l'électron fait partie. Leurs caractéristiques principales sont :

- ✓ Une partie d'entre eux sont chargés : l'électron ( $e$ ), le muon ( $\mu$ ) et le tau ( $\tau$ ) avec une charge commune  $-e$ .
- ✓ L'autre partie est neutre, neutrino électronique ( $\nu_e$ ), neutrino mu ( $\nu_\mu$ ) et neutrino tau ( $\nu_\tau$ ).
- ✓ Ils ne sont pas sensibles à l'interaction forte.

Les quarks, au nombre de 6 également (plus 6 anti-quarks) sont les constituants élémentaires d'une famille de particules appelées hadrons dont le proton et le neutron sont les plus connus. Leurs caractéristiques principales sont :

- ✓ D'être de spin  $\frac{1}{2}$ .



- ✓ De porter des charges électriques fractionnaires,  $2/3$  et  $-1/3$  de la charge de l'électron  $e$ .
- ✓ De porter une charge de couleur.
- ✓ D'être sensible à l'interaction forte.

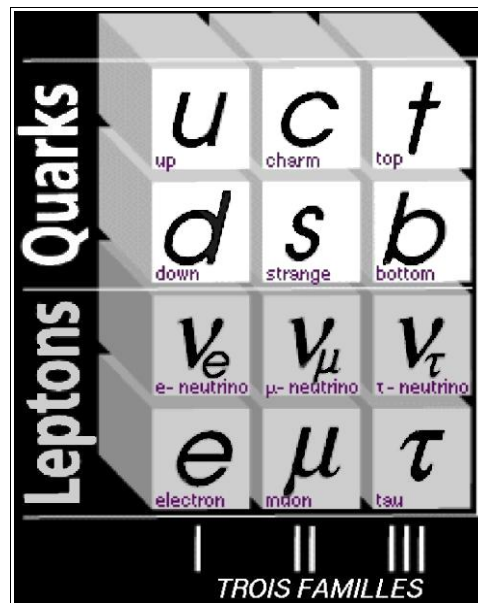


Figure 1 : Les différentes particules constituant la matière

Chacun de ces deux types de particules de matière peut être découpé en trois familles. Il est à noter que la matière telle que nous la définissons couramment, c'est à dire celle qui nous entoure, est constituée exclusivement des particules qui composent la première famille des leptons et des quarks, à savoir les quarks up et down et les leptons électron et neutrino électronique.

*x Les particules qui transmettent les interactions.*

Les particules que nous venons de voir et qui forment la matière ne sont pas isolées entre elles, elles ont des interactions. Ces interactions, au nombre de quatre, sont véhiculées par des messagers qui forment une nouvelle famille de particules, les bosons.

Ces interactions sont responsables de l'ensemble des phénomènes physiques qui existent dans l'univers.

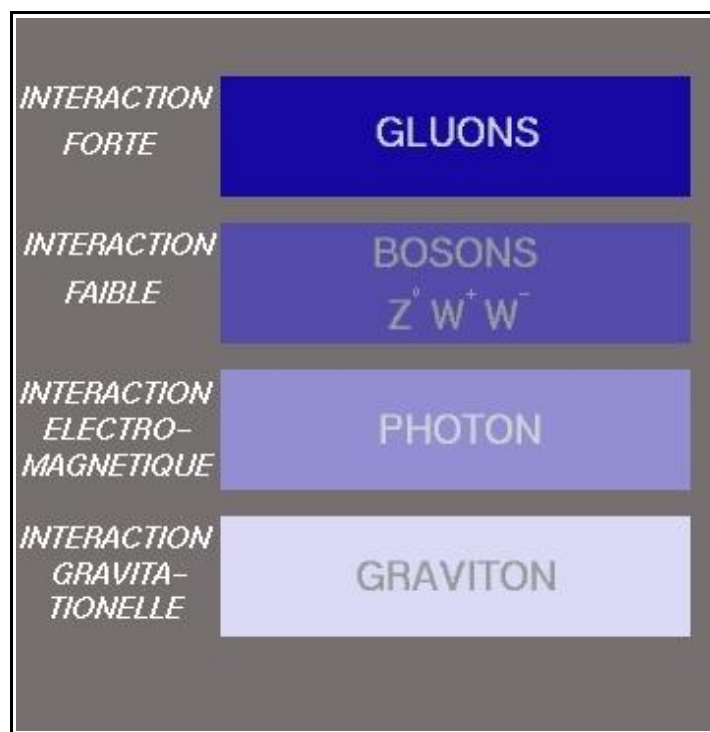


Figure 2 : Les particules d'interactions

- *l'interaction électromagnétique.*

C'est l'interaction qui intervient dans tous les phénomènes électriques et magnétiques, c'est-à-dire dans la quasi-totalité des phénomènes physiques qui font notre vie quotidienne, à l'exception des phénomènes liés à la gravitation.

Le boson qui véhicule cette force est le photon, celui-la même qui constitue la lumière. Le photon a la particularité d'être sans masse, ce qui caractérise une autre propriété de cette interaction qui est d'avoir une portée infinie.

- *l'interaction faible.*

C'est l'interaction qui est mise en jeu dans les phénomènes de désintégration de type Bêta. Elle est véhiculée par trois particules, le  $Z^0$  et les  $W^+$ , dont la découverte au CERN a été de première importance car elles étaient prédites par le modèle et leur découverte a correspondu aux prévisions.

- *l'interaction forte.*

C'est la force qui permet la cohésion des quarks. En effet les quarks ne peuvent pas exister à l'état libre c'est-à-dire seuls. Ils sont nécessairement liés avec d'autres quarks et cette liaison est réalisée par l'interaction forte qui est véhiculée par huit bosons appelés gluons. Cette interaction est notamment à la base des réactions de fusion nucléaire dont, la manifestation la plus commune est la production d'énergie au coeur de notre étoile, le soleil. La portée de cette interaction est extrêmement faible, c'est la plus courte des quatre .

- *l'interaction gravitationnelle.*

C'est la seule interaction qui n'est pas prise en compte par le modèle standard et pourtant c'est la première force qui, historiquement, fut découverte. Elle est véhiculée par une particule appelée graviton dont l'existence est purement théorique. A l'instar de l'interaction électromagnétique la portée de la gravitation est également infinie.

- x *Le problème des masses.*

De cet édifice constitué des particules de « matière » et des particules de « forces »

ressort un défaut, celui de la masse de ces particules. En effet, que ce soit pour les fermions ou pour les bosons, aucune masse ne leur est associée par le modèle. C'est pourquoi le physicien Peter Higgs a introduit une nouvelle brisure dans le modèle afin de générer les masses; c'est ce que l'on appelle le mécanisme de Higgs. Cependant le mécanisme induit également la présence d'une nouvelle particule, le boson de Higgs, qui à l'heure actuelle n'a pas encore été observée.

Il est important de noter que l'interaction électromagnétique et l'interaction faible ont été unifiées dans les années 1960–1970 pour ne faire qu'une interaction appelée interaction électrofaible. Des efforts prometteurs mais pas encore aboutis sont déployés pour essayer d'unifier cette interaction électrofaible avec l'interaction forte, ce qui ne laisserait plus que deux interactions.

Bien évidemment, l'idée de pouvoir réduire l'ensemble des forces à une seule interaction a été évoquée par la communauté scientifique, mais les divergences qui existent entre le modèle gravitationnel et le modèle standard rendent cette idée utopique dans le cadre de nos connaissances actuelles. Pourtant il est difficile de remettre en cause l'un des modèles, tellement ils sont capables, chacun dans son domaine, d'expliquer les observations. Ceci tend à faire penser qu'ils ne sont tous les deux qu'une approximation d'un modèle plus général.

## 1.2 LES LIMITES DU MODELE STANDARD

Bien que complet dans son domaine et ayant une grande confiance de la part des physiciens, le modèle standard présente des incohérences et des défauts.

La première des critiques que l'on peut faire à ce modèle concerne le fait que nous avons déjà évoqué, à savoir la non prise en compte de l'interaction gravitationnelle. Il est en effet très difficile de se satisfaire d'un modèle qui exclut une composante telle que la gravitation. Le modèle de la gravitation est actuellement décrit par le modèle de la relativité générale, qui lui aussi a fait ses preuves.

La seconde critique importante concernant le modèle standard est liée au nombre élevé de paramètres libres du modèle. Le lot de paramètres non prédictibles est en effet assez important. C'est notamment le cas pour ce qui est des masses des particules, le mécanisme de Higgs donnant une masse aux particules mais n'en prédisant pas la valeur.

Un autre reproche concerne le fait que le nombre de familles leptoniques est égal à trois. Ce nombre de trois familles n'est pas justifié théoriquement .

Sans remettre violemment en cause le modèle standard, on peut dire que de nombreux indices poussent les théoriciens à ne voir dans ce modèle qu'une approximation d'un modèle plus général.

## **1.3 AU DELA DU MODELE STANDARD**

De nouveaux modèles théoriques ont été élaborés afin d'essayer de construire un modèle plus complet, notamment en prenant en compte l'interaction gravitationnelle.

Ainsi de nombreux modèles sont apparus et parmi ceux-ci une classe de modèles dits supersymétriques. Cette classe de modèles est à l'heure actuelle celle sur laquelle sont fondés la plupart des espoirs. Ceci est d'autant plus vrai qu'ils ouvrent la voie vers ce que l'on pourrait appeler une nouvelle physique des particules.

### **1.3.1 LES MODELES SUPERSYMETRIQUES (SUSY)**

Les modèles supersymétriques (SUSY) sont basés sur l'introduction d'une nouvelle symétrie entre bosons et fermions. La première conséquence de la présence de cette symétrie est l'introduction pour chacune des particules bosoniques et leptoniques de nouveaux partenaires dit supersymétriques. Aux électrons, neutrinos, quarks et autres sont associés de nouveaux compagnons supersymétriques de même masse.

Aucun signe de l'existence de superparticules n'a pour l'instant été constaté, ce qui a mené les physiciens à supposer que la symétrie que l'on a introduite doit être brisée. Cette brisure a parmi ses conséquences, de permettre de décaler la gamme de masse de ces superparticules et ainsi d'expliquer leur non-observation.

L'existence de ces superparticules pourrait également permettre d'apporter une solution au problème de la nature de la matière noire de l'univers qui reste encore une énigme.

### 1.3.2 LE MODELE MSSM

L'introduction d'une supersymétrie peut bien évidemment être appliquée au modèle standard. On définit ainsi le Modèle Super-Symétrique Minimal (MSSM) qui n'est que l'extension SUSY du modèle standard pour lequel on a minimisé le nombre de nouvelles particules introduites.

Par convention, il a été décidé de nommer les partenaires supersymétriques des particules du modèle standard par le nom de s-particules. Ainsi le partenaire de l'électron est le s-électron. En dehors de cette cohorte de particules supersymétriques supplémentaires il existe une autre différence qui caractérise le MSSM et ceci concerne le boson de Higgs. En effet dans le modèle MSSM deux doublets de Higgs sont nécessaires pour générer les masses des fermions ce qui implique que la brisure de symétrie est constituée non pas de un boson de Higgs mais de cinq. On s'attend donc à avoir dans ce cas, deux bosons de Higgs scalaires chargés  $H^\pm$ , deux scalaires neutres  $H$  et  $h$  et enfin un pseudo-scalaire  $A$ .

Ce dernier point a son importance car la découverte de plus de un boson de Higgs sera la preuve indirecte de la validité des modèles supersymétriques et donc de l'existence des particules supersymétriques.

### 1.3.3 LE MODELE SUGRA

On vient de voir le modèle supersymétrique minimum qui comme le modèle standard présente encore l'inconvénient de ne pas tenir compte de l'interaction gravitationnelle. C'est pourquoi il existe un modèle dit de Super GRAvité.

Le modèle SUGRA introduit également une brisure de symétrie mais dans un modèle dit de grande unification (GUT). Le modèle de grande unification est une théorie qui présente les différentes interactions comme le résultat d'un découplage d'une unique interaction fondamentale. Dans ce modèle les trois interactions, forte, électromagnétique et faible ne seraient dans la gamme d'énergie de  $10^{16}$  GeV qu'une seule et même force.

## 1.4 LES RECHERCHES ACTUELLES

Dans les deux sous-chapitres précédents nous avons passé en revue, de façon sommaire, la théorie de base de la physique des particules. Nous avons vu le bestiaire des particules prédites par le modèle standard, ainsi que l'existence de particules vectrices d'interactions et même de façon extrêmement superficielle l'existence de nouveaux modèles. Cependant il ne faut pas pour autant imaginer ou croire que « tout est fait » et que la physique des particules est une branche de la physique gelée où tous les domaines de recherche ont été exploités .

De nombreuses questions restent posées, comme la validité des modèles supersymétriques, ou l'existence du boson de Higgs. Sont aussi menées des recherches pour affiner les mesures et la connaissance des particules déjà découvertes.

### 1.4.1 LE BOSON DE HIGGS

Comme exposé précédemment le boson de Higgs est la dernière particule prédite par le modèle standard non encore observée. Ceci représente une motivation importante pour une grande part de la communauté des physiciens des hautes énergies. Le LEP durant ses

dernières années de fonctionnement s'était fixé parmi ses buts la découverte de ce boson.

Les non-observations du Higgs ne sont pas pour autant des échecs car elles permettent de fixer des limites sur le modèle, et ainsi de définir des domaines possibles pour l'existence de ce boson. On définit ainsi des domaines d'exclusion de masses qui restreignent les domaines de recherche des futures expériences.

#### -Dans le modèle standard

On s'attend donc dans le cadre du modèle standard à l'observation d'un boson de Higgs. Aucune prédiction théorique n'est possible pour définir sa région de masse, seules des contraintes issues de l'observation peuvent être posées. Les études réalisées auprès d'autres accélérateurs excluent de façon certaine un Higgs de moins de 80 GeV. Les dernières études menées au LEP donnent même comme limite inférieure :  $m_H > 107$  GeV. Pour ce qui est de la limite de masse supérieure elle est de :  $m_H < 1$  TeV. Cette limite supérieure vient de la condition de non-violation de l'unitarité dans ce secteur de masse.

Le graphe (Figure 3) présente les taux de branchement des principaux canaux de désintégration du boson de Higgs dans le cadre du modèle standard en fonction de la masse du Higgs.



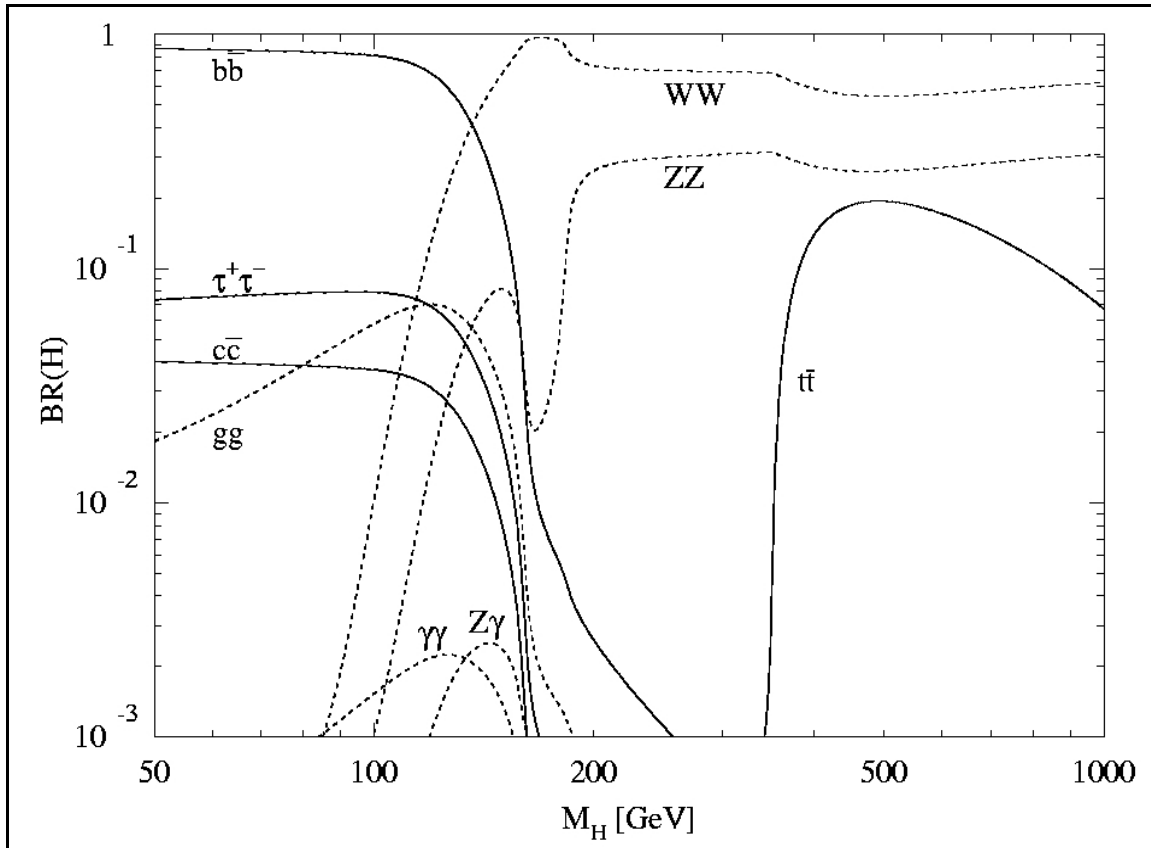


Figure 3 : Taux de branchement du Higgs standard

Le tableau (Table 1) montre les principaux canaux générateurs de bruit de fond en fonction des canaux susceptibles d'être recherchés. Les bruits de fond réductibles sont ceux qui sont susceptibles de pouvoir être différenciés du signal, car ils possèdent une caractéristique différente, alors que les bruits de fond irréductibles présentent une signature parfaitement identique au signal que l'on souhaite observer et sont donc impossibles à filtrer.

| <i>CANAL</i>                 | <i>Principaux bruits de fond irréductibles</i>   | <i>Principaux bruits de fond réductibles</i> |
|------------------------------|--|--|
| $H \rightarrow \gamma\gamma$ | $q\bar{q} \rightarrow \gamma\gamma, gg \rightarrow \gamma\gamma$<br>$qg \rightarrow q\gamma \rightarrow q\gamma\gamma$ | $Z \rightarrow e^+e^-$                       |

| <i>CANAL</i>                                  | <i>Principaux bruits de fond irréductibles</i>                           | <i>Principaux bruits de fond réductibles</i>                                     |
|---|--|--|
| $WH \rightarrow l\nu b\bar{b}$                | $WZ \rightarrow l\nu b\bar{b}$<br>$W b\bar{b} \rightarrow l\nu b\bar{b}$ | $Wjj, t\bar{t} \rightarrow l\nu\bar{\nu} b\bar{b}$                               |
| $t\bar{t} H \rightarrow b\bar{b} b\bar{b} l$  | $t\bar{t} Z \rightarrow t\bar{t} b\bar{b}$                               | $Wjjj, W b\bar{b}j$  |
| $H \rightarrow ZZ^* \rightarrow 4l$           | $Z\gamma \rightarrow ll\bar{l}\bar{l}$                                   | $t\bar{t} \rightarrow ll\bar{l}\bar{l}, Z b\bar{b} \rightarrow ll\bar{l}\bar{l}$ |
| $H \rightarrow ZZ \rightarrow ll\nu\bar{\nu}$ | $ZZ \rightarrow ll\nu\bar{\nu}$  | $Zj$   |
| $H \rightarrow WW \rightarrow l\nu jj$        | $Zj, Wj$   |  |

Table 1 : Principaux canaux de désintégration du Higgs

Les dernières observations du LEP avant son arrêt fin 2000 ont laissé apparaître une zone intéressante autour de 115 GeV [1][2].

#### -Dans le modèle MSSM

Dans ce cas, on s'attend donc à la présence de cinq bosons de Higgs. Un pseudo scalaire neutre A, deux scalaires chargés  $H^\pm$  et deux scalaires h et H. Ici encore on ne peut que poser des limites sur les masses de ces bosons. De plus, les limites dépendent d'autres paramètres libres ( $\tan\beta, m_A$ ) du modèle. C'est pourquoi dans ce cas on parle généralement d'espaces dans lesquels sont définies des limites.

### **1.4.2 LES SUPERPARTICULES**

Des expériences pouvant conduire à une observation indirecte de l'existence de la supersymétrie sont en cours de réalisation. Cependant la recherche directe de superparticules est également explorée par la communauté des physiciens des particules.

Dans le cadre des modèles avec conservation de la R-parité<sup>1</sup>, la particule supersymétrique la plus légère (LSP, Lightest Supersymmetric Particle) est stable, ce qui fait de cette LSP la particule la plus recherchée dans le cadre d'une recherche directe de superparticules. On considère que dans le cadre du MSSM, la LSP doit être un neutralino.

Dans le cadre de modèles avec violation de la R-parité on a deux conséquences phénoménologiques notables :

- ✓ La désintégration de la LSP en particules standards.
- ✓ La possibilité d'une production simple de superparticules.

Ceci conduit nécessairement à la recherche indirecte de superparticules, en étudiant notamment les diverses topologies qui peuvent résulter de la désintégration de la LSP.

### 1.4.3 LES ETUDES MENEES

Bien que le boson de Higgs et les superparticules soient encore absentes du catalogue établi par les physiciens, il ne faut pas pour autant croire que la physique des particules se borne à découvrir des nouvelles particules. Leurs études et la détermination précise de leurs caractéristiques se révèlent tout aussi intéressantes. Le quark top n'a été découvert au TEVATRON que très récemment et ses caractéristiques n'ont pas encore été mesurées de façon très précise. Ceci est vrai également pour d'autres particules pour lesquelles il est intéressant de déterminer, de la façon la plus précise, les différents paramètres comme notamment leur masse.

Enfin parmi les questions en vogue, il est à remarquer que bien que leur existence soit démontrée depuis de nombreuses années, on est encore incapable de donner une masse aux neutrinos. Cette question et sa conséquence directe, à savoir si les neutrinos oscillent, justifie à elle seule un nombre important de recherches qui sont soit en train d'être

---

<sup>1</sup> La R-parité correspond à la conservation de  $R = (-1)^{3B+L+2S}$  où B, L et S et sont les nombres Baryonique, Leptonique et de Spin

menées, soit d'être évaluées.

Toutes ces recherches, que l'on peut qualifier de « systématiques », servent notamment à poser des limites de plus en plus contraignantes sur les modèles, à fixer de façon plus précise les différents paramètres de ces modèles. Plusieurs disciplines scientifiques ont des modèles qui prennent en compte de telles systématiques. Il s'agit en particulier des différents modèles de production d'énergie au sein des systèmes stellaires qui sont très sensibles à certains des paramètres que des études systématiques en physique des particules peuvent déterminer .

## 1.5 LES SIGNATURES PHYSIQUES

Observer une particule comme le boson de Higgs, c'est faire ressortir du bruit de fond le signal de sa présence. Pour cela on va définir quels sont les critères que l'on doit appliquer aux événements<sup>2</sup> afin d'extraire le signal du bruit. Ces critères portent sur la caractérisation des particules qui composent l'événement; c'est le cas lorsqu'on regarde la nature d'une particule ou l'impulsion d'une autre. Mais ces critères peuvent également porter sur l'événement dans son ensemble; c'est notamment le cas si on s'intéresse à la présence d'un jet dans l'événement ou à l'énergie manquante totale de l'événement. C'est cet ensemble de critères de sélection, portant sur la physique contenue dans l'événement que l'on appelle signatures, ou triggers, physiques.

### 1.5.1 LES DIFFERENTES SIGNATURES

Dans le sous-chapitre précédent nous avons vu les différents canaux dans lesquels est susceptible de se désintégrer le boson de Higgs. Afin de définir les sélections qu'il sera nécessaire d'appliquer aux événements pour diminuer le bruit de fond de chacun des canaux il est nécessaire de bien comprendre quels seront les intervenants dans cette sélection [3].

---

<sup>2</sup> Un événement est le nom générique que l'on donne à l'ensemble des produits de la collision.

$$H \rightarrow \gamma\gamma$$

Dans ce canal de désintégration la difficulté réside dans la suppression des photons et électrons qui sont à l'origine du bruit de fond et non du signal. Cette suppression des électrons et photons passe nécessairement par leur identification et leur séparation, ce qui nécessite d'avoir un calorimètre électromagnétique très performant, possédant une résolution en énergie très bonne ainsi qu'une bonne résolution spatiale. Ces deux conditions sont nécessaires pour espérer sortir le pic de désintégration du Higgs du bruit de fond et obtenir une bonne résolution en masse.

$$H \rightarrow b\bar{b}$$

Ce type de désintégration du Higgs provient des productions associées de Higgs auxquelles on s'attend et qui sont WH, ZH,  $t\bar{t}H$ . Dans ce cas le fait de pouvoir identifier et séparer les jets  $b$  est primordial, de même que l'identification des leptons qui sont présents dans un certain nombre d'états finaux. Tous cela demande des performances à la fois au niveau des détecteurs de muons mais aussi au niveau des détecteurs internes et des calorimètres notamment le hadronique pour déterminer l'énergie des jets.

$$H \rightarrow ZZ^* \rightarrow 4l$$

Ce canal de désintégration donne dans l'état final quatre leptons. C'est donc l'identification de ces leptons qui sera primordiale dans ce cas, ce qui nécessite une identification fine des électrons et positrons au niveau du calorimètre électromagnétique mais aussi des muons dans les chambres à muons et bien sûr des détecteurs internes.

$$H \rightarrow ZZ \rightarrow ll\nu\bar{\nu}$$

Ce canal présente la particularité d'avoir deux neutrinos dans son état final. N'étant pas détectable, l'énergie emportée par les neutrinos va apparaître comme un manque d'énergie au niveau de l'événement. Ce manque ne sera perceptible que si les calorimè-

tres sont capables de mesurer finement l'énergie de l'événement. La détection des deux leptons requiert encore les chambres à muons, le détecteur interne et le calorimètre électromagnétique.

$$H \rightarrow WW \rightarrow l\nu jj$$

Ce mode de désintégration du Higgs est particulièrement envisagé pour des masses du boson de Higgs élevées ( $m_H < 1 \text{ TeV}$ ). Ici tous les détecteurs devront être mis en jeu pour la détection.

Cette énumération des canaux de désintégration du boson de Higgs et des caractéristiques du signal, fait ressortir clairement que l'on peut envisager d'effectuer les sélections d'événement selon trois grands axes qui sont :

- ✓ *Séparation électrons photons* [4]: la principale difficulté concerne l'identification des électrons, qui sont présents dans de nombreux états finaux du Higgs, et la nécessité de ne pas les confondre avec des photons. C'est pourquoi cette séparation des deux particules est très importante.
- ✓ *Jet/ETmiss* [6][5]: la présence de jets ou d'énergie manquante représente une caractéristique importante d'un événement. C'est donc tout naturellement que des sélections vont porter sur ces deux caractéristiques.
- ✓ *B-tagging* [7]: le fait de pouvoir caractériser un jet comme provenant d'un hadron beau représente également un critère de sélection extrêmement important étant donné le grand nombre de canaux de désintégration du Higgs qui présentent un tel candidat.

## 1.5.2 LES PERFORMANCES A ATTEINDRE

Le taux de production du Higgs au sein de l'accélérateur LHC est estimé à  $10^{-2}$  Hz pour une fréquence de collisions de  $40 \cdot 10^6$  Hz. Le tableau (Table 2) montre le résultat des simulations de la production de Higgs et des bruits de fond. Ici le nombre d'événement

ment signal et bruit sont données pour une fenêtre en masse de  $m_H \pm 1,4\sigma$  et pour une luminosité intégrée de  $100 \text{ fb}^{-1}$ .

| <b>Masse du Higgs<br/>(GeV)</b> | <b>80</b> | <b>100</b> | <b>120</b> | <b>140</b> |
|---------------------------------|-----------|------------|------------|------------|
| Signal de production directe    | 502       | 947        | 1190       | 915        |
| Signal<br>WH, ZH, $t\bar{t}$ H  | 85        | 98         | 93         | 58         |
| Bruit de fond $\gamma\gamma$    | 41700     | 41400      | 29000      | 20600      |
| Bruit de fond<br>jet-jet        | 5400      | 5950       | 4600       | 3550       |
| Bruit de fond<br>jet- $\gamma$  | 12500     | 9100       | 5800       | 4100       |

Table 2 : Nombre d'événement Higgs par canal pour une luminosité intégrée de  $100 \text{ fb}^{-1}$  dans le LHC[8]

On définit de la sorte les signatures correspondant à la présence de leptons, isolés ou non, de jets, d'énergie manquante, qui vont caractériser les différents canaux de recherche du Higgs. Cela est également vrai pour les autres recherches notamment sur la physique du B. Le tableau (Table 3) présente les signatures attendues pour les canaux de désintégration du Higgs du modèle standard. Les leptons (l) identifiés sont soit des électrons soit des muons et l'indicatif i signifie que l'isolation de la particule est requise. Les valeurs représentent les minimums en impulsion transverse ( $P_T$ ) qui sont requises. Ces valeurs sont pour la basse luminosité.

| <b>CANAL</b>                        | <b>SIGNATURE</b>          |
|-------------------------------------|---------------------------|
| H $\rightarrow \gamma\gamma$        | $\gamma 40i + \gamma 25i$ |
| H avec H $\rightarrow \gamma\gamma$ | $2\gamma 25i + 1 25$      |

| <i>CANAL</i>                                       | <i>SIGNATURE</i>                       |
|--|--|
| H+jets avec H $\rightarrow\gamma\gamma$            | $\gamma 60i + \gamma 40i$              |
| $t\bar{t}$ H, WH, ZH avec H $\rightarrow b\bar{b}$ | $1 20i + 2b15$                         |
| H $\rightarrow ZZ^* \rightarrow 4l$                | $2 120i + 2 17i$                       |
| H $\rightarrow ZZ \rightarrow ll \nu\bar{\nu}$     | $1 20i + 1 10i + E_T^{\text{miss}} 40$ |
| H $\rightarrow WW^* \rightarrow ll \nu\bar{\nu}$   | $2l 40i + E_T^{\text{miss}} 50$        |
| $q\bar{q}$ H avec H $\rightarrow WW$               | $i 50i + E_T^{\text{miss}} 50$         |

Table 3 : Signatures caractéristiques par canal de désintégration d'un boson de Higgs[8]

## 1.6 RESUME

Dans ce premier chapitre, j'ai décrit très sommairement l'état actuel des connaissances en physique des particules et les principaux axes de recherche. Une description des principaux canaux et signatures physiques attachés au boson de Higgs a également été faite.





## CHAPITRE 2

# LE LARGE HADRONS COLLIDER ET SES DETECTEURS

Une des composantes de toute science est l'expérimentation, complément indispensable de la théorie. La physique des particules n'échappe pas à ce schéma et doit, pour confirmer ou infirmer un modèle théorique, se confronter à l'observation. Qui dit observation dit moyens d'observer. C'est pourquoi il est nécessaire de disposer d'instruments qui permettent de mettre en évidence les phénomènes que l'on souhaite analyser.

L'expérimentation en physique des particules nécessite de nos jours deux types d'instruments. D'un côté les accélérateurs de particules qui vont mettre l'observateur dans les conditions physiques nécessaires pour que les phénomènes physiques se déroulent, et de l'autre les détecteurs qui vont révéler ces phénomènes.

Les performances auxquelles on va demander à ces deux catégories d'instruments de répondre dépendent étroitement de la physique que l'on veut observer, ou du moins des domaines que l'on souhaite explorer.



## 2.1 LES BESOINS

La recherche auprès des expériences de physique des particules est actuellement, pour la majorité des projets, orientée sur deux grands axes que sont la recherche du boson de Higgs et la validation, ou l'invalidation, des modèles supersymétriques.

Ces deux axes de recherches présentent une difficulté commune, à savoir la rareté des phénomènes recherchés. Leur mise en évidence nécessite donc de créer un très grand nombre de situations observationnelles afin d'espérer y trouver les quelques cas représentatifs de la physique recherchée.

### 2.1.1 LA LUMINOSITE

Cette contrainte de rareté nécessite d'accumuler un maximum de statistique, ce qui implique de la part des accélérateurs qui sont susceptibles de faire cette physique de produire un maximum de situations observables. Cela est possible par une luminosité élevée de l'accélérateur. La luminosité d'un accélérateur de particules correspond au taux de collisions de particules que l'on produit en son sein.

La contre-partie d'une luminosité élevée est bien sûr que le nombre d'observations sera plus abondant et donc que la quantité d'informations à analyser sera plus conséquente.

### 2.1.2 L'ENERGIE

L'autre paramètre important pour la physique étudiée est l'énergie qui sera disponible pour produire les phénomènes physiques. En effet les créations de particules nécessitent d'atteindre un certain seuil en énergie, à partir duquel elles peuvent se matérialiser.

Les accélérateurs de la nouvelle génération franchissent et franchiront la barre du TeV, ce qui représente une véritable avancée par rapport aux instruments précédents qui se situent autour de la centaine de GeV. Cela reste néanmoins encore très éloigné des va-

leurs que peuvent fournir des sources de rayonnements naturelles telles que le rayonnement cosmique. Ici aussi comme pour l'augmentation de la luminosité, l'augmentation de l'énergie apporte certes un gain de potentiel de production de particules, mais aussi augmente notablement la quantité des particules produites et donc aussi la difficulté d'analyse.

Pour résumer les besoins techniques que nous impose la physique que l'on souhaite faire, on peut dire que la rareté des phénomènes recherchés nous conduit à produire un maximum d'observations en un minimum de temps. Cela est possible par l'augmentation de la luminosité des accélérateurs. Mais aussi les seuils d'apparition de ces phénomènes recherchés nous obligent à voir à la hausse les énergies que peuvent produire ces accélérateurs. Dans un cas on augmente la fréquence de nos observations, et dans l'autre on augmente la taille de ces observations, ce qui correspond dans tous les cas à une augmentation importante du flot de données à analyser et bien sûr augmente parallèlement les contraintes sur les moyens de détections, nécessitant ainsi des détecteurs de plus en plus complexes et de plus en plus rapides en réponse.

## 2.2 LES ACCELERATEURS

Les accélérateurs de particules représentent la gamme d'instruments qui permettent aux physiciens de sonder la matière. Ils sont utilisés pour créer les conditions notamment énergétiques, dans lesquelles de nouveaux phénomènes sont susceptibles d'avoir lieu.

A l'heure actuelle la quasi-totalité des accélérateurs sont de type circulaire : les particules suivent une trajectoire circulaire qui les fait passer régulièrement dans des cavités accélératrices où elles gagnent de l'énergie. Ainsi tour après tour elles s'accélèrent. La contre-partie de ce système est la difficulté rencontrée pour maintenir la trajectoire circulaire au fur et à mesure du gain d'énergie du faisceau. Cette trajectoire circulaire est engendrée par un système de champs magnétiques très intenses qui s'appliquent à l'ensemble du faisceau. Le gain en énergie lui, est réalisé par un ensemble de cavités radio-fréquence (cavités RF) où règne un champ électrique qui accélère les particules.

Il existe une autre façon d'avoir accès à des particules très énergétiques. On ne peut

pas parler à vrai dire d'accélérateur, car il s'agit en fait d'utiliser ce que la nature met à notre disposition, à savoir les rayonnements cosmiques. Les rayonnements cosmiques sont pour la majorité d'entre eux beaucoup plus énergétiques que le plus puissant des accélérateurs de construction humaine. Cependant, en plus de ne pas connaître initialement la nature des particules mises en jeu, ces rayonnements présentent l'inconvénient de ne pas être contrôlés en temps et en espace. C'est à dire que contrairement à un accélérateur artificiel, on ne contrôle pas le faisceau de particules accélérées. De plus pour une partie d'entre eux, il est nécessaire d'aller observer dans l'espace ce qui complique considérablement les possibilités d'observations.

### 2.2.1 LES COLLISIONNEURS

Pour atteindre des énergies de l'ordre du TeV la solution la plus appropriée consiste à utiliser une gamme d'accélérateurs appelés collisionneurs.

Les collisionneurs présentent l'avantage de fournir dans le centre de masse une énergie équivalente à quasiment le double de celle du faisceau produit. Le principe est extrêmement simple, contrairement à la réalisation. Il consiste à projeter de façon frontale deux faisceaux de particules en un point où se trouve un détecteur et cela pour observer les produits de la collision. En réalité on ne projette pas des faisceaux continus de particules, mais plutôt des faisceaux constitués de paquets de particules.

Les accélérateurs à cible fixe se différencient des collisionneurs notamment par le fait que la totalité des produits de la collision sont projetés en avant du point de collision. Il est donc seulement nécessaire de construire un détecteur sur cette partie avant. Dans ce cas on doit alors prendre en compte l'effet d'entraînement qui contient l'ensemble des produits de la collision dans un faible angle au solide. Pour un collisionneur le problème est tout autre: les produits de la collision peuvent aussi bien être présents en avant qu'en arrière du point de collision, ce qui nécessite de construire des détecteurs qui englobent le point d'impact sur  $4\pi$  stéradians.

Une autre caractéristique vient du fait que, pour des accélérateurs de haute énergie (collisionneurs compris), il est indispensable d'injecter les particules constituant le faisceau avec une certaine énergie initiale. Il est nécessaire d'employer des accélérateurs de

moindre importance pour injecter les particules dans les accélérateurs de haute énergie. C'est pourquoi les accélérateurs d'ancienne génération sont généralement reconvertis en injecteurs pour les instruments de nouvelle génération. C'est notamment le cas au CERN où le PS (Synchrotron à Protons) et le SPS (Super Synchrotron à Proton) servent dorénavant d'injecteurs pour le LEP et pour le futur LHC.

Il existe une façon efficace de gagner encore de l'énergie. Cette méthode consiste non pas à projeter des leptons, comme c'était le cas au LEP, mais des hadrons, qui présentent l'avantage de moins produire de rayonnement synchrotron que les leptons, ce qui est un grand avantage pour atteindre des énergies élevées. Par contre les hadrons ont l'inconvénient (par rapport aux leptons) de ne pas être des particules élémentaires mais un agrégat de quarks. Cela a pour conséquence que l'énergie est portée par plusieurs particules élémentaires et non pas une seule comme c'est le cas avec un collisionneur leptonique. L'énergie totale par particule du faisceau s'en trouve ainsi divisée. Bien sûr les difficultés techniques sont notablement différentes entre un collisionneur leptonique et un hadronique, ne serait-ce que par le fait que l'énergie augmentant, il est nécessaire d'avoir des systèmes de courbure de faisceaux plus puissants.

Il existe actuellement un collisionneur hadronique près de Chicago, le TEVATRON, qui permet d'atteindre une énergie de 2 TeV par des collisions de type proton-antiproton.

### 2.2.2 LE TEVATRON

Le TEVATRON est un collisionneur hadronique qui se trouve au FERMILAB à côté de Chicago. Il est entré en service en 1987 et permet d'atteindre une énergie dans le centre de masse de l'ordre de 2 TeV. Deux expériences, CDF [10] et DØ [9], sont installées auprès de cet accélérateur qui doit rester en service pendant encore quelques années.

Le TEVATRON, suite à une première campagne de prise de données, a été arrêté afin d'augmenter ses performances, notamment sa luminosité qui doit passer de  $0,16 \cdot 10^{31} \text{ cm}^{-2} \text{ s}^{-1}$  à  $5 \cdot 10^{31} \text{ cm}^{-2} \text{ s}^{-1}$  et son énergie de 1,8 à 2 TeV.

Le principal avantage d'utiliser une particule et son antiparticule dans un collision-

neur est qu'il n'est pas nécessaire d'avoir un système magnétique de courbure très complexe. En effet les deux faisceaux qui circulent en sens opposés sont dans ce cas courbés par le même champ magnétique. Par contre le taux de production d'antiprotons et surtout leur conditionnement en paquet n'étant pas facile, il en résulte une limite de luminosité.

Le TEVATRON est à l'origine de l'observation du quark top, dernier des 6 quarks prédits par le modèle standard.

### 2.2.3 LE LEP

Le LEP (Large Electron Positron collider) est un collisionneur leptonique situé au CERN, sur la frontière franco-helvétique. Construit durant les années 1980 il est entré en service en 1989 et vient de s'arrêter il y a quelques mois pour laisser la place à un collisionneur de nouvelle génération. Il était équipé de quatre détecteurs : ALEPH, DELPHI, OPAL et L3.

Le succès du LEP notamment par ses études spectroscopiques sur le  $Z^0$ , a conduit les états membres du CERN à étudier la construction de son successeur, le LHC, qui sera contrairement à son « concurrent » américain un collisionneur proton-proton permettant d'atteindre une énergie de 14 TeV et cela avec une luminosité très élevée. Il entrera en fonction courant 2005 et sera le plus puissant instrument d'investigation de la matière au monde.

## 2.3 LE LHC: Large Hadron Collider

Alors que le LEP commençait à peine à accélérer ses premiers électrons, des discussions ont été engagées afin de définir les spécificités de son successeur. Il était évident que l'accélérateur qui devrait se substituer au LEP serait orienté pour la découverte si possible du boson de Higgs

Ainsi dès 1994 le Conseil du CERN décida de construire le Large Hadron Collider dont la mise en service est prévue courant 2005.

### 2.3.1 GENERALITES

Le LHC sera un collisionneur qui dans un premier temps accélérera deux faisceaux de protons mais qui pourra également faire collisionner des faisceaux d'ions notamment de plomb.

Quatre points de collisions sont prévus auprès desquels prendront place quatre détecteurs : deux détecteurs dit généraliste ATLAS et CMS, et deux détecteurs plus spécifiques que sont LHCb et ALICE.

#### 2.3.1.1 SITUATION GEOGRAPHIQUE

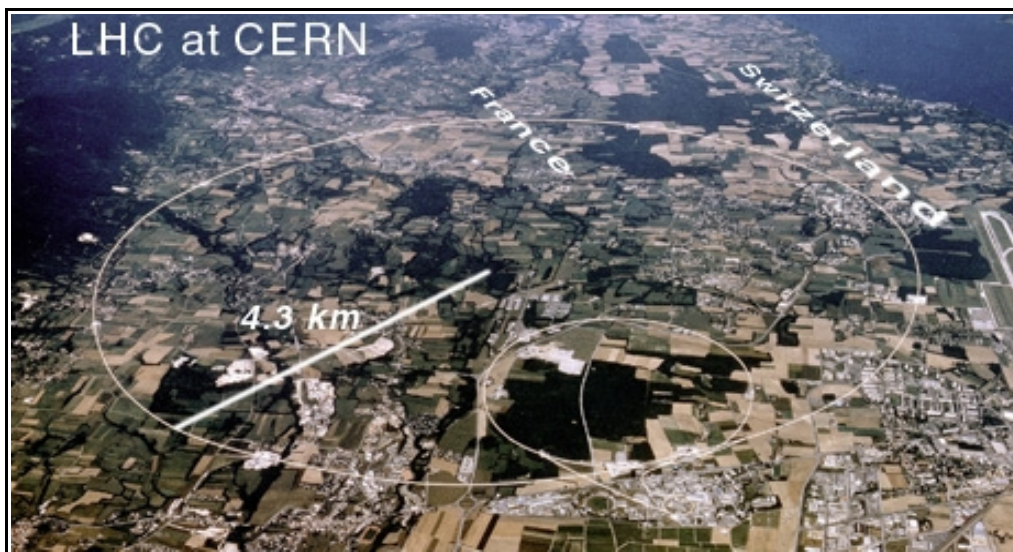


Figure 4 : Vue aérienne et représentation du tunnel du LHC

L'un des coûts les plus importants dans la construction d'un accélérateur vient de la partie de génie civil nécessaire à la mise en place de l'instrument. En effet les accélérateurs actuels sont placés dans des tunnels circulaires de plusieurs kilomètres de long qui se situent enfouis profondément sous terre. C'est notamment le cas du LEP qui se trouvait dans une galerie circulaire de 27 km de long situé sous la frontière franco-helvétique.



Afin de minimiser les coûts de construction du LHC, il fut décidé qu'il devrait prendre place dans la même galerie que celle du LEP. Ceci impose que les deux machines ne pourraient pas cohabiter, la naissance du LHC signifiant alors la mort du LEP.

### 2.3.1.2 LA COLLABORATION

La réalisation d'un tel instrument ne peut pas être supportée par une seule nation mais uniquement par un ensemble. Le LHC sera le fruit d'une collaboration qui regroupe la plupart des pays ayant des programmes de recherches en physique des particules. Les retombées des développements techniques sont nombreuses et enrichissantes pour une multitude de sociétés et cela dans de nombreux domaines.

### 2.3.2 LES CARACTERISTIQUES

Les caractéristiques du LHC ont nécessité d'importants efforts d'ingénierie et des développements spécifiques ont dû être réalisés, notamment en ce qui concerne les champs magnétiques nécessaires pour rendre la trajectoire des faisceaux circulaires. Regardons un peu en détail les principales caractéristiques techniques et physiques de ce collisionneur.

#### 2.3.2.1 L'ENERGIE

On a vu précédemment que l'énergie que fournit l'accélérateur dans le centre de masse doit être la plus importante possible pour révéler des phénomènes nouveaux. Le LHC sera capable de produire deux faisceaux de proton de 7 TeV chacun qui permettront d'atteindre dans le centre de masse une énergie de 14 TeV. Dans le mode collisionneur ionique le LHC pourra accélérer deux faisceaux d'ions qui fourniront dans le centre de masse une énergie de plusieurs centaines de TeV.

Le gain en énergie apporté par le LHC est énorme, près d'un facteur sept par rapport au TEVATRON. Le LHC sera le plus puissant accélérateur de particules au monde, et vraisemblablement le dernier des grands collisionneurs circulaires.

### 2.3.2.2 LA LUMINOSITE

L'autre point important dans la définition des caractéristiques d'un accélérateur est sa luminosité, c'est à dire le taux auquel il produit des collisions. Il est prévu deux périodes de fonctionnement du LHC, une première dite de "basse luminosité" ( $10^{33} \text{ cm}^{-2} \text{ s}^{-1}$ ) et une dite de "haute luminosité" ou luminosité nominale ( $10^{34} \text{ cm}^{-2} \text{ s}^{-1}$ ). Ceci correspond pour une année de fonctionnement à haute luminosité à une luminosité intégrée de 100 fb<sup>-1</sup>.

Cette luminosité a notamment été atteinte par une focalisation, donc une densification, des faisceaux mais aussi par une réduction de l'espacement temporel entre les paquets de protons. Typiquement le LHC produira une collision toutes les 25 nanosecondes ce qui correspond à une fréquence de 40 MHz.

La première année de mise en fonctionnement de l'accélérateur sera une année à basse luminosité. Elle correspondra à la période durant laquelle on va essentiellement comprendre le fonctionnement de l'accélérateur et surtout des détecteurs. C'est aussi une période durant laquelle des études particulières peuvent être envisagées, étant donné le niveau de bruit moindre généré par une luminosité plus faible. La basse luminosité présentant un avantage pour certaines études, notamment de physique du b, le LHC la produira de façon discontinue durant toute sa durée de fonctionnement.

La haute luminosité est la luminosité nominale pour laquelle l'accélérateur a été défini. C'est cette valeur que fournira l'accélérateur la plus grande partie de sa vie. Bien sûr il y a toujours des études menées pour éventuellement augmenter encore un peu cette valeur.

### 2.3.2.3 LES FAISCEAUX

Comme nous l'avons vu précédemment, la partie active d'un collisionneur est constituée des deux faisceaux. Dans le cas du LHC on aura affaire à un collisionneur qui contrairement au TEVATRON ou au LEP ne projettera pas deux faisceaux de nature différente (particule contre antiparticule) mais deux faisceaux constitués de la même particule : le proton.

Les protons seront dans un premier temps accélérés par un accélérateur linéaire

(LINAC) avant d'être injectés dans le PS où ils atteindront une énergie de 25 GeV après quoi ils seront injectés dans le SPS qui va pousser leur énergie jusqu'à 450 GeV. Enfin de là ils atteindront le LHC où aura lieu la dernière phase d'accélération, jusqu'à une énergie de 7 TeV.

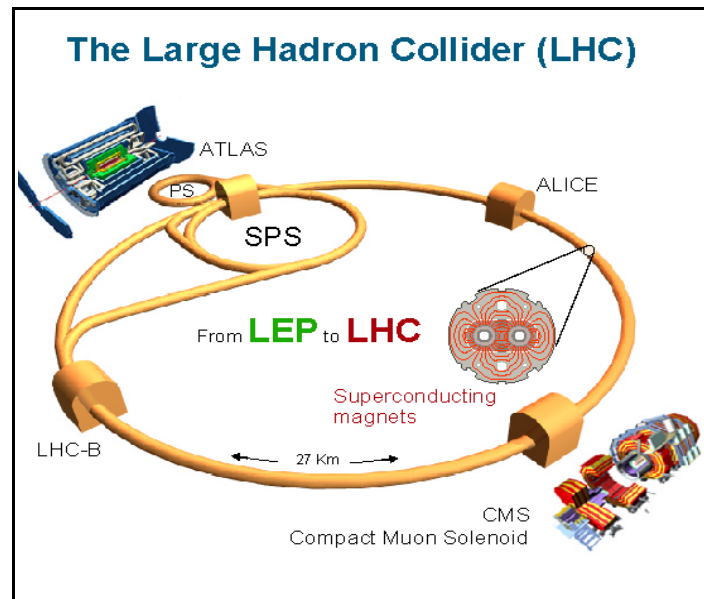


Figure 5 : Le LHC et ses expériences

Là, les deux faisceaux seront composés de 2808 paquets, chacun contenant  $10^{11}$  protons qui se croiseront toutes les 25 nanosecondes. A chaque tour, les paquets de protons de chaque faisceau traverseront 8 cavités accélératrices radio fréquence.

#### 2.3.2.4 LES AIMANTS DE COURBURE

Le fait d'installer le LHC dans le même tunnel que celui du LEP a nécessairement posé le problème de savoir comment tenir les faisceaux dans le rayon de courbure du tunnel. En effet, à rayon de courbure égal et à énergie plus élevée il est nécessaire d'appliquer un champ magnétique plus intense.

Pour maintenir la trajectoire d'un faisceau de 7 TeV sur un anneau de 27 km de long, il est nécessaire d'y appliquer un champ magnétique de l'ordre de 8,4 Tesla. Il faut éga-

lement garder à l'esprit qu'au moment de l'injection des protons dans le LHC, ils n'auront une énergie que de 450 GeV ce qui nécessitera seulement un champ magnétique de 0,5 Tesla.

La seule solution techniquement possible est l'utilisation d'aimants supraconducteurs. De plus afin de gagner de la place, mais aussi afin de permettre de n'avoir à installer qu'un seul système de cryostat pour les aimants, il a été décidé de placer les deux tubes à vide contenant les faisceaux dans une même structure mécanique. Ainsi on n'a besoin que d'un aimant supraconducteur dipolaire pour l'ensemble des deux faisceaux.

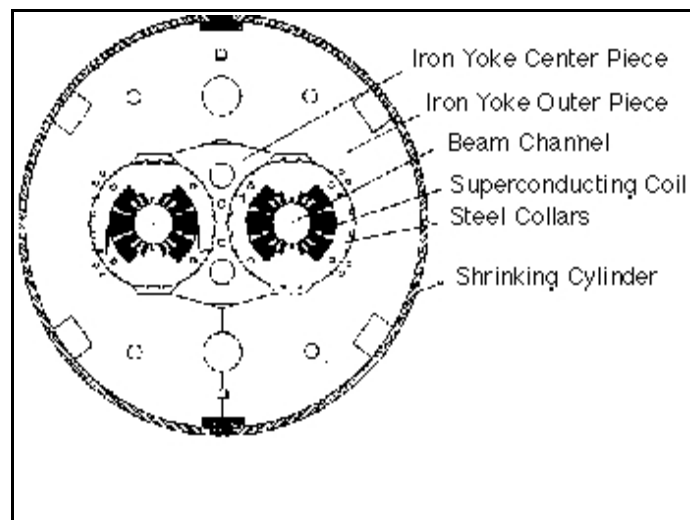


Figure 6 : Coupe transversale d'un dipôle du LHC

Le tableau (Table 4) résume les principales caractéristiques du Large Hadron Collider:

|                                 |  |
|---------------------------------|--|
| CIRCONFERENCE                   | 27 km                                  |
| TYPE DE FAISCEAU                | proton                                 |
| ENERGIE PAR FAISCEAU            | 7 TeV                                  |
| ENERGIE DANS LE CENTRE DE MASSE | 14 TeV                                 |
| LUMINOSITE NOMINALE             | $10^{34} \text{ cm}^{-2}\text{s}^{-1}$ |
| NOMBRE DE PAQUET PAR FAISCEAU   | 2808                                   |

|   |       |
|---|-------|
| CIRCONFERENCE                                     | 27 km |
| INTERVALLE TEMPOREL ENTRE CROISEMENTS             | 25 ns |
| DUREE DE VIE DU FAISCEAU                          | 22 h  |
| DUREE DE VIE DU FAISCEAU A LA LUMINOSITE NOMINALE | 10 h  |
| TEMPS DE RECHARGE DU FAISCEAU                     | ~ 2 h |

Table 4 : Caractéristiques principales du LHC

## 2.4 LES EXPERIENCES AUTOUR DU LHC

Les faisceaux du LHC se croiseront en quatre points distincts de l'anneau. En ces quatre points seront disposés quatre détecteurs de particules spécialement développés pour le LHC. Deux de ces détecteurs sont à vocation spécifique. Il s'agit de ALICE, dédié à l'étude des collisions ioniques et de LHCb dans les études seront orientées vers la physique du b. Les deux autres détecteurs, ATLAS et CMS, sont à vocation généraliste et leurs objectifs sont essentiellement la recherche de nouvelles particules.

### 2.4.1 ALICE : A Large Ion Collider Experiment

L'expérience ALICE s'inscrit dans un cadre un peu spécifique étant donné quelle est dédiée à une utilisation du collisionneur en mode ionique, c'est-à-dire lorsque celui-ci accélérera des ions et non plus des protons. Par ions on entend ions issus d'éléments plus lourds que l'hydrogène [11].

#### 2.4.1.1 : LES OBJECTIFS D'ALICE

L'intérêt d'utiliser des ions en lieu et place de simples protons est que les ions étant plus lourds, ils seront plus énergétiques. La contre partie de ce raisonnement est que les ions contiennent un nombre de particules élémentaires (quarks) encore plus grand que le

proton, qui n'en a que trois. Ainsi dans le cas de collisions d'ions de plomb, l'énergie est certes plus élevée mais aussi répartie sur un nombre de quarks beaucoup plus grand.

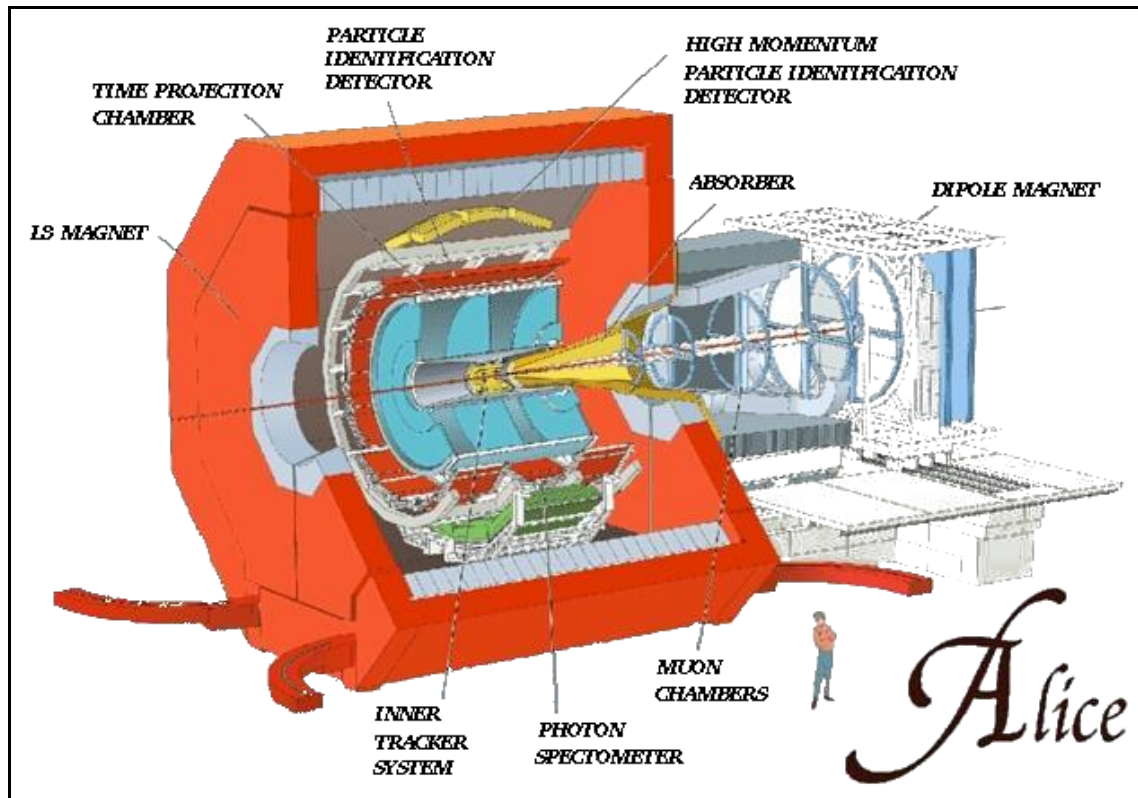


Figure 7 : Détecteur ALICE

Dans ce cas on ne peut pas espérer faire d'observation de nouvelle particule, mais on peut espérer créer ce que l'on appelle un plasma de quarks et gluons (QGP : Quark Gluon Plasma), c'est à dire une "soupe" de quarks et gluons, ce qui doit correspondre à l'état de notre univers juste après le Big Bang. Il est possible que cet état de la matière QGP existe encore de nos jours, notamment au sein des étoiles à neutrons.

L'intérêt essentiel sera dans ce cas alors d'étudier comment, à partir de cette soupe, les quarks et les gluons se recombinent pour donner naissance aux particules que nous connaissons mieux, telles que le proton ou le neutron. Sur le plan technique, une des difficultés auxquelles sera confrontée ALICE est le taux de particules qui vont naître de ce plasma. La quantité de particules sera beaucoup plus importante que dans le cas d'une

simple collision proton–proton, ce qui va engendrer une quantité très importante de données à acquérir, à traiter puis à analyser.

### 2.4.2 LHCb : A Large Hadron Collider Beauty experiment

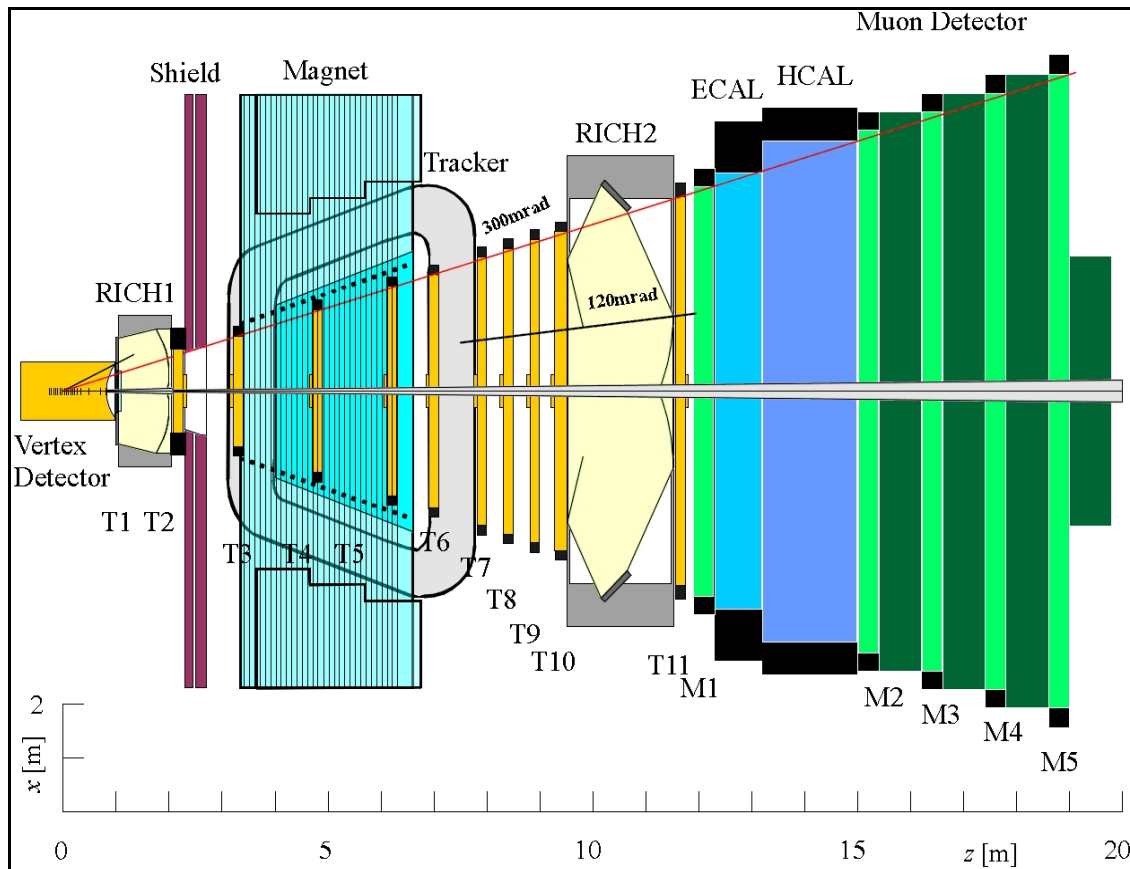


Figure 8 : Vue en coupe du détecteur LHCb

L'expérience LHCb [12] est la seconde expérience spécialisée mise en place auprès du LHC. Son objectif est, comme son nom l'indique, l'étude de la physique du quark beau. Le détecteur LHCb présente la particularité de ne pas être un détecteur symétrique, comme on peut le voir sur le schéma ci-dessous. Ce détecteur profitera du fort taux de création de b-hadrons dans le LHC pour étudier notamment la violation de la symétrie CP. C'est notamment à travers l'étude des mésons beaux que des violations de la symétrie CP sont attendues. Les triggers de LHCb devront en particuliers être capables de détecter la présence de muons de grande impulsion transverse.

Le point important concernant le détecteur LHCb est notamment la possibilité de reconstruire le vertex de la désintégration du hadron  $b$ , ainsi qu'une bonne identification des particules chargées. Cela nécessite la présence d'un important champ magnétique de l'ordre de 1.1 Tesla.

Un autre point important concerne l'identification et la séparation des particules issues de la désintégration des mésons beaux. Cette identification sera notamment réalisée à l'aide de la présence de deux détecteurs de type RICH (Ring-Imaging Cherenkov).

### 2.4.3 CMS : Compact Muon Solenoid

Le détecteur CMS [13] est un détecteur généraliste qui a vocation à explorer tous les secteurs de la physique auprès du LHC, notamment la recherche de nouvelles particules, ce qui correspond aussi à la tâche qui incombe au détecteur ATLAS. Le choix de construire deux détecteurs qui *a priori* ont les mêmes objectifs n'est pas innocente, car si les objectifs sont les mêmes, les moyens d'y parvenir diffèrent. En particulier le détecteur CMS s'avérera plus sensible que ATLAS dans certains canaux de recherche du Higgs et inversement pour d'autres canaux.



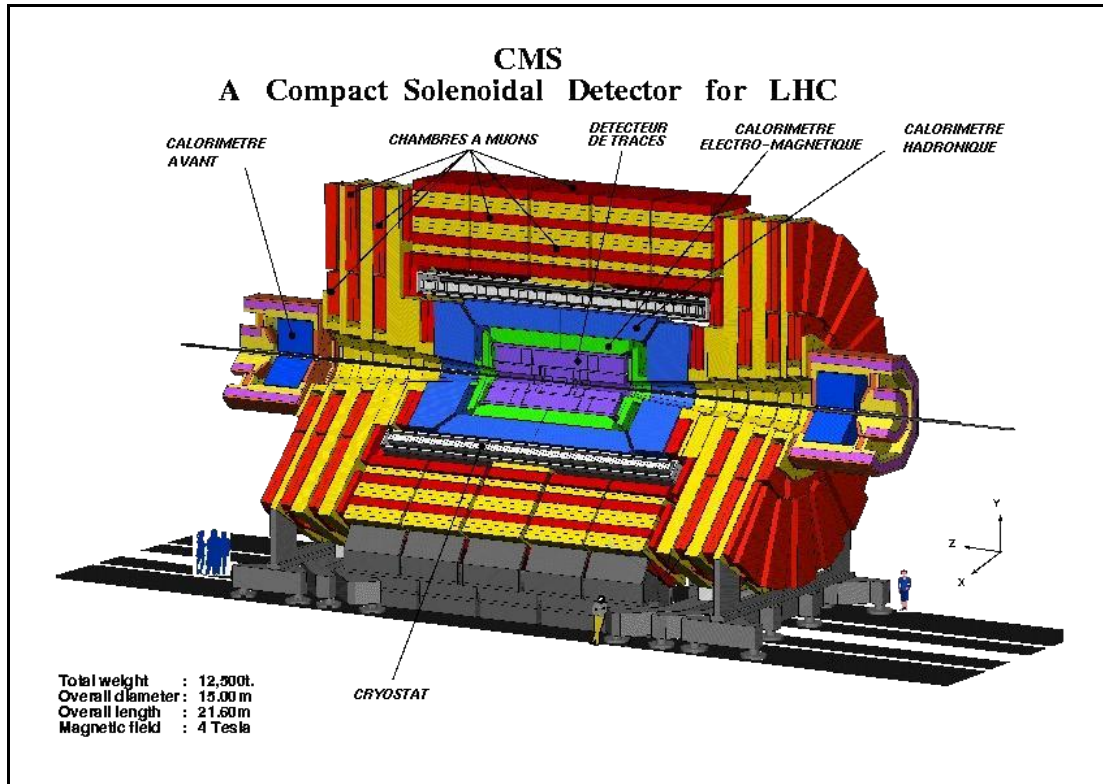


Figure 9 : Détecteur CMS

D'un point de vue structurel, le détecteur CMS diffère très peu du détecteur ATLAS. Une des différences essentielles réside dans le fait que le calorimètre de CMS est un calorimètre un peu spécial constitué de plusieurs milliers de cristaux. De plus il possède un champ magnétique intense de plus de 4 Tesla créé par un aimant supraconducteur qui se trouve au coeur du détecteur.

## 2.5 ATLAS : A Toroidal Lhc Apparatus

Le détecteur ATLAS sera le plus important des 4 détecteurs qui seront mis en place autour du LHC. Tout comme le détecteur CMS, ATLAS est un détecteur à vocation généraliste dont les objectifs essentiels sont la recherche du boson de Higgs et l'hypothétique découverte de superparticules. Actuellement près de 160 instituts et 2000 personnes sont engagés dans sa réalisation.

## 2.5.1 CARACTERISTIQUES GENERALES

Le détecteur ATLAS [8] est un cylindre de près de 22 mètres de diamètre et de 44 mètres de longueur pour un poids de l'ordre de 7000 tonnes. Toute son architecture est organisée en plusieurs couches cylindriques qui s'articulent autour du point d'interaction.

ATLAS peut être découpé en trois sous-systèmes correspondant à ce que l'on appelle des sous-détecteurs : un détecteur de traces (ou détecteur interne) placé au plus près du point d'interaction; un détecteur d'énergie (calorimètre); enfin un détecteur à muons qui représente la partie la plus externe du détecteur et qui est un détecteur spécifique à un type de particules, le muon.

A cela il faut ajouter la présence de deux aimants, un au niveau du détecteur de traces et un au niveau de détecteur de muons. Le premier de ces aimants est un solénoïde générant un champ magnétique de 2 Tesla et entourant le détecteur de traces. Le système magnétique associé au détecteur à muons est lui un imposant aimant supraconducteur toroïdal.

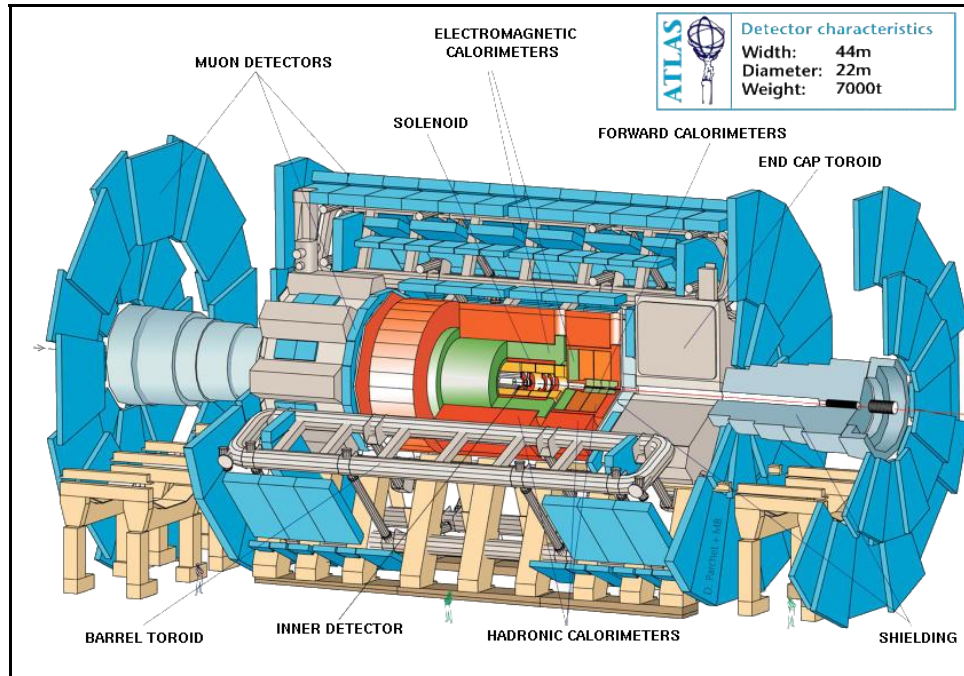


Figure 10 : Détecteur ATLAS

Enfin un dernier sous-système non encore évoqué est le système de déclenchement et d'acquisition de données qui a en charge la gestion, la collecte et le traitement de l'ensemble des données issues des trois sous-détecteurs.

Regardons un peu en détail chacun des sous-détecteurs d'ATLAS.

### **2.5.2 DETECTEUR INTERNE OU DE TRACE**

Le détecteur interne est le plus proche du point de collision des protons. C'est donc le premier qui sera traversé par les produits de la collision. Mais c'est aussi celui qui sera soumis à la plus forte dose de rayonnement, ce qui a pour conséquence de devoir prendre en compte ce paramètre dès sa conception, notamment en prenant en compte sa durée de vie et la possibilité de le remplacer après une certaine période. C'est un cylindre de 6,8 mètres de long pour un rayon maximal de 1,15 mètres.

Le détecteur interne est un ensemble de trois détecteurs différents. L'objectif des ces trois détecteurs est le même : cartographier la trajectoire des différentes particules qui les traversent, d'où le nom de détecteur de traces. Les détecteurs de traces sont en fait capables de traquer les particules chargées comme les électrons ou les positrons. Par contre, ils sont incapables de voir les particules neutres comme le photon. La connaissance des traces, couplée à la présence du champ magnétique généré par le solénoïde, permet de remonter à certaines caractéristiques physiques des particules comme l'impulsion, la charge, mais aussi de déterminer la position des vertex secondaires.

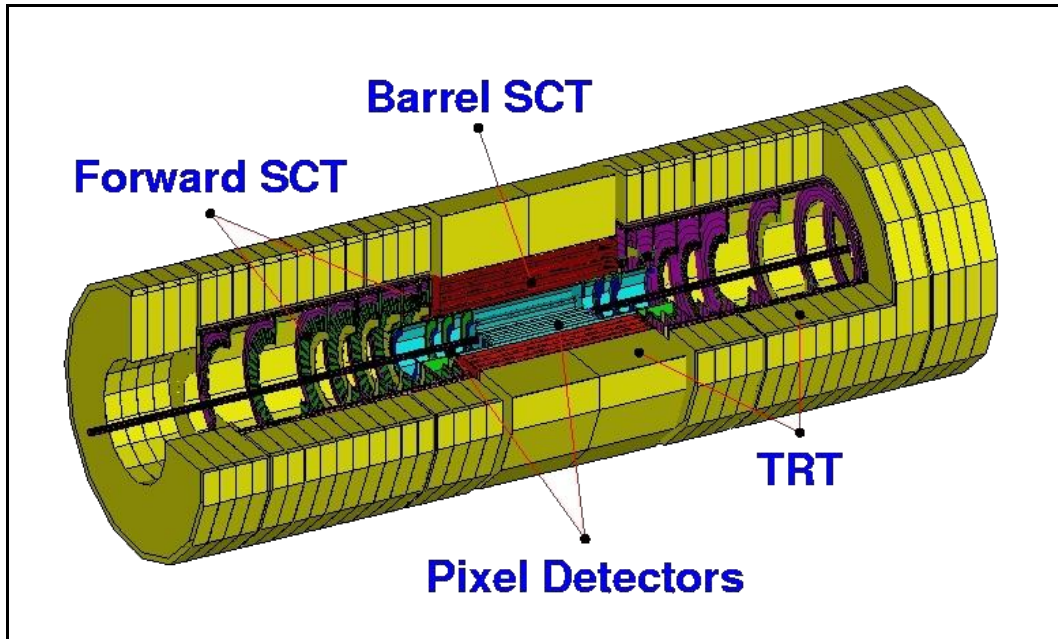


Figure 11 : Détecteur interne d'ATLAS

Les traces des particules chargées se déterminent par reconstruction de leurs trajectoires, à partir de plusieurs points de passage de la particule, points de passage donnés par les détecteurs internes.

Regardons de plus près les trois sous-détecteurs qui constituent le détecteur interne :

– Le détecteur à pixels

Il est constitué de trois couches cylindriques concentriques ( $R= 4 \text{ cm}$ ,  $11 \text{ cm}$  et  $14 \text{ cm}$ ) d'une matrice de pixels en silicium ainsi que de quatre disques de part et d'autres des cylindres. Chaque pixel de la matrice peut être interrogé pour savoir s'il a été traversé ou pas. On obtient ainsi une collection de points représentant les pixels touchés. La couche de pixels appelée couche B est placée extrêmement près du faisceau et sera particulièrement utilisée pour déterminer la position des vertex secondaires. La taille des pixels diffère suivant leurs positions et les couches, mais typiquement elle est de l'ordre de  $50 \mu\text{m}$  dans la direction  $R\phi$  et  $300 \mu\text{m}$  dans la direction  $z$ .

– Le détecteur à micro-rubans (SCT : Semi Conductor Tracker)

Il est constitué de quatre cylindres de semi-conducteurs qui, à la différence du détecteur précédent, sont agencés en bandes et non plus en pixels. Le SCT se trouve juste après le détecteur à pixels et sa résolution spatiale est de  $16 \mu\text{m}$  dans la direction  $R\phi$  et  $580 \mu\text{m}$  dans la direction  $z$ .

– Le détecteur à rayonnement de transition (TRT : Transition Radiation Tracker)

Il est composé de tubes dans lesquels se trouve un gaz et où court un fil. Le passage d'une particule ionise le gaz. La cascade électronique qui s'ensuit est récupérée sur le fil. Les TRT serviront notamment à l'identification des électrons.

Les principales performances requises pour les détecteurs internes sont:

- ✓ Reconstruction de traces pour  $|\eta| < 2.5^3$
- ✓ Efficacité de reconstruction supérieure à 95 % pour des particules chargées isolées d'impulsion supérieures à 5 GeV.
- ✓ Capacité de reconstruire le vertex déplacé des désintégrations du  $b$
- ✓ Pour des impulsions de 500 GeV une résolution en impulsion de  $\Delta P_t/P_t < 30 \%$  pour  $|\eta| < 2$  et de 50 % pour  $2 < |\eta| < 2,5$

L'une des caractéristiques communes aux trois détecteurs est de présenter un volume de matière assez faible. Ceci est une contrainte très importante sur le détecteur interne. En effet, il doit être le plus transparent possible pour les particules afin de ne pas fausser les mesures qui seront faites dans les détecteurs suivants, à savoir les calorimètres.

---

<sup>3</sup>  $\eta$  est la pseudo-rapacité, correspondant approximativement à l'inclinaison des traces par rapport à l'axe principal du détecteur.  $\eta = - \ln(\tan(\theta/2))$  où  $\theta$  est l'angle polaire.

### 2.5.3 LES CALORIMETRES

Derrière le détecteur interne, on trouve un autre type de détecteur : les calorimètres . Ils sont au nombre de deux et leur tâche est de mesurer les énergies des particules. Celles-ci sont dans un premier temps ralenties par un premier milieu dit absorbeur, constitué en général d'une matière dense. Dans ce milieu, la particule incidente crée une gerbe de particules filles qui sont, à leur tour, ralenties par le milieu et ainsi de suite. Derrière cette phase d'absorption se trouve un milieu dit de détection qui va mesurer les dépôts d'énergie. On alterne ainsi une couche d'absorption et une de détection pour former une structure de type sandwich. Ce type de calorimètre est défini comme un calorimètre à échantillonnage. A l'opposé de ce modèle, il existe une gamme de calorimètres où le milieu absorbeur et détecteur est le même : c'est ce que l'on appelle des calorimètres homogènes.

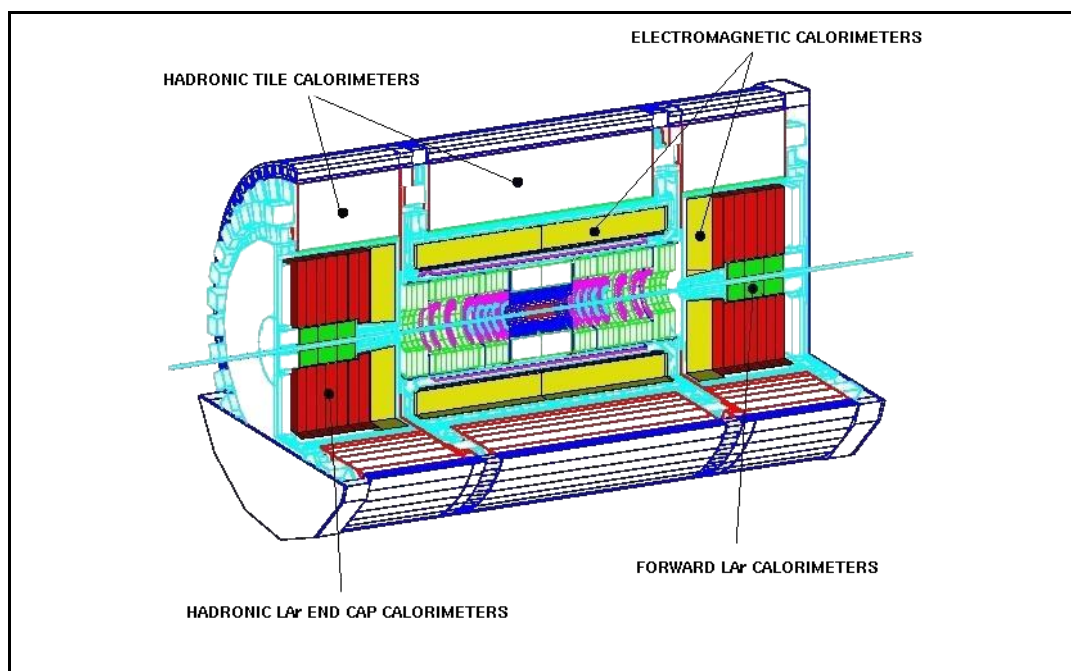


Figure 12 : Les calorimètres d'ATLAS

On trouve généralement deux types de calorimètres : un premier, optimisé pour me-

sur les énergies des électrons et photons : c'est le calorimètre électromagnétique. Le second calorimètre, se situant derrière, mesure l'énergie des particules hadroniques : c'est le calorimètre hadronique.

#### – Le Calorimètre électromagnétique

Son but est de mesurer le plus précisément possible l'énergie des électrons et photons. Il est constitué de deux parties. Une première appelée *tonneau* vient se placer autour du détecteur interne. Cette partie va jusqu'à  $|\eta| = 1,475$ . La seconde partie est composée de deux *bouchons* qui viennent fermer le tonneau des deux cotés. Chaque bouchon est composé de deux roues concentriques qui couvrent le domaine  $1,375 < |\eta| < 3,2$ .

Ce calorimètre est un calorimètre à échantillonnage dont le milieu absorbant est composé de plaques de plomb et le milieu détecteur est composé d'argon liquide. La présence d'argon à l'état liquide nécessite que l'ensemble soit maintenu à basse température dans un cryostat dans lequel baignera l'ensemble du calorimètre. Le sandwich constituant le calorimètre aura également une forme particulière en accordéon afin de rendre le calorimètre le plus hermétique possible. La longueur de radiation sera de  $24 X_0$  pour le tonneau et de  $26 X_0$  pour les bouchons.

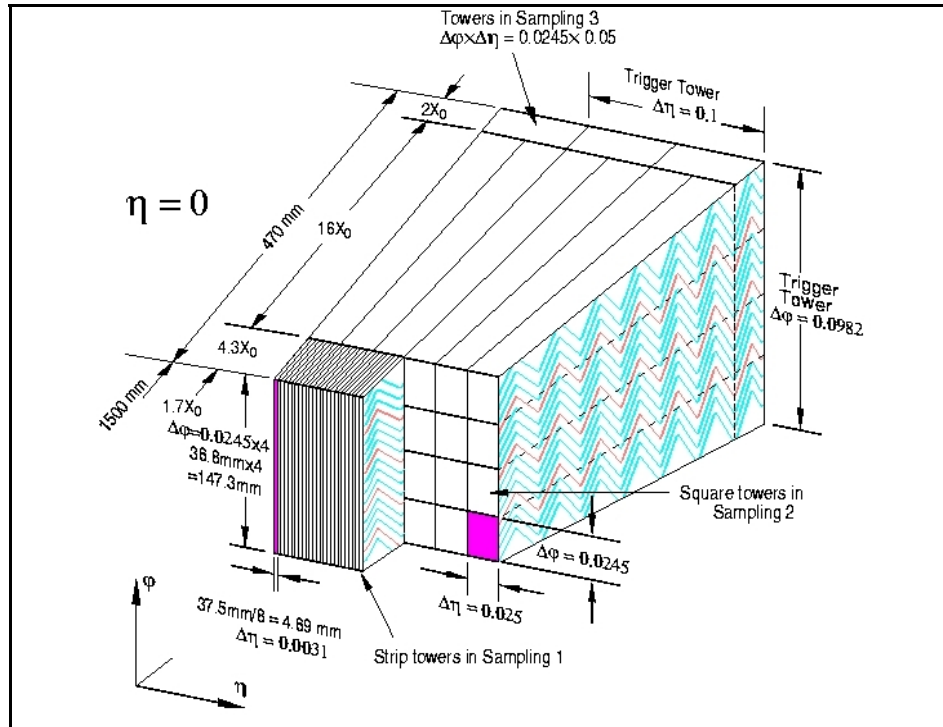


Figure 13 : Vue schématique d'un secteur angulaire du tonneau du calorimètre électromagnétique, avec ses différents compartiment et de la structure en accordéon.

Le calorimètre électromagnétique d'ATLAS sera constitué de trois compartiments (Sampling) et pour certains secteurs d'un pré-échantillonneur. La granularité des différents compartiments sera différente, de même qu'elle différera des bouchons au tonneau. Le pré-échantillonneur sert essentiellement pour l'identification des particules à des fins de séparation ( $\pi^0/\gamma$ ).

Les principales performances qui sont demandées au calorimètre électromagnétique sont :

- ✓ Identification des jets, électrons et photons de 1 GeV à 5 TeV.
- ✓ Mesure des énergies dans la gamme 50 MeV à 3 TeV
- ✓ Une résolution en énergie de  $\Delta E/E = (10\%/\sqrt{E(\text{GeV})}) \oplus 1\%$  dans la gamme d'énergie 10 à 300 GeV



### – Le Calorimètre hadronique

Le calorimètre électromagnétique étant optimisé pour les électrons et photons, c'est au calorimètre hadronique qu'incombe de mesurer l'énergie des divers hadrons présents dans l'événement. Celui-ci aussi sera un calorimètre à échantillonnage composé de plaques de fer et de cuivre pour la partie absorbante et de plaques de scintillateur pour la partie détection. La lumière créée dans les scintillateurs sera captée par une série de photomultiplicateurs afin de générer le signal. Ici le calorimètre sera organisé sous la forme d'un tonneau ( $|\eta| < 1,0$ ), de deux extensions du tonneau ( $0,8 < |\eta| < 1,7$ ) et de deux bouchons ( $1,5 < |\eta| < 3,1$ ) constitués de deux roues chacun. On trouve également un calorimètre dit "avant" constitué deux bouchons couvrant la zone extrêmement proche des faisceaux ( $3,1 < |\eta| < 4,9$ ).

Du point de vue de la résolution en énergie les performances attendues sont :

- ✓  $\Delta E/E = (50\%/\sqrt{E(\text{GeV})}) \oplus 3\%$  pour  $|\eta| < 3$
- ✓  $\Delta E/E = (100\%/\sqrt{E(\text{GeV})}) \oplus 10\%$  pour  $3 < |\eta| < 5$

## 2.5.4 LES CHAMBRES A MUONS

On arrive enfin au dernier détecteur d'ATLAS. C'est aussi le plus externe et surtout le plus volumineux. Il s'agit du détecteur à muons qui est composé d'un ensemble de détecteurs que l'on appelle chambres à muons. Le muon est un électron lourd qui peut traverser l'ensemble des détecteurs internes et calorimètres sans dommage. C'est pourquoi on peut placer ces chambres à muons le plus à l'extérieur. Les chambres à muons sont en fait des tubes où se trouve un gaz qui va être ionisé par le passage de la particule. C'est cette ionisation que l'on va détecter afin de déterminer la position du point de passage du muon.

Il existe différents types de chambres à muons tels que Monitored Drift Chamber (MDC), Cathode Strip Chamber (CSC), Resistive Plate Chambers (RPC) et les Thin Gap Chambers (TGC). De façon analogue à ce qui se passe dans le détecteur interne, un

champ magnétique toroïdal est appliqué à l'ensemble des chambres à muons pour mesurer l'impulsion des particules.

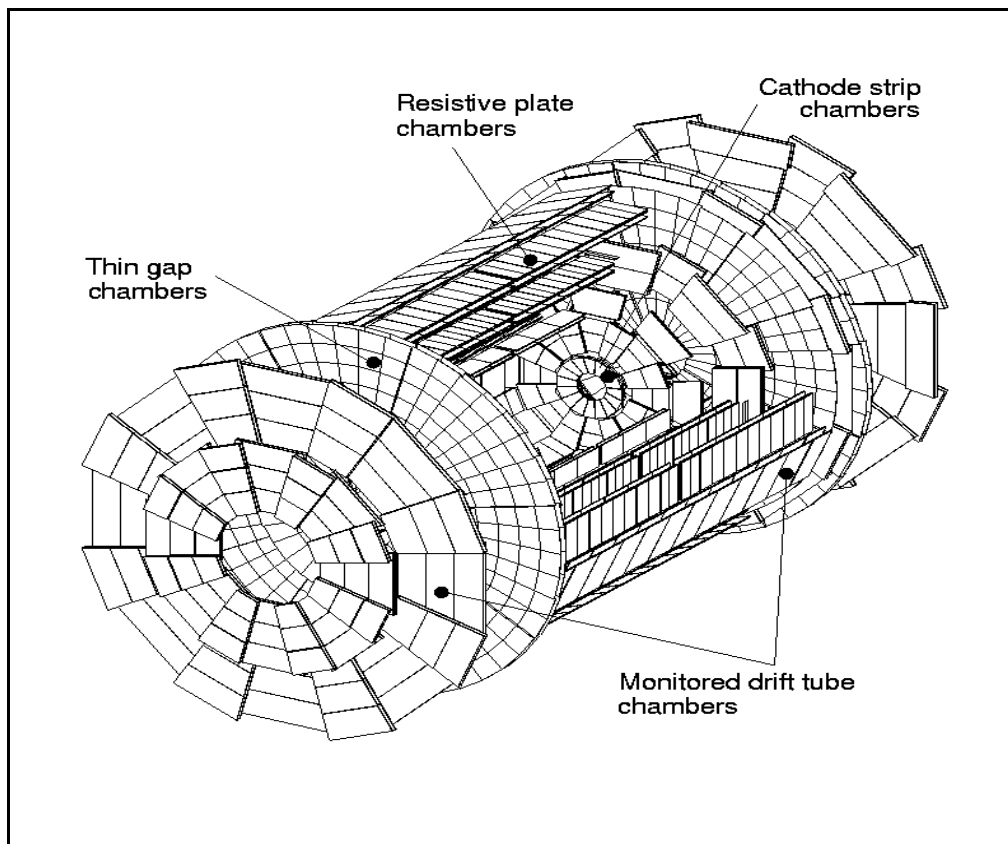


Figure 14 : Les chambres à muons d'ATLAS

L'un des points cruciaux concernant ces chambres à muons réside dans la gestion du bruit de fond qui peut être très important et créer un taux de bruit parasite relativement important.

On attend du détecteur à muons une résolution en impulsion de l'ordre de 2% pour un domaine d'énergie de 20 à 200 GeV.

On vient de passer en revue, de façon sommaire, les différents sous-détecteurs qui vont constituer ATLAS. Mais il faut garder à l'esprit que, quel que soit le phénomène physique utilisé pour effectuer cette détection, que ce soit par ionisation de gaz, par création de trous dans un semi-conducteur ou autre, les détecteurs n'ont qu'un seul but :

créer une information. Il nous reste encore à exploiter cette information. Mais avant, il faut l'acquérir, la contrôler, la trier. C'est le rôle du quatrième sous ensemble du détecteur, à savoir le système d'acquisition et de déclenchement.

## 2.6 RESUME

Après un rappel sur les accélérateurs, ce chapitre décrit le futur accélérateur hadronique (LHC) en cours de réalisation au CERN ainsi que les quatre expériences qui y sont attachées. Les différents sous-détecteurs du détecteur ATLAS sont décrits de façon plus complète.

## CHAPITRE 3

# ACQUISITION ET DECLENCHEMENT

Au sein du détecteur, ATLAS les événements seront générés à une fréquence de 40 MHz. Si on se base sur les sections efficaces de production et de désintégration des particules et les connaissances des caractéristiques du détecteur, on peut estimer que chaque événement correspondra à environ 2 Mo de données générées, ce qui signifie que le flux de données au sein du détecteur sera de l'ordre de  $8 \cdot 10^{13}$  octets/s (80 To/s).

Cette masse de données, générées au niveau des détecteurs, va devoir être collectée par ce que l'on appelle la chaîne d'acquisition. Cette chaîne a en charge la gestion de ce flot de données. Le système d'acquisition n'est pas localisé en un point du détecteur, mais commence au niveau le plus bas de chaque sous détecteur et va jusqu'aux unités de stockage.

Un autre point crucial concernant les données est le volume de celles-ci. En effet, le LHC est prévu pour fonctionner durant une quinzaine d'années. Il est totalement impensable, mais surtout techniquement impossible, de stocker un flux de 80 To/s durant une telle période. C'est pourquoi dans la chaîne d'acquisition d'ATLAS se trouve une série de systèmes de déclenchement (Triggers) qui vont diminuer le volume des données et cela durant l'acquisition[15].

---

Enfin un dernier point important doit être abordé. C'est celui qui concerne la supervision et le contrôle du système: supervision à la fois du flot des données, du système de déclenchement, mais aussi de l'état des différents composants du détecteur .

Dans ce chapitre, acquisition et déclenchement sont décrits séparément, bien que ces deux composantes soient extrêmement liées dans la réalité, et cela afin de rendre plus claire la compréhension du système. Le diagramme (Figure 15) montre de façon schématique les différents sous-systèmes qui composent le système de déclenchement et acquisition d'ATLAS et qui vont être décrits dans ce chapitre.

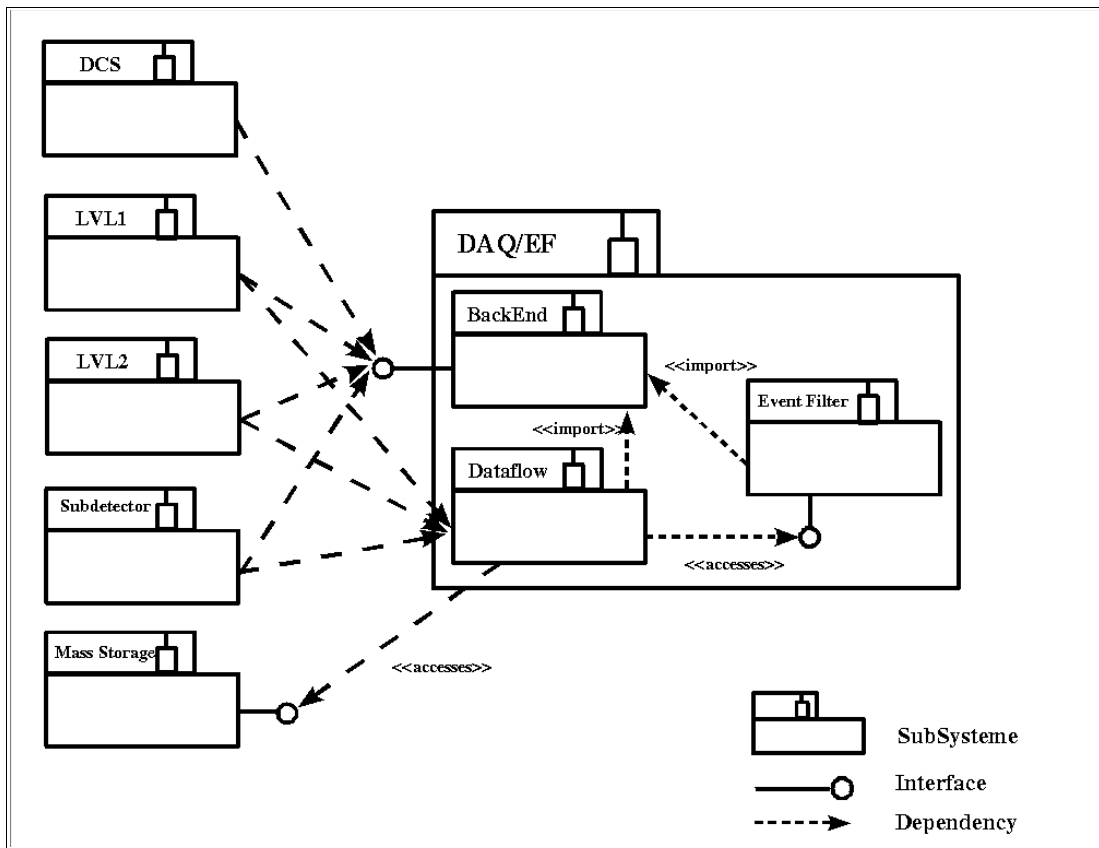


Figure 15 : Diagramme UML du système d'acquisition et de déclenchement d'ATLAS

## 3.1 DECLENCHEMENT

L'impossibilité de stocker les 80 Teraoctets de données qui sortiront par seconde du détecteur ATLAS a conduit à la décision d'insérer dans la chaîne d'acquisition du détecteur trois étages de déclenchement (ou triggers) qui vont ramener ce flot à une valeur de 200 Mo/s environ, au niveau des unités de stockages. Cette valeur de sortie a été déterminée en fonction de la quantité d'événements intéressants pour la physique que l'on peut attendre et aussi des ressources qui seront disponibles ultérieurement pour analyser ces données.

### 3.1.1 LES OBJECTIFS DES TRIGGERS

Cet objectif de passer de 80 To/s à 200 Mo/s sera réalisé à travers l'utilisation de trois modules différents nommés niveau 1 (LVL1), niveau 2 (LVL2) et filtre d'événements (Event Filter).

La solution envisagée est basée sur l'idée que la physique que l'on va chercher à mettre en évidence dans ATLAS sera noyée au milieu d'une physique bien connue. Un événement comportant un boson de Higgs, par exemple, ne représentera qu'un événement sur plusieurs millions, les autres événements ne reflétant qu'une physique déjà connue ou représentative d'un bruit de fond (collisions sur des molécules résiduelles dans l'accélérateur). L'idée consiste alors à rejeter le maximum d'événements qui ne sont pas susceptibles de porter une nouvelle physique. On va chercher dans cette méthode à enrichir notre lot d'événements en événements intéressants. Dans cette approche, la taille des événements reste la même, c'est le taux de ces événements qui diminue au fur et à mesure qu'ils passent dans les filtres. Pour être réalisable, cette méthode suppose que l'on sache, jusqu'à un certain point, dire si un événement est intéressant ou pas.

L'idée de compresser certaines données (notamment en fin de chaîne d'acquisition) n'est pas totalement exclue afin de diminuer un peu la taille des unités de stockage nécessaires. L'inconvénient d'une telle compression, qui se doit d'être non destructive, est de réclamer des ressources de calcul importantes.

### 3.1.2 LES CONTRAINTES SUR LES TRIGGERS

Le système de déclenchement d'ATLAS doit répondre à un ensemble de contraintes qui sont:

- ✓ Accepter tous les événements potentiellement intéressants au niveau de la physique. Cela nécessite l'utilisation d'algorithmes de qualité afin d'éliminer la possibilité de biaiser la sélection.
- ✓ Rejeter au maximum les bruits de fond d'origines diverses, afin d'enrichir l'échantillon en événements intéressants.
- ✓ Flexibilité : on veut dans une certaine mesure pouvoir changer les critères de sélection des événements.
- ✓ Respecter les contraintes de temps, la sélection devant s'effectuer dans le temps qui lui est imparti.
- ✓ Sécurité des données: on veut être sûr de ne pas perdre des événements potentiellement intéressants durant la phase de sélection.

### 3.1.3 VUE GLOBALE DES TRIGGERS

Le schéma ci-dessous (Figure 16) montre les trois niveaux de sélection du détecteur ATLAS et leur placement dans la chaîne d'acquisition, que nous décrivons ultérieurement.

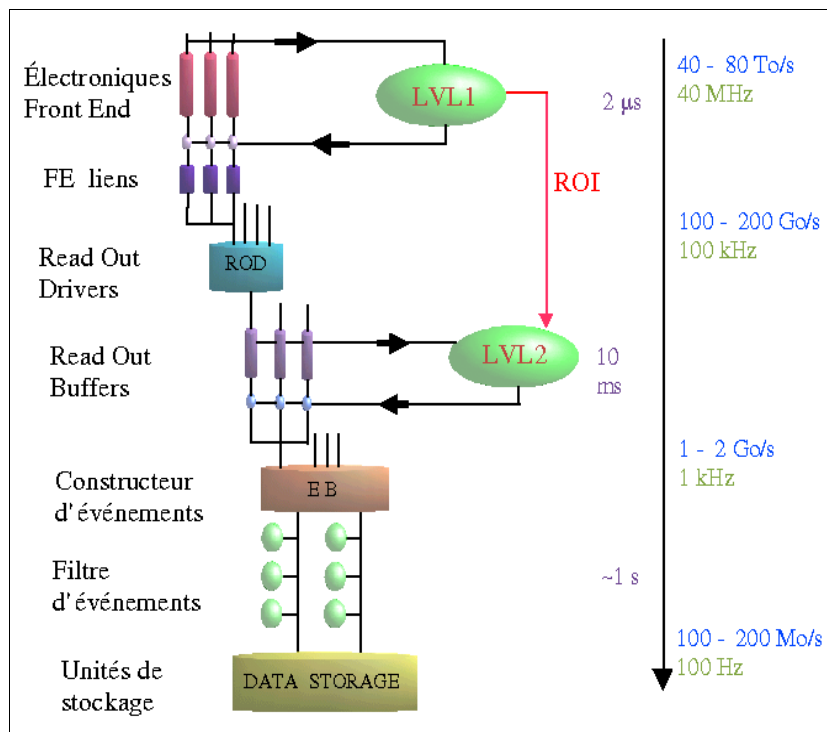


Figure 16 : Chaîne acquisition et triggers d'ATLAS

### 3.1.4 LVL1

Le premier niveau de trigger ou LVL1 est un niveau de sélection purement électronique. D'un point de vue géographique, il est localisé au niveau des détecteurs. Ce niveau doit effectuer sa sélection dans un laps de temps de  $2 \mu\text{s}$ .

Étant situé au niveau des détecteurs la sélection ne porte que sur les fragments d'information connus au niveau du détecteur et en aucun cas sur l'événement dans sa totalité.

Si l'un des fragments d'un événement passe les critères de sélection, une information



est transmise afin d'autoriser tous les fragments issus de cet événement à être envoyés dans un ensemble de mémoires tampons : les ROD (Read Out Drivers). Les informations qui arrivent aux ROD sont numériques mais, suivant les détecteurs, la phase de numérisation des signaux a lieu avant ou après que le trigger ait donné son autorisation de transfert des données dans les ROD, permettant ainsi de ne numériser dans certain cas que si l'événement passe le seuil de déclenchement du premier niveau.

Étant donné le temps qui est imparti au LVL1 pour prendre sa décision, les détecteurs les plus lents en réponse ne sont pas pris en compte dans ce niveau de trigger. Seuls les calorimètres et certaines chambres à muons sont utilisés dans cette prise de décision. Les critères qui vont amener à cette prise de décision sont de plusieurs ordres. Ils correspondent pour les calorimètres, au dépassement d'un seuil, à des dépôts d'énergie dans certains secteurs des calorimètres. Dans le cas des chambres à muons, le critère va porter essentiellement sur la présence d'une ou plusieurs particules ayant un moment transverse supérieur à un certain seuil.

Le trigger de premier niveau va ainsi créer une information de première importance pour les triggers suivants. Il va définir les secteurs du détecteur où se concentre l'information utile, c'est ce que l'on appelle les ROI (Regions Of Interest). La localisation des ROI et leur contenu en termes d'information physique (comme la présence de jets, le franchissement d'un seuil en énergie, ...) vont être transmises au niveau deux, pour lui permettre de connaître immédiatement quelles sont les zones du détecteur intéressantes à analyser, supprimant ainsi le besoin de parcourir l'ensemble du détecteur.

Ces informations sont également envoyées dans la chaîne d'acquisition afin d'être disponibles pour une analyse ultérieure. Du point de vue du système d'acquisition, le trigger de niveau un est un détecteur comme les autres, dont les données sont disponibles comme pour un détecteur classique. Les ROI, auxquelles le niveau deux de trigger devra se référer rapidement, sont également envoyées à un élément, le CTP (Central Trigger Processor), qui va déclencher le trigger de niveau deux, tout en lui passant l'information des ROI.

### 3.1.5 LVL2

Le deuxième niveau de trigger doit, lui, diminuer d'un facteur 50 à 100 le taux de données. Le LVL2 doit donc faire passer le flux de données à un taux de 1 à 2 kHz en sa sortie et cela dans un temps maximum de l'ordre de 10 ms. Ici on dispose de plus de temps, car n'oublions pas que, à ce niveau là nous avons 100 kHz de données entrantes, ce qui laisse un peu plus de temps pour faire la sélection que n'en avait le trigger de niveau un [16].

Le CTP va déclencher le trigger de niveau deux et lui passer l'information concernant les ROI, permettant ainsi au trigger de connaître les zones du détecteur intéressantes.

Le temps, de l'ordre de quelques millisecondes, imparti à ce trigger lui permet de faire une sélection sur des critères plus complets que ce qu'a fait le niveau précédent. Le niveau un n'a fait sa sélection que sur des seuils en énergie dans les calorimètres et sur une identification grossière des muons. Les critères de sélection du niveau deux seront plus complets, plus précis et ils porteront sur des arguments liés à la physique de l'événement.

Le LVL2 est capable de lire à la demande les données contenues dans des mémoires tampons (ROB : Read Out Buffer). Connaissant les ROI il est en mesure de n'interroger que les buffers contenant une information intéressante afin de reconstruire, à partir de ces données, des objets physiques tels que l'énergie déposée par un jet, puis appliquer à ces objets des critères de sélection. Cependant les ROB ne contenant que des fragments d'événement, le niveau deux pourra, certes reconstruire des objets physiques simples, mais il n'aura pas accès à l'ensemble des objets physiques de l'événement.

La réalisation technique de ce niveau n'est pas encore clairement établie . Une chose est sûre, c'est qu'il sera composé de processeurs, qui pour certains d'entre eux seront assistés de FPGA<sup>4</sup>. L'accès aux ROB se fera à travers un réseau non encore précisément défini, qui servira également pour accéder aux diverses bases de données qui seront

---

<sup>4</sup> FPGA: Field Programmable Gate Array: Les FPGA sont des composants contenant un grand nombre de portes logiques qui peuvent être programmées, ce qui en fait des composants capables d'exécuter des tâches simples à très grande vitesse.

requis par les programmes de reconstruction d'objets physiques qui tourneront sur les processeurs.

Les décisions du trigger de niveau deux seront également passées au filtre d'événements. Cependant la décision n'a pas encore été prise de savoir si ces informations devaient être écrites dans un ROB spécifique ou passées au niveau suivant par un autre mécanisme comme leur écriture dans une base de données.

Le troisième niveau de trigger, le filtre d'événements auquel est entièrement consacré le chapitre suivant, ne sera pas détaillé ici. Le filtre d'événements et le trigger de niveau deux constituent les filtres de hauts niveaux (HLT : High Level Triggers). Nous allons voir maintenant le système d'acquisition des données .

## **3.2 ACQUISITION DES DONNEES**

Dans le chapitre précédent, on a passé en revue sommairement les différents types de sous-détecteurs qui composent ATLAS. Ces sous-détecteurs ont en commun de transformer un phénomène physique (celui qui est à l'origine de la détection) en un signal électrique. Le système d'acquisition a en charge de collecter ces diverses informations. C'est un système d'une extrême importance, car c'est lui qui va permettre aux physiciens d'accéder ultérieurement à la mesure nécessaire pour l'analyse.

### **3.2.1 OBJECTIFS**

L'objectif unique du système d'acquisition des données est la collecte auprès des détecteurs, des informations issues de la physique et leur acheminement aux unités de stockage.

### 3.2.2 CONTRAINTES

Assumer la tâche qui lui incombe est la première des contraintes auxquelles est soumis le système d'acquisition. Il doit pouvoir assurer sa tâche de collecte dans le temps qui lui est imparti. N'oublions pas qu'au sein d'ATLAS, on aura une collision génératrice d'information toutes les 25 nanosecondes. Cette tâche est d'autant plus difficile que les détecteurs étant de natures différentes, ils ont des temps de réaction très disparates. C'est pourquoi les diverses données devront être marquées par un identifiant donné par l'accélérateur lui-même (signal TTC<sup>5</sup>) afin de pouvoir reconnaître les informations issues d'un même événement .

L'acquisition représente un point sensible dans un système de prise de mesures. En effet, on peut imaginer acquérir de l'information avec une partie du détecteur en panne, mais en aucun cas si c'est le système d'acquisition qui l'est. Un système d'acquisition défaillant et c'est la totalité du système qui est en échec. C'est pourquoi le système d'acquisition d'ATLAS doit être d'une grande robustesse et tolérant à quasiment tous les types de pannes et problèmes imaginables.

La diversité des détecteurs est aussi à l'origine de deux caractéristiques du système d'acquisition, qui sont sa modularité et sa « généralité ». Comme nous l'avons fait remarquer précédemment, les différents sous-détecteur d'ATLAS diffèrent énormément notamment en ce qui concerne les informations qu'ils fournissent : certains donnent des positions, d'autres des dépôts d'énergie, certains des temps de dérives. Le type d'information, mais aussi le format et le codage de l'information n'ont aucune corrélation avec le détecteur voisin, sinon qu'elles sont issues d'un même événement. Le système d'acquisition, lui, se doit de fédérer cet ensemble de données. Il doit être le moins spécifique possible vis à vis des détecteurs.

Enfin c'est au niveau de ce système que sera défini le format des informations qui seront analysées par la suite, C'est donc à ce niveau que doit être réarrangé, réorganisé l'ensemble de l'information de la façon la plus pratique possible pour les analyses ultérieures.

---

5 TTC : Timing, Triggers and Contrôle : signal d'horloge servant de référence à l'ensemble des systèmes liés au LHC.

### 3.2.3 ACQUISITION DE BAS NIVEAU

Les notions d'acquisition de bas et de haut niveau ne sont pas qualitatives, mais simplement pratiques. On différencie ainsi la partie de l'acquisition où l'on travaille sur des fragments de l'événement, de la partie où l'on travaille sur l'événement dans sa totalité.

Au niveau des détecteurs, les informations sont initialement des impulsions électriques qui sont numérisées par un système électronique. Ces données sont envoyées à travers un composant appelé « Front End » (FE) dans un autre composant le « Read Out Drivers » (ROD). Les Front End sont les interfaces de sortie des détecteurs qui possèdent une petite mémoire tampon (buffer) (Figure 17) permettant le stockage de quelques événements.

Les liaisons entre les Front End et les ROD sont réalisées par des liens, Front End Link (FEL), qui peuvent être de deux types, soit électrique, soit optique, suivant les détecteurs et la distance qui les sépare des ROD. En effet les liaisons optiques fonctionnent jusqu'à 300 mètres alors que les électriques sont limitées à moins de 30 mètres. Les liaisons optiques sont, par contre, plus sensibles au vieillissement par les radiations.

#### Les ROD: Read Out Drivers

Les Read Out Drivers sont schématiquement de simples mémoires tampons qui recueillent les données arrivant des FEL. En réalité les ROD ne se limitent pas nécessairement à un rôle de simple buffer, il est tout à fait possible d'envisager de leur allouer un potentiel de traitement ne serait-ce que pour formater les données par exemple. Bien entendu ce traitement doit être simple et rapide.

On compte plus de 1600 ROD dans ATLAS. Le tableau ci-dessous (Table 5) montre la répartition des ces ROD suivant les sous-détecteurs, ainsi que la taille moyenne du fragment d'événement contenu dans chaque ROD.

Le nombre de ROD attachés par sous-détecteur correspond à la façon dont les détecteurs sont compartimentés (Table 5). Par exemple, si on prend le cas des détecteurs à radiation de transition (TRT), on a deux roues et deux demi-tonneaux. Chaque tonneau est séparé physiquement en 32 zones et chaque roue en 96 zones, ce qui fait un total de  $32+32+96+96 = 256$  zones, d'où les 256 ROD attachés.

| <i>DETECTEUR</i> | <i>NB DE RODs</i> | <i>TAILLE DU<br/>FRAGMENT PAR ROD<br/>(Kbits)</i> |
|------------------|-------------------|---|
| TRT              | 256               | 8   |
| SCT              | 92                | 13  |
| Pixel            | 110               | 15  |
| CALO EM          | 794               | 14  |
| CALO HADRONIQUE  | 64                | 7   |
| MDT              | 192               | 6   |
| CSC              | 8                 | 2   |
| RPC/TGC          | 48                | ~0  |
| TOTAL            | 1564              | ~2 Mo   |

*Table 5: Nombre de ROD et occupation moyenne par sous-détecteur[15]*

A cette liste, il faut ajouter le (ou les) ROD qui vont recevoir des informations du système de triggers de premier niveau.

D'un point de vue des performances, les ROD devront être capables de recevoir et stocker de l'ordre de 135 Mo/s . Cette limite vient en fait essentiellement du lien par où vont transiter ces fragments d'événements. Compte tenu de la taille maximum de 1800 octets que devrait pouvoir recevoir un ROB, la fréquence maximum à laquelle va pouvoir fonctionner le niveau un est de 75 kHz.

Les ROD représentent le dernier étage des éléments de la chaîne d'acquisitions qui soient spécifiques aux détecteurs auxquels ils sont attachés. Au-delà on ne trouvera que des éléments qui sont semblables, quel que soit le sous-détecteur.

A chaque ROD est attaché un autre élément : le « Read Out Buffer » (ROB).

### Les ROB: Read Out Buffer

Chaque ROB est lié, via un lien « Read Out Link » (ROL) à un ROD, ce qui implique que l'on a autant de ROB que de ROD (Figure 17). La différence essentielle provient du fait que les ROB sont tous identiques quel que soit le détecteur [18]. Ils sont l'élément commun de base de la chaîne d'acquisition, ce qui permet de garder une grande flexibilité par rapport aux éventuelles évolutions technologiques susceptibles d'être apportées aux ROD et ROB. Ils sont des éléments « actifs » en ce qu'ils sont capables de distribuer, de collecter, et de stocker les données. Ces fonctions ne sont disponibles que par la présence d'un système de contrôle propre à chaque ROB .

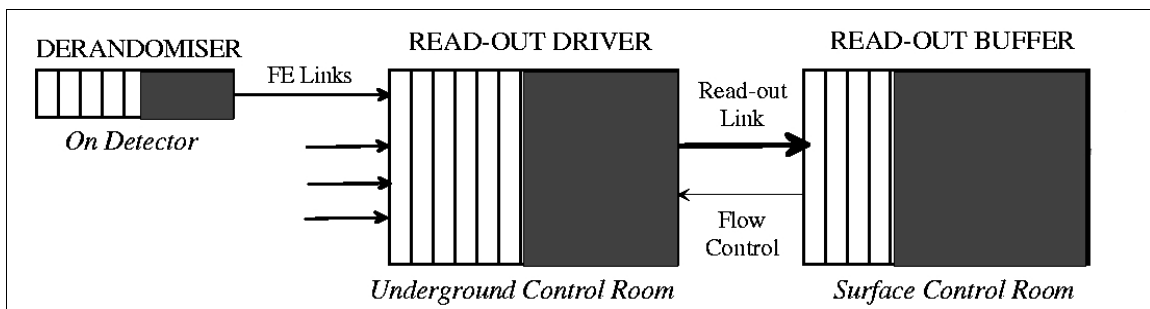


Figure 17 : Liaisons FE, ROD et ROB

### 3.2.4 ACQUISITION DE HAUT NIVEAU

Dans les stades précédents de la description du système d'acquisition d'ATLAS, on a vu que les différents composants ROD et ROB ne contiennent que des fragments d'événements. Il va donc falloir à présent rentrer dans une phase où l'on va collecter tous ces fragments, pour assembler l'événement dans sa globalité .

### L'EB: Event Builder

Le rôle de l'Event Builder est d'assembler les fragments contenus dans les ROB pour créer la donnée événement [19]. Il est en fait composé de cinq parties :

- ✓ Les commutateurs (switchs), qui transportent les fragments.
- ✓ Les sources, qui correspondent aux ROB.
- ✓ La destination, qui sera en fait l'interface d'entrée du prochain élément de la chaîne d'acquisition, à savoir le filtre d'événement.
- ✓ Le DFM: Le Dataflow Manager, qui a en charge la gestion du flot de données. C'est lui qui va assigner à chaque source la destination à laquelle elle doit envoyer son fragment d'événement. Lorsque tous les fragments sont collectés, le DFM indique à la destination que l'événement est complet.
- ✓ Le gestionnaire de réseau (network manager) qui, comme son nom l'indique, s'occupe de gérer le réseau.

A ce niveau de l'acquisition les événements formés auront une taille de l'ordre de 2 Mo et sortiront de l'Event Builder à une fréquence de 1 à 2 kHz (fréquence dictée par la sortie du niveau deux), c'est à dire que la bande passante de sortie sera de l'ordre de 2 à 4 Gb/s. Le système, tel que l'on vient de le décrire jusqu'à présent, c'est-à-dire depuis les détecteurs jusqu'à l'Event Builder, est appelé traditionnellement DAQ (Data Acquisition).

### L'EF: Event Filter ou filtre d'événements

Une fois l'événement complet, il est envoyé dans l'avant dernier étage de la chaîne d'acquisition; il s'agit de l'Event Filter. L'Event Filter est uniquement un niveau supplémentaire de sélection qui doit diminuer par dix le flot de données et n'apporte pas de modification à l'événement, si ce n'est la possibilité de le reformater et d'y ajouter de



nouvelles informations concernant justement les triggers appliqués à ce niveau. Le format de sortie de l'événement après l'Event Filter n'est pas encore fixé et plusieurs solutions peuvent être envisagées. On peut évidemment ne sortir que les informations brutes, issues des détecteurs que l'on appelle RAW data, ou bien reformater l'événement comme une collection d'objets ou bien les deux, cette liste n'étant pas exhaustive.

Enfin en sortie du filtre, l'événement est envoyé aux unités de stockage. A l'heure actuelle encore aucune décision n'a été prise en ce qui concerne ce système. Cela est essentiellement dû à l'évolution rapide des technologies de stockage de l'information. Attendre le dernier moment, c'est se donner la possibilité de choisir une meilleure (et moins chère) technologie. La relation triviale qui existe entre les moyens de stockage et le format des données à stocker explique naturellement que le choix du format de l'événement reste encore en suspens.

### 3.2.5 ACQUISITION GLOBALE

Le cheminement des événements à travers le système d'acquisition d'ATLAS que nous venons de voir, permet de définir les grandes structures qui composent le système de dataflow du détecteur. Reste encore à aborder le dataflow dans sa généralité, c'est-à-dire acquisition et déclenchement liés. Pour cela il est nécessaire de définir un élément primordial du dataflow qu'est le ROS.

#### Le ROS : Read Out System

Le ROS représente un sous-système du dataflow qui est capable de réceptionner les données des détecteurs, de les bufferiser et de les distribuer. D'un point de vue pratique le ROS est constitué d'un ensemble de ROB mais aussi des interfaces et des systèmes de contrôle qui s'y attachent. De façon schématique, on peut définir le ROS comme un ensemble comprenant :

- ✓ Un gestionnaire de données.
- ✓ Des ROBs.

- ✓ Un ensemble d'interfaces vers le LVL2, l'Event Builder,...
- ✓ Un système de contrôle local

Cette notion de ROS permet notamment de voir l'ensemble du dataflow d'ATLAS comme simplement un ensemble de quatre éléments que sont le ROS, l'Event Builder, le dataflow du niveau deux et l'interface avec le filtre d'événement, comme le montre le diagramme ci-dessous (Figure 18).

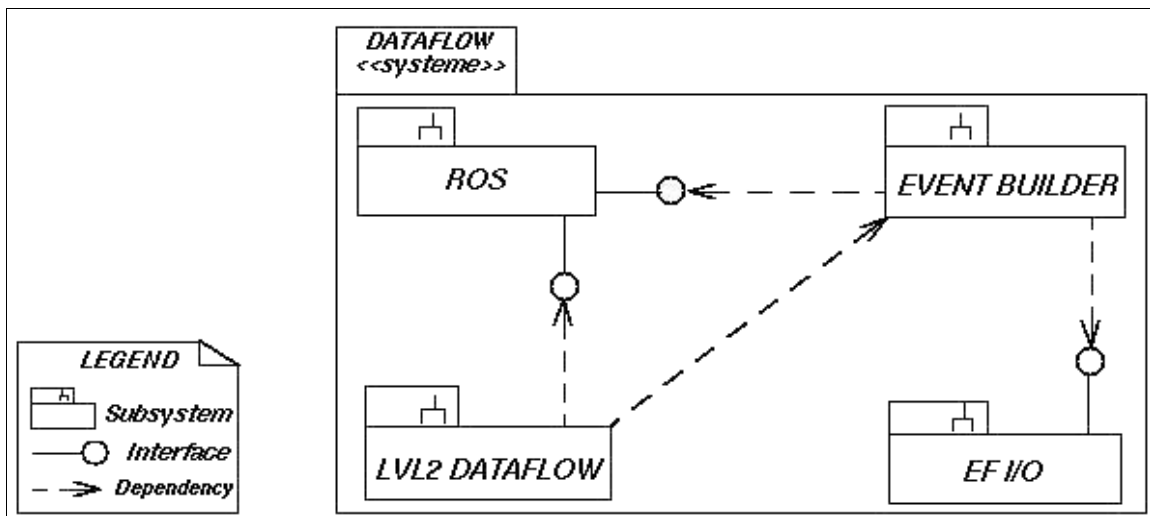


Figure 18 : Diagramme UML du dataflow système

Le ROS s'interface avec le trigger de niveau deux et l'Event Builder afin de leur fournir les données de l'événement. Ce même Event Builder est dépendant du filtre de niveau deux : ce n'est qu'après acceptation de l'événement par le niveau deux que l'Event Builder envoie, via une interface, les données au filtre d'événement.

Déclenchement et acquisition ont recours à de nombreuses fonctions ou services identiques qui sont fournis par un ensemble de logiciels de support appelé Back End Software.

### 3.3 LES LOGICIELS DE SUPPORT (BACK END SOFTWARE)

Les composants qui forment les logiciels de support pour l'acquisition sont pour une part des fonctions auxquelles les systèmes d'acquisition ou les triggers peuvent faire appel, mais aussi des services auxquels ont recours l'ensemble des systèmes (Figure 19) [20]. On trouve notamment un premier ensemble constitué des composants de base que sont:

- ✓ *Run Control* : C'est un composant qui contrôle de façon coordonnée les opérations (commandes) effectuées dans les divers systèmes. C'est notamment à l'aide de ce composant que sera démarré le système d'acquisition.
- ✓ *Configuration Database* : De très nombreux paramètres seront nécessaires à chaque système, notamment pour leur initialisation mais pas seulement. Ce composant sera en charge de la gestion de cette (ou ces) base(s) de données.
- ✓ *Message Reporting System* : Permet le transport, le filtrage et le routage des messages, notamment d'erreurs, susceptibles d'être émis dans le DAQ.
- ✓ *Process Manager* : C'est le composant qui permet le démarrage, l'arrêt et donne l'état (actif ou non) des différents processus qui sont nécessaires aux systèmes.
- ✓ *Information Service* : C'est un service qui se charge de collecter et distribuer des informations générées par les autres composants.

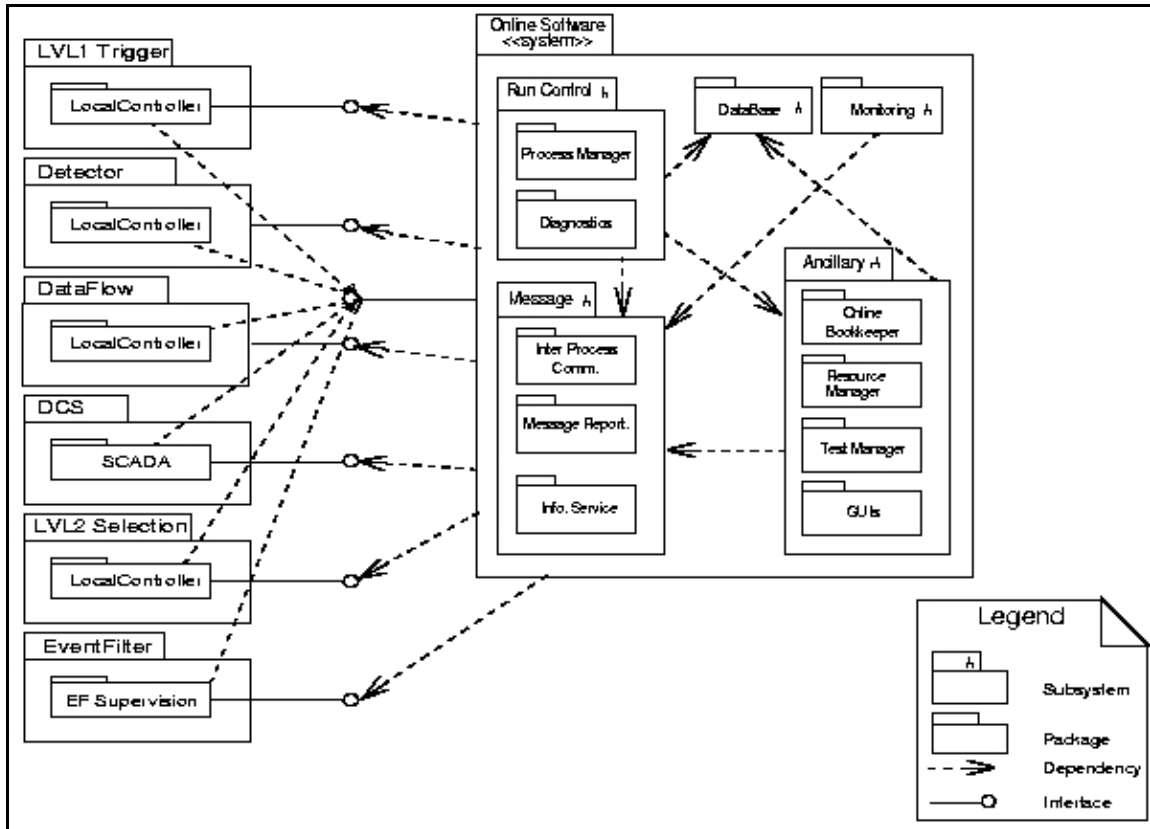


Figure 19: Interaction du Backend Software avec les autres systèmes

D'autres composants dits d'intégration sont également présents dans le Back End Software.

- ✓ *Partition and Resource Manager* : Les ressources matérielles et logicielles présentes sont très disparates. Il est donc nécessaire d'allouer les ressources de façon cohérente et cela afin d'éviter les conflits. C'est le but de ce composant.
- ✓ *Graphical User Interface (GUI)* : Ce composant sera finalement le seul qui sera visible par l'utilisateur du système. C'est cette interface entre l'utilisateur et les différents systèmes qui permettra notamment de contrôler l'acquisition, les triggers, les paramètres du run, les différentes configurations, ...
- ✓ *Bookkeeper* : Son rôle est de conserver l'historique des états des systèmes, comme les divers changements des taux des triggers ou des différents

paramètres.

- ✓ *Event Dump* : C'est un programme muni d'une interface graphique permettant de vérifier l'intégrité de l'événement et sa structure.

Le détecteur ATLAS possède un autre système qui a en charge la gestion des différents systèmes liés au détecteur.

### 3.4 LE DCS : DETECTOR CONTROL SYSTEM

Le DCS est un système complet dont le but est de gérer l'ensemble des paramètres liés au fonctionnement des différents sous-détecteurs : hautes tensions, flux de gaz, température, etc... Il ne se limite pas à un système d'acquisition. Il doit fournir trois types de fonctions:

- ✓ Acquérir de l'information
- ✓ Superviser ces informations
- ✓ Contrôler le détecteur. C'est à son niveau que doivent être signalées les erreurs provenant des différents systèmes d'ATLAS et au besoin réagir en conséquence. Pour l'exemple, on peut citer le contrôle du gaz dans les chambres à muons, ou bien la température au sein des cryostats, enfin tout ce qui concerne le bon fonctionnement des détecteurs.

L'une des tâches essentielles du DCS est la gestion des systèmes qu'il a sous son contrôle. Il doit le cas échéant réguler l'ensemble des dispositifs.

Le DCS qui est attaché à ATLAS répond au nom de SCADA (Supervisory Control And Data Acquisition system). Les informations collectées par le DCS proviennent en partie des Front End des différents composants du détecteur mais aussi d'une pléthore de

détecteurs annexes. Ces informations sont stockées au sein d'une base de données qui est rafraîchie en temps réel.

Les informations sont regroupées en partitions correspondantes généralement aux différents sous-détecteurs, bien que cela ne soit pas une règle absolue. En plus des ces partitions, trois autres partitions sont présentes. Une première, nommée « common services », inclut tout ce qui concerne les alarmes, l'archivage, ... La seconde, dite « common infrastructure », est responsable du contrôle des équipements comme par exemple le refroidissement. Enfin la troisième, nommée « external connection » est responsable des communications avec les systèmes externes. D'un point de vue matériel, le DCS nécessite la présence de très nombreux capteurs, actionneurs de tous types et cela sur l'ensemble du détecteur ATLAS. Certains de ces équipements sont exposés à des très importantes doses de rayonnements.

Les interactions du DCS avec les autres systèmes sont résumées sur la Figure 20 .

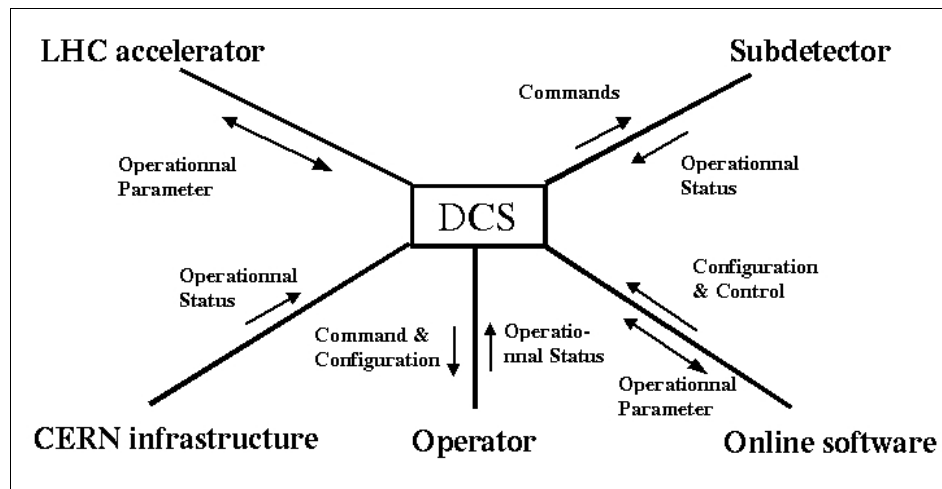


Figure 20 : Interaction du DCS avec les autres systèmes de contrôle

Les communications internes auront lieu essentiellement vers :

- ✓ Les sous-détecteurs
- ✓ Le système de contrôle (Opérateur) et cela grâce à l'utilisation des fonctions apportées par le Back End Software

Les communications vers les systèmes extérieurs se feront essentiellement vers les trois destinations suivantes :

- ✓ Les services techniques du CERN : Ils sont utilisés notamment pour tous ce qui concerne le refroidissement, l'approvisionnement en électricité, en gaz, la gestion de la ventilation, le contrôle des taux de rayonnements dans les différentes zones.....
- ✓ Le LHC: Une relation avec l'accélérateur est absolument nécessaire, ne serait ce que pour connaître l'état du LHC.
- ✓ Le système de contrôle général d'acquisition d'ATLAS.

Le diagramme suivant (Figure 21) présente le contexte dans lequel travaillent les triggers de haut niveau (LVL2 et EF ), le DAQ et le DCS.

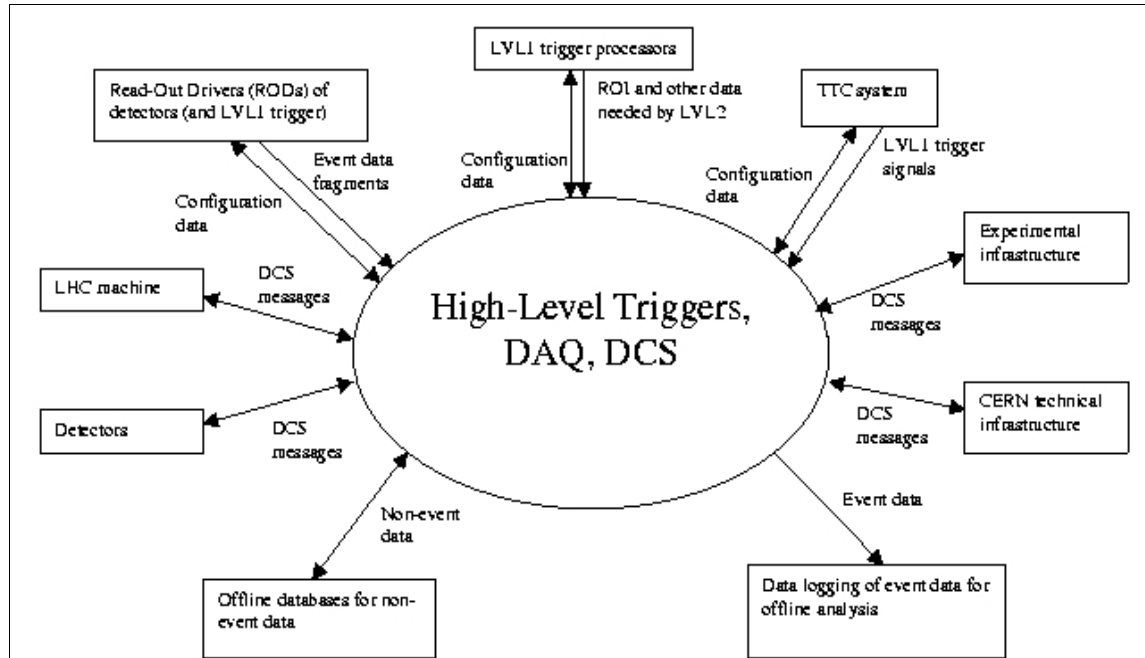


Figure 21 : Interaction entre les Triggers de Haut Niveau et les autres systèmes

### 3.5 RESUME

Dans ce chapitre je décris d'une part le système de déclenchement d'ATLAS et d'autre part le système d'acquisition, qui forment à eux deux le système de Triggers/DAQ de l'expérience. A ces deux entités il faut ajouter le DCS qui est le système qui a en charge le contrôle et la surveillance des différents systèmes du détecteur. Enfin, j'ai brièvement présenté le paquetage des logiciels commun à l'ensemble de ces éléments.





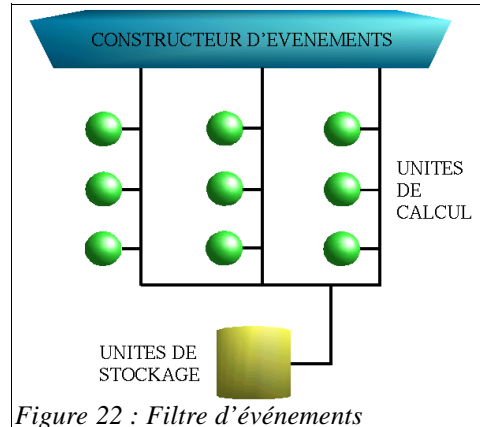
## CHAPITRE 4

# LE FILTRE D'ÉVÉNEMENTS

J'ai décrit dans le chapitre précédent la chaîne d'acquisition et les triggers d'ATLAS, en laissant de côté le troisième niveau de sélection des événements qu'est le filtre d'événements ou Event Filter . Dans ce chapitre nous allons détailler, tant du point de vue fonctionnel que du point de vue architectural, les diverses solutions envisagées pour ce niveau [21].

### 4.1 GENERALITES

Commençons par replacer le filtre d'événements dans son contexte. L'Event Filter se situe entre l'Event Builder et les unités de stockage. La sortie de l'Event Builder est constituée d'une centaine de ports sur lesquels seront connectées un ensemble d'unités de traitement de tâches. C'est cet ensemble d'unités qui constitue l'Event Filter. Chacune des unités de traitement des tâches est définie généralement comme un sous-ensemble du filtre, bien que ces sous-ensembles puissent désigner plusieurs d'unités de traitements. On appelle de tels sous-ensembles des sous-fermes de calcul.



Un tel ensemble d'unités de calcul correspond typiquement à un système de *meta-computing*. Ce calcul qui s'effectue sur des données dont la taille est importante, amène naturellement vers la notion de calcul distribué, à ceci près qu'une contrainte de temps réel est également présente dans notre cas.

#### 4.1.1 FONCTIONALITES ATTENDUES

Le filtre d'événements a trois objectifs:

##### 4.1.1.1 SELECTIONNER LES EVENEMENTS

Les sélections qui seront effectuées au niveau du filtre d'événements auront pour objectif de ne laisser passer que 100 à 200 événements par seconde sur les 1000 à 2000 qui sont acceptés par le niveau deux. Le temps estimé pour la sélection d'un événement est de l'ordre de la seconde. On s'attend à des temps de traitement qui seront de l'ordre de 0,1 seconde, pour les événements rapidement rejetés, à plus de 100 secondes pour les événements qui réclament une analyse plus complète.

Le taux de 200 événements par seconde retenu en sortie de chaîne d'acquisition, provient des estimations qui ont été faites sur la statistique nécessaire pour faire ressortir un signal de découverte, mais aussi des estimations des ressources disponibles qui permet-

tront d'analyser les données.

Par rapport aux deux niveaux précédents l'Event Filter présente la particularité d'effectuer sa sélection sur un événement entier. C'est le seul niveau de trigger où l'on dispose de l'ensemble de l'information concernant un événement. Les critères de sélection vont ainsi pouvoir être appliqués sur l'événement dans sa globalité et non plus uniquement sur des fragments.

#### 4.1.1.2 MONITORAGE DE L'EXPERIENCE

Le fait de disposer de l'événement dans sa globalité apporte la fonctionnalité nécessaire au second objectif de l'Event Filter, qui est de monitorer l'ensemble de la prise de données. En effet, au niveau de l'Event Filter, on pourra avoir des informations sur tout ce qui concerne l'expérience, tant au niveau du détecteur que de la physique (taux des différents triggers). Ce type de monitoring permet d'assurer un contrôle global de l'état de l'expérience.

Un autre type de monitoring également nécessaire à ce niveau est celui qui consiste à surveiller le filtre d'événements lui-même pour vérifier la bonne circulation des informations, l'utilisation correcte des ressources, mais aussi faire remonter au niveau de l'utilisateur tous les dysfonctionnements (erreurs, pannes) du système.

#### 4.1.1.3 FONCTIONS DE CALIBRATION ET D'ALIGNEMENT

L'étude de certains événements bien spécifiques (par exemple  $Z \rightarrow e^-e^+$ ), pour lesquels la physique est bien connue, va également permettre de définir ou simplement de contrôler l'exactitude des diverses calibrations et alignements qui sont utilisées pour effectuer la reconstruction. Cela se révèle d'une grande importance, notamment pour ne pas effectuer des reconstructions d'objets physiques fausses, qui amèneraient à faire des sélections biaisées. Ceci conduirait au résultat inverse de celui qui est attendu par le système de triggers d'ATLAS, à savoir sélectionner un maximum d'événements potentiellement porteurs d'une physique intéressante.

## 4.2 FACTORISATION FONCTIONNELLE

Les unités de traitement de tâches seront constituées de grappes de calculateurs. Cette architecture en unités indépendantes permet une factorisation simple et efficace du filtre. Les calculateurs seront constitués de systèmes classiques. Par système classique, on entend un système qui ne nécessite pas de développement particulier d'un point de vue matériel.

Plusieurs types de calculateurs sont envisagés, depuis de simples PC jusqu'à des systèmes multiprocesseurs, en passant par de simples stations de travail. Dans le cas de simples PC, les unités de traitement de tâches attachées aux différents ports de l'Event Builder constitueraient ce que l'on appelle une ferme de PC. Pour ce qui est des systèmes multiprocesseurs, on peut imaginer n'avoir qu'un système multiprocesseurs attaché à chaque port ou alors une ferme de système multiprocesseurs. Bien sûr la factorisation de ces unités de traitement de tâches rend compatible les deux systèmes, certains ports étant attachés à une ferme de PC, d'autres à une machine multiprocesseur et enfin d'autres à une ferme de machines multiprocesseurs ou de stations.

D'un point de vue fonctionnalité le filtre peut être séparé en deux grandes parties:

- ✓ Une première partie correspondant à l'ensemble des fonctions systèmes du filtre.
- ✓ Une deuxième partie qui correspond aux fonctions de traitement des données.

### 4.2.1 FONCTIONS SYSTEMES

#### 4.2.1.1 GESTION DU FLOT DE DONNEES (DATAFLOW)

Comme nous venons de le voir, le filtre d'événements est constitué de plusieurs unités de calcul. L'une des tâches auxquelles sera confronté le système est de distribuer les événements sortant de l'Event Builder aux différentes unités de calcul. Ce travail est d'une importance capitale, car c'est à ce niveau qu'a lieu la répartition du travail au sein des

unités de calcul. Il est nécessaire de bien répartir la tâche de sélection sur l'ensemble de l'Event Filter. Cette répartition est caractérisée par ce que l'on appelle le *load balancing*.

Le traitement nécessaire à la sélection d'un événement ne pouvant pas être efficacement parallélisé, la solution d'affecter un événement à un processeur s'est imposée pour le filtre d'événement. Cette impossibilité de paralléliser les tâches d'analyses qui vont être déployées sur le filtre provient essentiellement des algorithmes de reconstruction qui devront être appliqués à l'événement et qui présentent un degré de parallélisation quasi-nul. Le dataflow a pour tâche d'amener l'événement à traiter jusqu'au processeur.

Le dataflow est l'ensemble des processus qui participent au passage des événements à travers le filtre, ce qui à ce niveau correspond au passage de blocs de 2 Mo de données. Les méthodes de transport de tels blocs de données sont nombreuses. On peut imaginer que le dataflow utilise des protocoles de transport génériques, c'est à dire ne dépendant quasiment pas de l'architecture hardware du système comme c'est le cas avec du TCP/IP<sup>6</sup> ou CORBA<sup>7</sup>. Mais des méthodes de transport spécifiques à l'architecture peuvent être employées, comme un passage d'information par mémoire partagée pour des systèmes de type SMP<sup>8</sup>.

#### 4.2.1.2 SUPERVISION DES TACHES

Au niveau de l'Event Filter la tâche de supervision peut être découpée en deux parties: une première, dite de contrôle, qui consiste à piloter les divers composants constituant le filtre notamment le dataflow et une seconde, dite de monitoring, qui consiste à surveiller le filtre d'événement.

##### Le contrôle:

La première des tâches qui incombe au contrôle est de pouvoir provoquer le démarrage, l'arrêt des processus qui doivent être actionnés dans le filtre d'événement. A cela s'ajoute également la possibilité de modifier les différents paramètres des éléments qui

6 Transmission Control Protocol/Internet Protocol : Protocole de communication le plus répandue sur les réseaux de type Ethernet.

7 Protocole de communication utilisant des objets communs.

8 SMP: Symetric Multi-Processor

constituent le dataflow.

### Le monitoring du filtre

Le monitoring du filtre est l'ensemble des processus qui participent à la surveillance du filtre d'événement, comme la bonne distribution des événements. Cela passe par la surveillance des processus qui ont justement en charge cette distribution, mais aussi par la mesure des performances en terme de bande passante de cette distribution. C'est également ce monitoring qui doit vérifier la bonne répartition des événements sur l'ensemble des unités de calcul du filtre. Le monitoring doit également nous informer sur l'état des ressources du filtre d'événements, et notamment déceler les signes d'une panne ou d'une erreur. Tous cela nécessite une phase de collecte puis de présentation de ces informations à travers une interface utilisateur.

## **4.2.2 FONCTIONS LIEES AUX TRAITEMENTS DES DONNEES**

### *4.2.2.1 TACHES D'ANALYSES*

Les fonctions de filtrage ont pour objectif de rendre possible la sélection des événements. Au niveau du filtre d'événements la sélection ne pourra s'effectuer qu'après une phase dite de reconstruction qui va créer les objets sur lesquels sera effectuée la sélection.

Le filtre d'événements peut également, de façon non systématique, vérifier les sélections qui ont été faites au niveau deux (voire un) servant ainsi de vérificateur pour ces deux niveaux. Cela est d'autant plus facile que le niveau deux fournit à l'Event Filter ses critères de sélection et que l'événement reçu par l'Event filter contient les informations (ROI et autres) du LVL1 qui avaient été envoyées dans un ROB.

Les tâches d'analyses vont également permettre de réaliser la surveillance des détecteurs, par un monitoring des résultats de la reconstruction d'événements, dits de calibration, dont la physique est parfaitement connue.

C'est également à ce niveau qu'aura lieu le monitoring de la physique qui est en train d'être acquise, permettant ainsi de connaître les principales caractéristiques des événements qui passent dans le filtre. C'est par exemple l'information sur les différents taux de sélection réels par canal physique.

#### 4.2.2.2 MONITORAGE DES APPLICATIONS

Le monitoring correspond à la surveillance des applications qui tourneront au sein du filtre. Ces applications correspondent pour la plupart à des programmes de reconstruction d'objets physiques, et cela à partir des données des détecteurs.

L'intérêt d'effectuer le contrôle de la bonne marche de ces applications au niveau du filtre provient de plusieurs points :

- ✓ Effectuer le contrôle des algorithmes au moment de leur utilisation et non pas après par l'analyse des résultats.
- ✓ S'assurer que l'on effectue les sélections sur des objets qui sont correctement définis.
- ✓ Définir au plus tôt les caractéristiques des événements.
- ✓ S'assurer en temps réel de la fiabilité des algorithmes de sélection.

#### 4.2.2.3 PUISSANCE DE CALCUL

Les estimations actuelles donnent un besoin de calcul de l'ordre de 250 SPECInt95.s pour effectuer la sélection d'un événement. Ceci correspond, lorsqu'on ramène ce chiffre à l'ensemble du filtre, à une valeur de 250 000 à 500 000 SPECInt95.s.

Cette estimation correspond au minimum pour n'exécuter au sein du filtre que la tâche de sélection des événements. Le monitoring des applications nécessitera l'apport d'une puissance de calcul supplémentaire qui sera, quelle que soit la situation, inférieure à celle requise si on effectue ce monitoring en dehors du filtre.



## 4.3 CONTRAINTES TECHNIQUES

Le filtre d'événements est un élément à part entière de la chaîne d'acquisition du détecteur ATLAS. Il est donc soumis aux mêmes règles et contraintes que n'importe lequel des autres éléments, la plus évidente étant bien sûr de pouvoir absorber le flot de données entrantes. Cependant sa position en fin de chaîne, et ses objectifs, lui confèrent d'autres contraintes dont une partie influence directement sa conception. On peut diviser ces contraintes en deux catégories : celles, générales, qui concernent l'ensemble du filtre et celles davantage liées à la physique, ou du moins aux moyens de sélection qui seront mis en oeuvre dans le filtre.

### 4.3.1 CONTRAINTES GENERALES

Les contraintes techniques auxquelles est soumis le filtre sont de plusieurs ordres. Certaines sont d'ordre matériel et portent sur l'architecture matérielle du filtre, tandis que d'autres sont d'ordre logiciel et s'appliquent aux logiciels qui seront implantés dans le système. Cependant dans la majorité des cas, on retrouve les mêmes contraintes sur le matériel et sur la partie logicielle.

La plus évidente des spécificités techniques à laquelle le filtre devra répondre est bien sûr de pouvoir gérer en ligne le flot de données entrantes (1 à 2 Go/s), mais aussi le flot de données sortantes (100 à 200 Mo/s). Pour cela le filtre doit mettre à disposition le potentiel de calcul et de transport nécessaire au filtrage des événements. L'exécution de cette tâche ne peut être menée à bien que si certaines contraintes sont satisfaites.

#### 4.3.1.1 LA ROBUSTESSE

La contrainte technique la plus importante, en dehors de celle de pouvoir absorber le flot de données, est la robustesse. C'est la capacité d'un système à absorber un certain niveau de dysfonctionnement sans pour autant remettre en cause sa fonctionnalité.

L'architecture matérielle et logicielle du filtre devra prendre en compte cette contrainte. De la même façon que les autres éléments de la chaîne d'acquisition, le filtre d

l'événements devra être capable de fonctionner dans la quasi-totalité des circonstances. Une fois l'accélérateur en marche, il n'est pas question de ne pas prendre des données. Dans le cas de l'architecture matérielle on parle de robustesse et dans le cas de l'architecture logicielle on parle plus généralement de fiabilité.

Une panne matérielle ne doit pas couper le flot de données, ce qui implique une architecture possédant un minimum de points et de systèmes critiques.

Du point de vue logiciel, la destruction d'un processus ne doit en aucun cas mener à un blocage des autres, ce qui implique une architecture logicielle très bien segmentée et non inter-dépendante. De la même façon les données ne doivent pas être susceptibles d'engendrer des problèmes. Il peut s'agir en particulier de l'arrivée dans le filtre de données qualifiées de corrompues et qui ne doivent pas amener le système dans un état bloquant.

#### *4.3.1.2 LA FLEXIBILITE*

La seconde des contraintes à laquelle devra se soumettre le filtre est la possibilité de le re-configurer dynamiquement afin d'isoler et compenser des secteurs du système qui seraient inopératoires, et cela à des fins de maintenance. Tous ceci doit s'effectuer sans pour autant remettre en cause le fonctionnement du filtre. Cette caractéristique est connue sous le terme de flexibilité. La flexibilité s'applique à la fois sur la partie matérielle et logicielle du flot de données.

Cette contrainte caractérise le filtre d'événements par rapport au niveau deux qui lui ne pourra pas modifier sa taille dynamiquement.

#### *4.3.1.3 LA SECURITE DES DONNEES*

La sécurité, dans le sens de garantie contre la perte de données, est également une contrainte importante au niveau du filtre d'événement. Les événements arrivant dans le filtre ont déjà franchi deux niveaux de sélection, ce qui leur confère déjà un potentiel intéressant justifiant cette contrainte de sécurité.

D'un point de vue technique, cela implique que les événements qui sont en train d'être

traités dans une unité de calcul ne soient pas perdus si cette unité de calcul tombe en panne.

Du point de vue software c'est quasiment la même chose : des instabilités de processus sont toujours possibles et cela à tous les niveaux, distribution d'événements ou analyse et sélection. Mais en aucun cas l'événement ne doit pas être perdu.

#### **4.3.1.4 L'EVOLUTIBILITE**

Le détecteur ATLAS est prévu pour fonctionner une dizaine d'années. Il est évident que la technologie va évoluer durant ce laps de temps, de même que des parties du système vont tomber en panne, ou simplement subir des périodes de maintenance. Pourtant le filtre devra continuer à être opérationnel et aussi accepter les évolutions que nous dictera la technologie. C'est pourquoi, dès sa conception, il faut garder à l'esprit cette contrainte concernant le filtre qu'est la possibilité de faire évoluer le système.

C'est également au niveau de l'évolutibilité et de la maintenance du système qu'une autre contrainte prend toute son importance : c'est celle qui concerne la simplicité du système. Les évolutions et la maintenance du filtre seront d'autant plus faciles, et donc fréquentes, que le système aura été conçu selon un modèle simple. Cela passe également par la description détaillée à travers une documentation de l'ensemble du filtre.

#### **4.3.2 CONTRAINTES LIEES AU TRAITEMENT DES DONNEES**

Avant de commencer ce chapitre, on doit définir ce qu'est un objet physique, terme couramment employé dans le reste de la rédaction. Les données qui arrivent des détecteurs ne portent pas en elles les informations caractéristiques des particules, comme par exemple leurs impulsions, leurs charges, ... Elles possèdent seulement l'information qui permet de remonter à ces caractéristiques. De ce fait pour « voir » la physique de l'événement, il est nécessaire de reconstruire, à partir des informations des détecteurs, des objets représentant ces caractéristiques, sur lesquels une analyse physique pourra être faite. C'est cette collection d'objets que l'on appelle objets physiques. C'est notamment l'objet "impulsion", l'objet "trace" etc.... Ces objets physiques n'ont pas nécessairement une réalité, même au niveau de l'implémentation d'un code, mais ils définissent les caractéristi-

ques physiques sur lesquels vont porter les analyses et les sélections.

Le filtre d'événements est un système composé d'unités de calcul et de systèmes de communication, sur lesquels portent des contraintes (comme nous venons de le voir). C'est au sein de ces unités de calcul que vont être effectuées les sélections qui vont diminuer par dix le taux d'événements. Ces sélections vont porter sur la physique qui est présente dans l'événement. Le filtre d'événements possède deux points forts pour effectuer sa sélection

- ✓ Il dispose d'un temps relativement grand, par rapport aux deux précédents niveaux de sélection, pour prendre sa décision, ce qui signifie qu'il pourra appliquer ses critères de sélection sur des objets physiques qui nécessitent, pour être créés, un temps de calcul important.
- ✓ Le filtre d'événements est, comme nous l'avons déjà dit, le seul filtre d'ATLAS qui possède l'ensemble de l'information concernant un événement. Les critères de sélection pourront porter sur plusieurs caractéristiques propres à l'événement. On pourra par exemple poser des critères de sélection demandant des corrélations entre des objets physiques différents.

La construction des objets physiques nécessaires à la sélection a une grande influence sur la définition du filtre d'événements, en terme de puissance CPU, mémoires,....

Afin d'éliminer les biais qui pourraient apparaître lors de la reconstruction au sein de l'Event Filter, le filtre d'événements utilisera dans la mesure du possible les mêmes algorithmes, et surtout les mêmes implémentations de code, que ceux qui seront utilisés dans les analyses hors faisceau. Cette stratégie permet de ne pas avoir à développer un deuxième lot de programmes de reconstruction spécifiques au filtre, et permet ainsi de faire évoluer simultanément les programmes de reconstruction, hors et en faisceau.

#### *4.3.2.1 L'ENVIRONNEMENT DE RECONSTRUCTION*

Les applications qui seront implantées dans le filtre d'événements manipulent un certain nombre de données, provenant soit de l'événement, soit de bases de données diverses. Un environnement de travail (framework) commun a été instauré, afin de faciliter la

communication, mais aussi la manipulation des informations entre les différentes applications. C'est notamment au niveau de cet environnement qu'est défini le format des données qui sont manipulées par les différentes applications. Cet environnement est également présent lors des analyses hors faisceau qui sont faites des données et pour lesquelles des programmes de reconstruction sont utilisés.

La totalité des programmes de reconstruction qui sont développés par la collaboration ATLAS sont basés sur un environnement de travail qui répond au nom de ATHENA. Cet environnement apporte aux applications les fonctionnalités communes, et c'est à son niveau qu'est défini l'ordre dans lequel les reconstructions doivent s'effectuer.

#### 4.3.2.2 LA RECONSTRUCTION

La phase qui consiste à créer les objets physiques à partir des données des détecteurs, est appelée la reconstruction.

La reconstruction est une phase qui ne concerne pas uniquement le filtre d'événements. On retrouve cette phase dans l'analyse faite après l'acquisition des données. Elle consiste à reconstruire à partir des données des détecteurs les objets physiques qui vont être analysés. Pour cela des algorithmes développés spécialement sont employés. On trouve des algorithmes de reconstruction de traces qui, à partir des informations données essentiellement par l'ensemble des détecteurs internes, vont déterminer la trajectoire des particules, mais aussi des algorithmes qui vont calculer les dépôts d'énergie, ou déterminer la trajectoire des muons, ou bien encore déterminer la présence et la position de vertex secondaires.

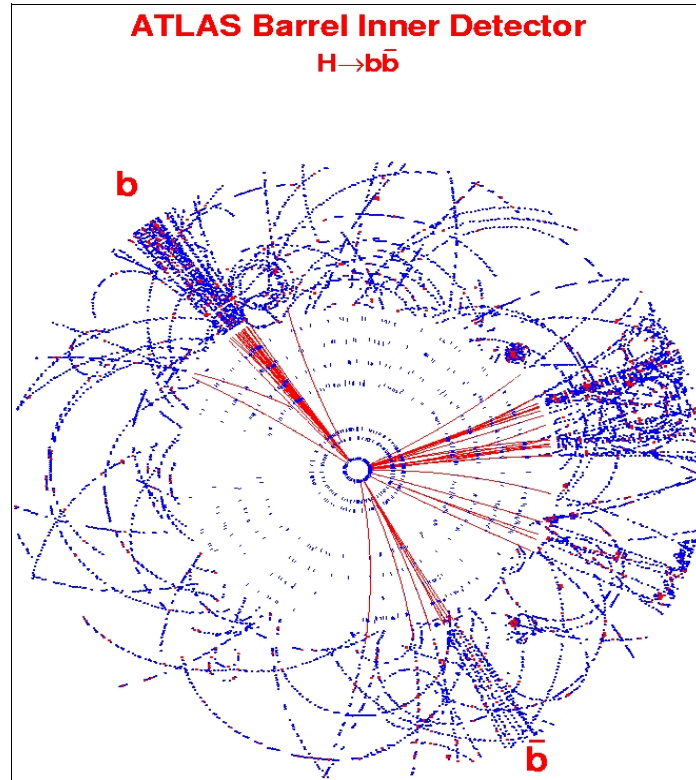


Figure 23 : Trace issue de la reconstruction d'un événement Higgs en  $b\bar{b}$

Ces algorithmes peuvent être relativement simples et ne demander que peu de ressources de calcul comme c'est le cas pour la reconstruction des dépôts d'énergies dans les calorimètres. Ce type d'algorithmes ne présente pas de difficultés algorithmiques particulières, contrairement aux algorithmes de reconstruction de traces qui peuvent se révéler très gourmands en calcul. Typiquement dans ce dernier cas on a affaire à un problème de combinatoire qui peut se résumer à cette tâche : retrouver les traces de particules les plus probables, et cela à partir d'une collection de points de passage. Le dessin (Figure 22) montre le résultat d'une reconstruction de traces. Cette reconstruction correspond à la simulation de la désintégration du boson de Higgs en  $b\bar{b}$ . Cela est d'autant plus difficile que parmi les données issues des détecteurs se trouve une part importante de bruit d'origine diverse. Cela peut être du bruit provenant du rayonnement ambiant ou de l'électronique d'acquisition. Ce bruit se caractérise par la présence de points de passages qui ne correspondent pas à un élément provenant de l'événement.

Parallèlement à ce problème de temps de calcul, vient s'ajouter la notion de quantité de données à traiter pour reconstruire l'événement. Dans ce cas, on se trouve en situation inverse par rapport à précédemment. Les détecteurs de traces ne génèrent finalement que très peu de données correspondant uniquement aux points de passage des particules, alors que les calorimètres mesurent le dépôt d'énergie dans chacune des cellules qui les composent, ce qui correspond au final à la principale contribution des données présentes dans un événement .

Concernant la reconstruction, il faut noter que la plupart des algorithmes nécessitent l'accès à des informations qui sont autres que celles données par le DAQ. C'est par exemple l'accès à la géométrie du détecteur, nécessaire pour connaître les positions des différents détecteurs. C'est aussi l'accès aux fichiers de calibrations. Cela signifie, étant donné que chaque unité de calcul du filtre aura à s'occuper d'un événement, qu'il faut amener ces fichiers de calibrations et de géométrie jusqu'aux unités de calcul.

La relation un processeur, un événement vient de la non parallélisation de la reconstruction de l'événement. En effet en plus de n'être que très peu parallélisable, les programmes de reconstruction sont pour la plupart dépendants séquentiellement, l'exécution d'un algorithme de reconstruction pouvant nécessiter des informations qui ne seront disponibles qu'après l'exécution d'un autre algorithme.

Certains types d'algorithmes de reconstruction peuvent s'avérer très gourmands en temps de calcul, ce qui pose un problème au niveau du filtre d'événements qui doit effectuer la reconstruction et la sélection de l'événement dans un temps de l'ordre de la seconde. C'est pourquoi certaines des reconstructions qui seront effectuées dans l'Event Filter seront moins raffinées que celles effectuées lors de l'analyse, période où la notion de temps de calcul est beaucoup moins cruciale.

Enfin il est nécessaire d'avoir des programmes de reconstruction parfaitement fiables et qui ne sont pas susceptibles de se retrouver dans un état bloquant (du point de vue de la conception du logiciel ou de la consommation des ressources). Cette contrainte doit être parfaitement prise en compte dans la conception des programmes de reconstruction.

### 4.3.2.3 LA SELECTION

Une fois les objets physiques reconstruits, les critères de sélection peuvent être appliqués pour déterminer si un événement est intéressant ou pas. Ces sélections sont simples et correspondent au dépassement d'un seuil pour différents objets physiques reconstruits, les dépassements de seuil pouvant être corrélés entre différents objets physiques de nature différente. Par exemple, la sélection peut réclamer la présence d'un jet identifié comme originaire d'un quark  $b$  de 30 GeV et la présence de deux leptons d'énergie minimale de 50 GeV .

Enfin la possibilité de changer les critères de sélection de façon régulière et rapide, contraint le filtre d'événements à être un système dynamiquement re-configurable du point de vue des sélections qui y seront effectuées. Cette souplesse dans la sélection des événements au niveau du filtre fait partie de ses principales qualités.

## 4.4 LES UNITES DE CALCUL

Les unités de calcul qui constitueront le coeur du filtre d'événements seront des systèmes capables de produire du calcul en grande quantité. Aucun développement n'étant prévu pour ces unités, elles devront être constituées de systèmes de calcul standards disponibles auprès de l'industrie informatique. Deux types d'approches peuvent être envisagés.

Une première tend à factoriser la puissance de calcul en un nombre élevé d'éléments de moyenne puissance. C'est typiquement dans ce cadre qu'apparaît la notion de ferme de calcul. Une ferme de calcul est un réseau composé d'un nombre important de noeuds. A chaque noeud se trouve une petite fraction de la puissance de calcul globale de la ferme. Typiquement ces unités de calcul sont des machines monoprocesseur.

La seconde solution consiste à utiliser des machines de très forte puissance de calcul mais en nombre plus réduit. Dans ce cas là, la solution consiste à utiliser des machines parallèles, la puissance de calcul étant obtenue par la mise en parallèle de plusieurs pro-



cesseurs au sein d'une même machine, partageant ainsi les ressources telles que la mémoire ou les systèmes d'accès. Ce type de machines peut également être mis en réseau sous forme de ferme. Dans ce cas, à puissance égale, le nombre de noeuds de la ferme est inférieur à celui qu'il l'aurait été pour des unités de calcul monoprocesseur.

#### 4.4.1 LES MONO-PROCESSEURS

Les différents types de machines monoprocesseur qui peuvent être mises en réseau sont :

- ✓ Des simples ordinateurs de type personnel, PC, Imac, ... Ces unités de calcul présentent l'avantage d'être produites en très grand nombre et donc d'être disponibles très facilement. Le rapport puissance de calcul / prix est de plus en plus intéressant pour ce type de calculateur. La principale caractéristique de ce type de calculateur en fait, est à la fois un avantage et un inconvénient : ils sont standards, ce qui assure une forme de compatibilité et une maintenance facilitée mais qui d'un autre côté peut limiter fortement l'adéquation entre cahier des charges et performances attendues.
- ✓ Des stations de travail, qui présentent l'avantage d'être un peu plus spécifiques (systèmes d'exploitation, ...) que de simples ordinateurs "familiaux". Les processeurs équipant ce type d'ordinateurs sont généralement plus puissants en termes de calcul. Il en résulte une diminution de la taille de la ferme de calcul.
- ✓ Enfin à titre informatif, il faut mentionner une autre famille de "calculateurs" susceptibles de fournir une puissance de calcul. Il s'agit de l'ensemble des systèmes embarquant un processeur et qui remplissent de plus en plus notre vie quotidienne. C'est le cas notamment des diverses consoles de jeux ou portables de type WAP<sup>9</sup>, qui possèdent des processeurs susceptibles de fournir une puissance de calcul non négligeable. Ce type de calculateurs présente certes une puissance de calcul intéressante mais le reste de leur profil ne correspond pas aux demandes formulées pour le filtre (quasiment pas utilisable en dehors de leur

<sup>9</sup> Désigne initialement un protocole de communication implementé sur de nombreux appareils de téléphonie mobile. Le nom WAP désigne de plus en plus l'appareil lui-même.

fonctionnement de base, évolution impossible à prévoir, ...).

#### 4.4.2 LES SYTEMES MULTI-PROCESSEURS

Cette classe de machine est connue sous le nom de SMP (Symetric Multi-Processor). Il existe deux familles de systèmes qui mettent en jeu des processeurs fonctionnant en parallèle. Elles se différencient par la méthode par laquelle elles accèdent à la zone mémoire.

- ✓ Le type NUMA (Non Uniform Memory Access) est un type de machine où à chaque processeur est attaché une mémoire bien précise. Seule une partie de la mémoire est partagée avec le reste des processeurs. L'accès à la mémoire propre des processeurs est généralement très performant par rapport à l'accès à la mémoire partagée, qui s'avère plus lent.
- ✓ Le type UMA (Uniform Memory Access) où tous les processeurs de la machine ont la même priorité pour accéder à la mémoire de la machine. L'accès dans ce type de machine se fait généralement via un commutateur d'architecture "cross-bar<sup>10</sup>".

### 4.5 CAHIER DES CHARGES ET PROTOTYPAGE

#### 4.5.1 CAHIER DES CHARGES

Les diverses considérations établies dans ce chapitre nous ont permis de définir les principales structures, ainsi que les principales caractéristiques auxquelles devra se soumettre la conception du filtre.

---

<sup>10</sup> Commutateurs permettant de mettre en relation processeurs et mémoire.

- ✓ La présence d'une structure permettant de distribuer les événements aux tâches de reconstruction et de sélection.
- ✓ La présence d'une structure s'occupant du traitement des données (reconstruction et sélection).
- ✓ La présence d'une structure collectant les événements sélectionnés et les envoyant vers les unités de stockage.
- ✓ Présence d'un système de surveillance du flot des données.
- ✓ Présence d'un système de contrôle du flot des données.
- ✓ Présence d'un système de monitoring des applications tournant au sein du filtre.
- ✓ Indépendance vis à vis des protocoles de communication.
- ✓ Indépendance vis à vis des technologies employées.

#### **4.5.2 PROTOTYPES**

Le cahier des charges étant ainsi défini, se pose la question de savoir quels peuvent être les systèmes qui y répondent et même s'il existe des systèmes qui y répondent, ou s'il faut les développer. Les conditions techniques (volume des données, utilisation en temps réel, robustesse, ...) représentent un ensemble de contraintes d'un niveau bien supérieur à tous les systèmes qui peuvent déjà exister en physique des hautes énergies. Les avancées technologiques ne pourront que partiellement absorber cette différence et un travail d'investigation s'avère nécessaire pour élaborer des systèmes qui pourront répondre aux exigences du Filtre d'événements.

Les systèmes requis en physique des hautes énergies regroupent des caractéristiques auxquelles peu de systèmes industriels ou provenant d'autres domaines de recherche sont confrontés. Il est donc difficile de trouver un exemple extérieur de système qui présenterait un cahier des charges similaire. C'est pourquoi une phase de prototypage est nécessaire pouvant porter à la fois sur la partie matérielle du filtre mais aussi sur l'ensemble des systèmes qui pourront y être implementés. Le choix de développer une nouvelle technologie étant exclu pour des raisons de coût et de temps, nous nous sommes tournés vers l'utilisation de technologies déjà existantes.

Les différents choix technologiques possibles ont amené la collaboration ATLAS à étudier trois prototypes de filtres. Ces prototypes diffèrent notamment par le type de calculateur qui est employé pour effectuer la tâche de sélection, mais aussi par le protocole de communication utilisé. Cette phase de prototypage a pour objectif de définir et comprendre tous les tenants et aboutissants qu'implique le choix d'une technologie.

#### 4.5.2.1 PROTOTYPE PC

Je développerai ce prototype en détail dans le prochain chapitre; c'est pourquoi je n'en dirai que quelques mots. Le concept de base de ce prototype est d'utiliser des ordinateurs PC courants, tels qu'on peut les trouver chez n'importe quel revendeur. Ces PC seront reliés entre eux par un réseau de type Ethernet, formant ainsi une ferme de PC. Le prototype de ce filtre d'événements a été développé au Centre de Physique des Particules de Marseille en collaboration avec l'IFAE<sup>11</sup> de Barcelone.

Un second prototype basé également sur des PC a été étudié à l'Université d'Alberta au Canada. Dans le cas de ce prototype les PC utilisés présentent la particularité de posséder des cartes mères susceptibles d'accueillir quatre processeurs de type iX86, l'idée étant de constituer une ferme de machines multi-processeurs. Ici on ne parle pas de machine possédant plusieurs dizaines de processeurs mais plutôt d'une ferme PC "multiprocesseurs".

#### LES AVANTAGES

---

<sup>11</sup> Institut de Fisica d'Altes Energies : Institut de physique des hautes énergies

Les avantages d'une telle configuration sont nombreux. Tout d'abord le prix de revient du système, les PC étant maintenant des ordinateurs possédant des puissances de calcul intéressantes pour un prix des plus compétitif. Le second avantage est la flexibilité d'une telle architecture, augmenter la puissance de calcul du filtre revient à augmenter la taille de la ferme.

### LES INCONVENIENTS

Le fait d'utiliser un système standard, présente l'inconvénient d'être pas nécessairement optimisé pour nos besoins. De plus, d'un point de vue matériel, des simples PC, même de qualité, demandent une maintenance bien supérieure à un système basé sur une architecture plus propriétaire ou du moins plus spécifique.

A cela il faut également ajouter les problèmes d'administration que génère la gestion de plusieurs milliers de PC. En particulier, dans un environnement d'utilisation de type Event Filter, il est absolument nécessaire que tout les noeuds de la ferme, ici les PC, soient, d'un point de vue "système", identiques (synchronisation des paquetages de reconstruction, des bases de données, des systèmes,...) afin de ne pas biaiser la sélection.

#### 4.5.2.2 PROTOTYPE MULTIPROCESSEUR

Ce prototype a été développé à l'Université de Pavie (Italie) . L'intérêt de ce prototype est de profiter des caractéristiques de son architecture pour améliorer les performances du filtre, voire pour simplifier sa conception. L'architecture logicielle qui a été implémentée sur ce prototype est la même que celle qui est présente sur le prototype PC. Seul le protocole de communication diffère, afin de tirer pleinement partie des avantages du SMP. Des tests ont été réalisés sur des configurations ayant de 8 à 20 processeurs [22].

### LES AVANTAGES

L'avantage essentiel d'un système de type SMP est de pouvoir profiter du système de mémoire partagée pour "transférer" les données. Dans le cas d'un prototype multiprocesseur, les processus peuvent ainsi accéder à la même zone mémoire, ce qui pour un système classique correspondrait à une copie physique d'information. Le second avantage

d'un système SMP est de fournir de façon plus compacte une puissance de calcul importante. Une simple machine équipée de huit processeurs équivaldrait à 8 PC avec l'avantage de ne pas nécessiter la présence d'un réseau. De plus, contrairement à une ferme de PC qui nécessite que chaque PC accède aux diverses bases de données nécessaires à la reconstruction, le système multiprocesseur ne nécessiterait qu'un seul accès (une seule copie en mémoire de l'information), qui serait partagé par l'ensemble des processeurs présents sur la machine. On peut ainsi envisager de n'attacher à chaque port de sortie de l'Event Builder qu'un seul ordinateur SMP.

Le nombre de machines étant moins important dans le cadre d'un Event Filter de type SMP, il est aussi plus facile de s'assurer de l'homogénéité de l'ensemble.

### LES INCONVENIENTS

Un des inconvénients des systèmes SMP réside dans le fait qu'à puissance égale ils sont plus chers que des simples PC. De plus, pour gérer de tels systèmes, il est nécessaire d'utiliser des systèmes propriétaires, notamment pour le système d'exploitation. Un autre point faible de ce type de solution vient du fait que la flexibilité de l'ensemble du filtre s'en trouve amoindrie, et que même si les problèmes matériels sont moins fréquents, ils sont en contre partie plus handicapants pour le système, car affectant une part du filtre plus conséquente.

## **4.6 RESUME**

Dans ce chapitre, j'ai décrit dans un premier temps les fonctionnalités qui doivent être remplies par le filtre d'événements, tant du point de vue de la physique, c'est-à-dire la sélection des événements intéressants, que de la gestion du système. J'ai également mis en évidence les différentes contraintes logicielles et matérielles auxquelles est soumise la conception du filtre et notamment celles qui portent sur la qualité du code de reconstruction. Ces considérations ont abouti à l'élaboration d'un cahier des charges comprenant un volet concernant la conception générale du filtre, et un autre volet s'intéressant aux différentes solutions matérielles possibles. L'aboutissement de ce chapitre est la présentation des prototypes de filtre d'événement développés pour

102

ATLAS.

## CHAPITRE 5

### LE PROTOTYPE PC

Dans ce chapitre, je vais présenter en détail les caractéristiques et les choix techniques qui ont été réalisés autour du prototype PC de filtre d'événements. Cela concerne à la fois le flot des données et le monitoring. Ces deux parties seront détaillées, notamment en présentant le design du flot des données qui sort du contexte du prototype PC, car celui-ci est indépendant de l'architecture du filtre. Les choix faits ont été pour la plupart validés par une série de tests à grande échelle qui seront détaillés dans un dernier sous-chapitre.

#### 5.1 FINALITE DU PROTOTYPE

La conception de ce prototype a pour objectif, tout comme les deux autres prototypes, d'évaluer les conséquences d'un choix technologique qui dans notre cas correspond à l'utilisation d'une ferme de PC reliés par un réseau de type Ethernet.

L'une des justifications de l'élaboration de ce prototype, est la validation du concept qui consiste à mettre en oeuvre au sein du filtre uniquement des technologies dites grand public, à la fois par l'utilisation d'unités de calcul fabriquées à grande échelle et à faible coût (ordinateur PC), liées entre elles par une technologie conventionnelle basée sur un simple réseau Ethernet.



Les évaluations de ce prototype portent à la fois sur la réponse qu'il apporte au cahier des charges défini pour le filtre, mais aussi sur la gestion d'un filtre d'événements basé sur le concept de ferme de PC. Enfin à travers ce prototype on va chercher à mettre en avant diverses technologies et à évaluer leur intérêt dans un tel système.

## 5.2 LE FLOT DES DONNEES

Dans le chapitre précédent j'ai énuméré les différentes contraintes qui portent sur le filtre d'événements. Une partie de ces contraintes peut être satisfaite uniquement par la façon dont les événements vont être distribués aux unités de calcul. C'est donc le design du flot des données qui va pour une part importante répondre aux exigences qui portent sur le filtre .

### 5.2.1 LE DESIGN

Le design général du flot des données doit répondre aux problèmes de :

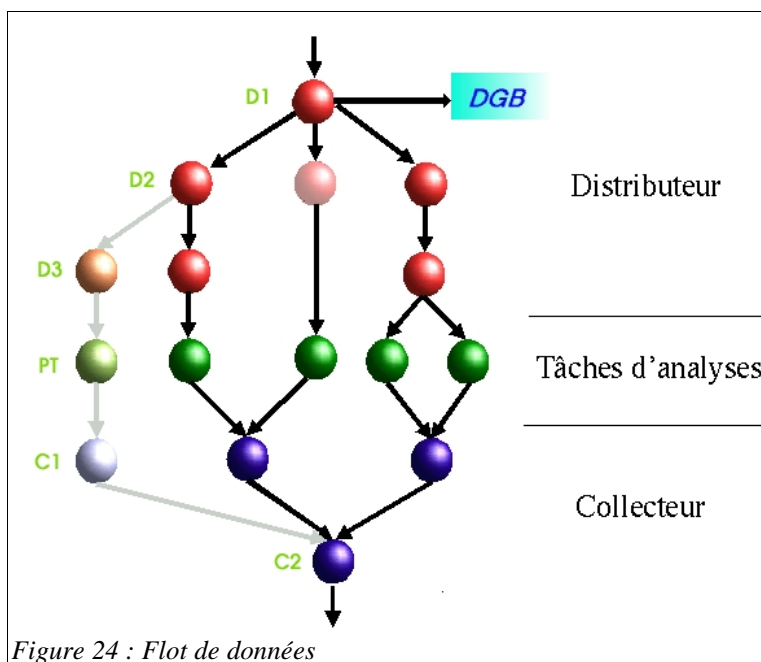
- ✓ Flexibilité.
- ✓ Absorption du flot de données.
- ✓ Sécurité des données.

#### 5.2.1.1 LES MODELES DE DISTRIBUTION

La distribution des événements aux unités de calcul peut être réalisée essentiellement de deux façons.

Dans un premier cas, le système de distribution des événements est chapeauté par un élément qui a en charge la sécurité, l'homogénéité de la distribution, etc... : c'est le gestionnaire du flot de données. Une autre solution consiste à concevoir le système de distribution de telle façon que la sécurité et la bonne distribution des événements soit naturelle. Cette solution, dite *event driven*, présente l'avantage de ne pas nécessiter la présence d'un gestionnaire de flot de données, dont les algorithmes sont souvent très compliqués. C'est sur cette voie que nous nous sommes orientés.

L'architecture du filtre est développée sur la notion de composants, ce qui permet de garder une grande flexibilité au système. Les composants représentent un sous-ensemble d'éléments du flot des données du filtre d'événements. Chacun de ces éléments est indépendant et exécute une tâche précise [23].



Cette conception modulaire du système de transport des événements apporte une souplesse certaine au filtre. Il suffit d'augmenter le nombre de composant pour agrandir la taille du système. D'autre part cette notion de composants indépendants les uns des au-

tres permet également d'apporter une réponse au besoin de robustesse du flot des données, les composants étant remplaçables et indépendants (seul le passage de l'événement les relie entre eux).

L'architecture retenue pour le flot des données découpe le filtre en trois parties : le distributeur, les tâches de traitement (Processing Task) et le collecteur.

Les événements arrivent dans l'Event Filter à travers la SFI (Sub Farm Input), qui est la porte d'entrée dans le filtre d'événement. La SFI est attachée de façon univoque à un port de sortie de l'Event Builder. De façon analogue, on a en sortie la SFO (Sub Farm Output) qui fait la liaison avec les unités de stockage.

Un événement qui arrive dans le filtre voit devant lui tout d'abord le distributeur qui va, dans le composant D1 le distribuer suivant son type (contenu dans le bloc d'en-tête) vers les composants D2, après l'avoir sauvegardé dans un dispositif que nous avons nommé DGB (Distributor Global Buffer). On peut ainsi aiguiller sur des canaux différents les événements de calibration et les événements de physique. A partir de D2, les événements vont être tirés par les tâches de traitement, en utilisant éventuellement des tampons intermédiaires (composants D3).

Les tâches de traitement représentent le coeur du filtre. C'est en particulier là que va se faire la sélection, qui est le premier objectif du filtre. Dans un premier temps il va y avoir reconstruction des objets physiques nécessaires à la sélection, puis sélection de l'événement. Les événements retenus sont envoyés dans le dernier élément du filtre qui est le collecteur. Les événements rejetés sont éliminés du DGB.

Le rôle du collecteur est complémentaire de celui du distributeur. Il va regrouper les événements qui ont franchi la sélection, avant de les envoyer vers l'unité de sauvegarde. En fin de collecteur les événements sont envoyées dans les unités de stockage et éliminés du DGB.

Le rôle du DGB est un rôle de sécurité. Si pour diverses raisons (problèmes logiciels ou matériels) un événement est perdu dans le filtre, on possède toujours une copie de celui-ci dans le DGB. Il suffit de connaître l'état du DGB pour savoir si des événements n'ont pas correctement passé le filtre. La gestion de ces événements rentre dans le cadre plus général de la gestion des erreurs dans le filtre d'événements. On s'assure ainsi de la non perte d'événement dans le filtre d'événements quel que soit le problème rencontré. Le DGB se doit de répondre à un certain nombre de critères qui sont :

- ✓ Avoir une grande robustesse.
- ✓ Accepter un flot de données conséquent.
- ✓ Être d'accès aléatoire, les événements n'étant pas nécessairement effacés dans le même ordre qu'ils ont été stockés.
- ✓ Être accessible rapidement en écriture.
- ✓ Pouvoir rapidement effacer un événement.
- ✓ Avoir une taille suffisante.

## 5.2.2 LES COMPOSANTS

Le design général du flot des données doit répondre aux contraintes suivantes :

- ✓ Liberté au niveau des protocoles de communication et cela à tous les niveaux (flot des données, monitoring, autres).

- ✓ Répartition homogène des événements sur l'ensemble du filtre.
- ✓ Gérer la disponibilité des tâches d'analyses.

Afin de satisfaire au mieux cela, le flot de données a été organisé en composants modulaires reliés par une relation de type client–serveur. Ils sont composés de quatre types d'éléments comme on peut le voir sur le dessin suivant (Figure 25).

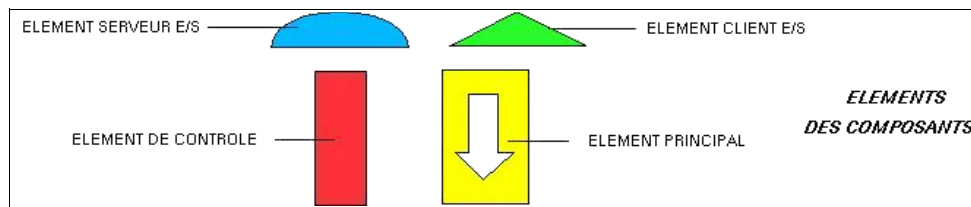


Figure 25 : Éléments des composants du flot des données

- ✓ Un élément d'entrée/sortie client ou serveur
- ✓ Un élément de contrôle : c'est lui qui fait l'interface entre le composant du flot des données et le superviseur.
- ✓ Un élément principal : ce peut être une simple FIFO<sup>12</sup>, dans le cas des composants de la partie distributeur et de la partie collecteur. Dans le cas des tâches d'analyses, cet élément principal implémente les programmes de reconstruction et de sélection.

C'est l'assemblage des ces éléments qui va nous permettre de créer les composants du flot des données. Le schéma (Figure 26) présente les diverses combinaisons possibles.

<sup>12</sup> FIFO : First In First Out , il s'agit d'une simple file où les éléments rentrent et sortent dans le même ordre.

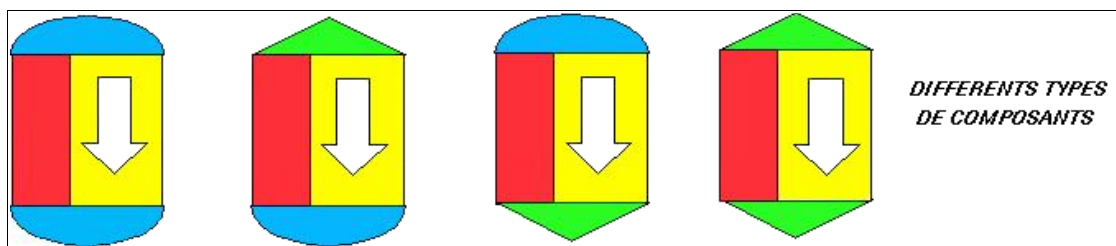


Figure 26: Différents composants

Des composants de type serveur–serveur, client–serveur, serveur–client ou client–client peuvent être ainsi construits. Cette fragmentation des composants en éléments distincts a été réalisée en séparant distinctement les éléments d’entrée–sortie, et cela afin de bien séparer la partie communication du reste du composant. L’implémentation d’un nouveau protocole de communication ne nécessite alors que l’implémentation de nouveaux éléments d’entrée–sortie.

Cette architecture de composants éclatés en éléments distincts présente également l’avantage d’être compatible avec une description objet du code.

La communication entre les différents éléments peut être réalisée selon deux schémas présentés sur la Figure 27.

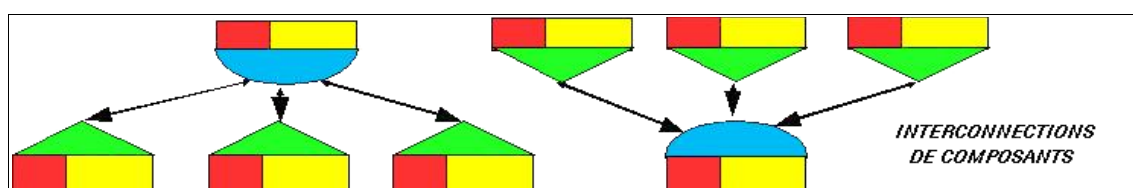


Figure 27: Communication inter-composant

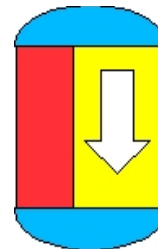
Cette communication peut avoir lieu d’un composant serveur vers plusieurs composants clients, ou de plusieurs clients vers un composant serveur. En entrée, un composant client va chercher les événements, alors qu’un serveur attend les événements qui lui sont envoyés. De même en sortie, un composant client envoie les événements alors qu’un serveur attend une requête pour envoyer l’événement.

Dès leur démarrage, les composants s'identifient auprès d'un service de noms ("naming service", NS). Cette identification permet au NS de connaître le nom, la localisation et le protocole de communication utilisé par chaque composant. Ce même service de noms répond aux requêtes des composants clients concernant la localisation et le protocole de communication utilisé par les composants avec lesquels ils doivent se mettre en relation.

Il est aussi possible d'utiliser des protocoles de communication différents en entrée et en sortie dans un même composant. Cela est possible car l'information sur le type de communication est présente dans le naming service pour chaque composant. Ainsi le composant devant se connecter sur un autre "sait" quel protocole utiliser.

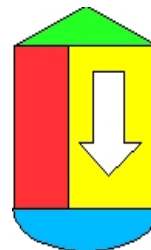
Les différents composants présentés sur la Figure 26 sont organisés de la façon suivante :

Le composant D1 est le point d'entrée du système, il est aussi, avec le composant C2, un point critique du flot des données pour la sous-ferme considérée. Ce composant est situé juste derrière le port de sortie de l'Event Builder qui envoie les événements à la vitesse dictée par le niveau deux, autrement dit le point d'entrée du composant D1 sera de type serveur. D1 distribue sur des composants D2 les événements en fonction du type détecté. Il sera donc également serveur en sortie et son élément principal consistera en une FIFO, avec éventuellement la possibilité d'effectuer un tri.

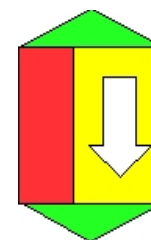


Les composants D2 et D3 sont les composants qui se trouvent juste avant les composants de traitement. Ce sont eux qui vont distribuer les événements sur les tâches d'analyses auxquelles ils doivent servir de "réservoir"

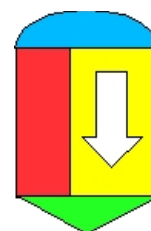
d'événements. Ce sont donc des FIFO. Ils doivent être susceptibles d'alimenter les éléments PT mais aussi de ne prendre de nouveaux éléments que s'il reste de la place au sein de leur FIFO. C'est pourquoi ces éléments sont serveurs en sortie et clients en entrées.



Les composants en charge de la partie analyse et sélection de l'événement (PT) seront ceux qui seront les plus exigeants en terme d'utilisation des ressources, notamment CPU, leur élément principal étant l'ensemble des programmes de reconstruction et de sélection d'événements. Ces composants ne pourront qu'être clients en entrée, afin de ne réclamer un nouvel événement que lorsque le précédent aura été analysé et éventuellement sélectionné. Ils ne pourront également qu'être clients en sortie, l'événement ne pouvant être rejeté ou accepté et envoyé vers le composant collecteur qu'après que l'analyse soit finie.



Les composants C1 sont les pendants des composants D3. Ils sont là pour absorber les événements acceptés par le filtre et sortant des composants PT. Ils seront donc nécessairement serveurs en entrée et clients en sortie. Leur élément principal reste ici aussi une simple FIFO.



Enfin le composant C2 est l'analogue du composant D1.

Il est important de noter que les éléments D3 et C1 ne sont pas absolument indispensables. On peut très bien connecter directement les composants PT sur des composants D2 et C2. De la même façon, on peut introduire plusieurs étages intermédiaires de composants distributeurs et collecteurs dans le système, permettant de diminuer le nombre de connections affectées à chaque composant. De façon plus générale, les composants que



nous venons de décrire peuvent très bien s'articuler selon de nombreux modèles, permettant ainsi de définir différentes structures de flots de données.

La composition du flot de données du filtre sous forme de modules eux-mêmes constitués d'éléments distincts ayant une fonction propre, permet de définir deux niveaux de modularité, ajoutant ainsi de la souplesse au système. Cette souplesse est particulièrement mise en évidence dans le choix des protocoles de communications employés.

### **5.2.3 ADEQUATION DES SOLUTIONS APPORTEES**

Le design de haut niveau du flot de données qui vient d'être présenté, ainsi que celui des composants qui le constitue, permet de répondre à une partie du cahier des charges du filtre d'événements.

Le design de haut niveau répond ainsi au besoin de flexibilité du filtre par l'utilisation de composants modulaires indépendants qui, tels les éléments d'un jeu de construction, permettent de construire dynamiquement l'arborescence du flot de données. Ce même design modulaire permet également de configurer la taille de cette architecture de façon simple et ainsi de pouvoir adapter la capacité du système de flot de données selon les besoins, c'est-à-dire selon le flot de données envoyé par le constructeur d'événements.

La factorisation de travail de filtrage à travers plusieurs branches indépendantes du flot de données répond en partie au besoin de sécurité et de robustesse en limitant l'incidence que pourrait avoir la perte d'un composant. La connexion de plusieurs fermes de filtrage à différents ports de sortie du constructeur d'événements répond également à ce besoin. La sécurité de données passe par la définition et l'étude d'un système de récupération d'erreurs, qui reste encore à évaluer.

Le design des composants tel qu'il est décrit permet, par l'utilisation de relations de type client serveur, de mettre en place un système automatique de répartition des événements sur les différentes unités de calcul. Cette distribution homogène des données est

effectuée naturellement par les tâches d'analyses qui sont les composants qui sont demandeurs d'information (client) vis à vis des composants distributeurs, mais aussi fournisseurs vis-à-vis des collecteurs. Ainsi la gestion de la disponibilité des ressources de calcul (des tâches d'analyse) se fait de façons automatique.

Enfin la modularité des composants permet de laisser libre le choix des protocoles de communication qui peuvent être implantés. D'une façon plus générale, le design de haut niveau et celui des composants est indépendant de toute architecture matérielle.

## 5.2.4 LES COMMUNICATIONS

### 5.2.4.1 LE MATERIEL ET COUCHE DE BAS NIVEAU

La segmentation des entrées-sorties au sein des composants, grâce à la présence d'éléments de communication, permet d'envisager quasiment tous les types de protocoles pour transférer les événements de composants en composants. Nous allons passer en revue les plus communs de ceux-ci. D'un point de vue purement matériel, il existe essentiellement trois types de moyens de communication:

- ✓ Les réseaux filaires ou optiques, auxquels sont associés des routeurs et commutateurs.
- ✓ Des commutateurs de type crossbar, qui sont des commutateurs simples et très performants. Ils sont souvent présents dans les architectures SMP. Les versions les plus sophistiquées de ces crossbars permettent de rendre quasiment homogènes les temps de communication. Adresser la mémoire attachée à un processeur ou adresser de la mémoire distante via un tel crossbar est quasiment la même chose en termes de performance.
- ✓ Les bus. Cette technique consiste à relier les processeurs entre eux à travers un bus qui peut être spécifique.

Les technologies par crossbar ou bus ne nécessitent pas l'utilisation de protocoles à proprement parler pour être utilisés. C'est essentiellement par passage de pointeur ou de référence que va se faire le passage d'informations entre applications. Il n'est pas impossible d'implanter un protocole de communication de haut niveau sur ce type d'architecture. Mais dans ce cas on n'utilisera pas au mieux les spécificités et les performances du système.

Sur les technologies filaires et optiques, plusieurs couches de protocoles sont appliquées. Les premières couches définissent les propriétés génériques et caractérisent ainsi des réseaux particuliers:

#### ✓ Réseau de type Ethernet.

Ce système de communication par réseau est le plus répandu de tous ceux qui existent. Les cartes d'accès à de tels réseaux existent sur la totalité des systèmes informatiques et généralement sont liées au reste de la machine par un bus de communication de type PCI<sup>13</sup>. Les cartes réseaux Ethernet présentent de nos jours des capacités intéressantes, le Gigabit par seconde étant dorénavant commun. Ce type de réseau fonctionne par échange de paquets d'information de taille relativement importante et possède une qualité de service peu élevée .

#### ✓ Réseaux non Ethernet

D'autres réseaux différents de l'Ethernet existent. On trouve notamment des réseaux qui prennent plus en compte la notion de qualité de service. On trouve des réseaux de type Ethernet évolués tel que MyriNet. Les protocoles appliqués sur ces types de réseaux sont généralement des protocoles propriétaires utilisant au mieux les spécificités du réseau.

---

<sup>13</sup> Peripheral Component Interconnect : Bus d'interconnexion de périphérique, c'est la norme la plus répandue actuellement

### 5.2.4.2 LES PROTOCOLES DE HAUT NIVEAU

De nombreux protocoles de communication de plus haut niveau peuvent être déployés sur un réseau de type Ethernet, le plus répandu étant TCP/IP<sup>14</sup>. Il en existe d'autres tels que IPX ou NetBUI ou bien d'autres développés par différents constructeurs.

Il faut également citer des protocoles de communication bien connus comme MPI<sup>15</sup> ou PVM<sup>16</sup>. Ce ne sont pas à proprement parler des protocoles de communication mais plutôt des bibliothèques qui uniformisent les communications quel que soit le protocole situé en amont (TCP/IP, CORBA ou autres). Chaque protocole de communication ayant sa propre implémentation de MPI ou de PVM, le développeur n'a pas à se soucier de la nature de la communication. Il utilisera simplement les fonctions offertes par la librairie correspondante. On trouve ainsi des versions de MPI adaptées pour la communication inter-processus par mémoire partagée.

### 5.2.5 LES TECHNOLOGIES RETENUES

Que cela soit au niveau du design général du flot des données ou des composants, tous les choix technologiques sont possibles. La partie de la collaboration ATLAS qui s'occupe de l'acquisition et du déclenchement a lancé l'étude de trois types de prototypes de filtres d'événements. Les trois prototypes se différencient par leur architecture matérielle mais partagent le même design.

- ✓ Prototype PC:

---

14 Transmission Control Protocol/Internet Protocol : Protocole de communication le plus répandu sur les réseaux de type Ethernet.

15 Message Passing Interface : Interface commune de communication commune à de très nombreux systèmes, développée par un grand nombre d'industriels.

16 Parallel Virtual Machine : Ensemble logiciel permettant de fédérer un ensemble des machines reliées par un réseau comme si c'était une seule machine parallèle

Le premier prototype est constitué d'un ensemble de PC standards reliés entre eux par un réseau Ethernet constituant ainsi une ferme de PC. C'est sur ce prototype étudié au CPPM qu'a été développée la notion de composants constitués de sous-éléments. Du point de vue des communications, le choix s'est porté vers une communication basée directement sur le protocole TCP/IP, étant donnée la nature du réseau qui relie les machines et la souplesse d'utilisation attendue du prototype.

✓ Prototype SMP:

Ce prototype, développé à l'Université de Pavie en Italie, utilise lui aussi des composants modulaires tels qu'ils ont été décrits précédemment. L'implémentation du code est identique dans les deux cas. Seul le protocole de communication change. Dans ce prototype le passage d'événements entre les composants se fait par passage de référence.

✓ Prototype hybride:

Le prototype développé à l'Université d'Alberta au Canada est un hybride des deux prototypes précédents. Il s'agit d'une ferme de PC qui présente la particularité de posséder des cartes mères susceptibles d'accueillir quatre processeurs, ce qui en fait des mini-SMP. Ce prototype ne partage que le design global du filtre avec les deux autres prototypes. L'implémentation des composants est différente et ils ne sont pas modulaires. Le protocole de communication entre les différents PC est TCP/IP. Des liaisons par SCI<sup>17</sup> sont également envisagées.

Dans le reste du présent chapitre nous nous intéresserons plus particulièrement au prototype PC.

---

17 SCI: Scalable Coherent Interface: bus de communication dédié

## 5.2.6 L'IMPLÉMENTATION

Le design des composants comme nous avons vu, utilise une philosophie « orienté objet » [24]. Il est naturel d'effectuer une implémentation du code en utilisant un langage suivant la même philosophie, permettant ainsi de définir chacun des éléments des composants comme un objet. Ainsi on trouve un objet s'occupant des entrées de type client, un autre pour les entrées de type serveur et de même pour les sorties. C'est notamment en implémentant une nouvelle version de ces objets que l'on peut changer le protocole de communication des composants du flot des données, les autres objets restant inchangés. Ainsi on construit les composants à partir d'un objet principal, d'un objet de contrôle et d'un ensemble d'objets d'entrée-sortie correspondant à divers protocoles de communication.

Le code du flot des données a été implémenté en langage C++ initialement sur une ferme composée de quatre PC ayant Linux (Red Hat 6.2) pour système d'exploitation. Cette implémentation est le résultat de plusieurs itérations entre développement de code et tests.

Le portage du code sur Solaris et True64 Unix a également été réalisé. Portage ne signifie pas développement spécifique pour chacun des systèmes d'exploitations cités, mais plutôt implémentation fonctionnelle sous ces systèmes d'exploitation. Cette volonté d'avoir une implémentation compatible avec plusieurs systèmes d'exploitation répond à deux exigences.

- ✓ D'un point de vue conceptuel, le développement sur différents systèmes d'exploitation permet une appréhension des problèmes liés au code plus facile. Des problèmes qui peuvent rester cachés, ou rester difficilement détectables sous un seul système, apparaissent plus clairement sous un autre. Cela est également vrai pour ce qui est de la diversité des compilateurs, certains étant plus contraignants que d'autres.
- ✓ Le fait de développer un code fonctionnel sous plusieurs systèmes d'exploitation correspond parfaitement à l'idée qui est de présenter le filtre d'événements comme un système générique susceptible de s'adapter à différentes architectures

matérielles et logicielles.

Une version du code a également été implémentée en langage JAVA, l'intérêt d'une telle version étant la portabilité de facto d'un code écrit en un tel langage. Nous verrons lors des tests que du point de vue performance, cette version du flot des données s'est avérée décevante. Par contre, d'un point de vue conceptuel, le développement du code en langage JAVA nous a apporté une expertise intéressante, utilisée ensuite pour une nouvelle itération de l'implémentation.

Concernant le protocole de communication, nous avons été guidés par le fait que nous travaillions sur une ferme de PC connectés par un réseau Ethernet. Notre choix s'est donc naturellement porté sur un protocole TCP/IP. Afin de tester le flot des données sur le prototype SMP, une communication par passage de référence a également été implémentée.

Des objets d'entrée-sortie implémentant un protocole de communication de type MPI peuvent être envisagés, afin de prendre en compte les spécificités matérielles de l'architecture sur laquelle pourrait être déployé le code.

## **5.3 SUPERVISION : CONTROLE ET MONITORAGE**

La supervision consiste à surveiller et contrôler le flot des données. L'élément "utile" de la supervision est l'interface graphique qui va présenter à l'utilisateur l'ensemble des informations mais également lui permettre de contrôler le flot des données [25].

### **5.3.1 CONTROLE DU FLOT DES DONNÉES**

Le contrôle du flot des données consiste dans un premier temps en la possibilité de démarrer et arrêter les composants constituant le flot des données. Pour cela il est néces-

saire de connaître la configuration matérielle du filtre, c'est à dire le nombre d'unités de calcul disponibles (nombre de processeurs) et le type de ces unités. Il faut également connaître la configuration logicielle que l'on souhaite déployer sur le filtre. Cette configuration correspond à l'affectation des composants aux diverses unités de calcul, à définir les relations entre les composants et à définir les paramètres des composants comme la profondeur des FIFO par exemple.

Le contrôle du flot des données ne se limite pas à la possibilité de démarrer et arrêter un processus. Il est également utile de pouvoir changer durant le fonctionnement du filtre les paramètres de certains composants, du moins pendant la phase de prototypage. Il doit ainsi être possible de changer la taille de FIFO d'un composant ou de redéfinir les paramètres de reconstruction et de sélection.

### **5.3.2 MONITORAGE DU FLOT DES DONNÉES**

En dehors des périodes de démarrage et d'arrêt, où le superviseur contrôle le filtre d'événements, celui-ci effectuera essentiellement une tâche de surveillance: surveillance de l'état des composants (actif ou inactif), surveillance des bandes passantes, de la bonne répartition des tâches, des remplissages des FIFO, des consommations de CPU, utilisation de la mémoire, etc...

Un élément important d'étude à ce niveau est de définir quelles sont les informations pertinentes à fournir à l'utilisateur et quelle est la hiérarchisation que l'on va faire des informations. Le filtre d'événement est un ensemble de plusieurs milliers de processeurs sur lesquels tourneront plusieurs milliers de processus. L'étude de la pertinence des informations monitorées est une chose importante et difficile qui demande un échange avec les personnes qui vont avoir en charge l'utilisation du système.

### **5.3.3 CONTRAINTES SUR LE SUPERVISEUR**

Contrôle et surveillance sont les deux objectifs qui incombent au superviseur du filtre d'événement. Ces objectifs et les conditions dans lesquelles doit s'insérer ce système lui confèrent un liste de contraintes qui sont :



- ✓ Être robuste.
- ✓ Garantir la fiabilité des informations qui sont monitorées.
- ✓ Être indépendant : de l'architecture du filtre certes mais aussi du flot des données lui-même, la seule interaction entre supervision et flot des données se faisant à travers l'élément de contrôle des composants du flot des données.
- ✓ Ne pas dépendre de l'échelle du flot des données.
- ✓ Présenter une interface graphique conviviale et ergonomique.

#### 5.3.4 SOLUTIONS POSSIBLES

On peut très bien envisager de séparer le superviseur en deux sous-systèmes, un pour le contrôle et un pour la surveillance. Ainsi le contrôle et la collecte des informations se feraient à travers deux systèmes distincts, seule l'interface utilisateur étant commune. Dans ce schéma de conception, de nombreuses solutions sont envisageables. Une première approche pourrait comporter deux systèmes indépendants, le contrôle ne nécessitant qu'un système de gestion de processus et de communication (pour contrôler les composants), et le monitoring ne nécessitant qu'un moyen pour récupérer de l'information. Cette récupération pourrait être faite par une collecte auprès de chaque unité de calcul ou simplement par envoi de cette information par chaque composant à un serveur central. Cette solution permet l'utilisation d'une pléthore de systèmes tels que des bases de données, des services de messagerie, des services d'information, de gestionnaire de processus, ... Cette conception de deux sous-systèmes qui devront interagir entre eux risque d'aboutir à un système complexe et lourd, ce qui n'est pas proche de l'idée de concevoir un système robuste.

Une autre solution consiste à exécuter les deux tâches, de contrôle et de monitoring à travers un même mécanisme, limitant de ce fait le nombre de systèmes de communications liés au filtre et diminuant ainsi sa complexité. Cette solution a été retenue et a été réalisée par l'utilisation d'une technologie innovante qui est celle des agents mobiles.

## 5.3.5 TECHNOLOGIES RETENUES

### 5.3.5.1 BASE DE DONNÉES DE CONFIGURATION

Précédemment nous avons établi qu'il était nécessaire de définir la configuration de flot des données que nous souhaitons déployer. Cela a été réalisé en utilisant un fichier de configuration qui contient un ensemble d'informations telles que le nom, le type et les paramètres de chaque composant affecté à chaque unité de calcul.

Dans le cadre du prototype PC, toutes ces informations sont définies dans un fichier XML<sup>18</sup> qui est généré par une interface graphique écrite en langage Tcl/Tk. A travers cette interface, on définit le service de nom (Naming Service), les composants distributeurs, les composants de traitement de tâches et les composants de collecte. A chacun de ces composants correspondent des paramètres qui sont également configurables à l'aide de l'interface. Il s'agit en particulier de la profondeur des FIFO. Dans le cas où l'on simule les tâches de traitement, des paramètres supplémentaires sont utilisés comme le taux de réjection des événements.

On définit également à travers cette interface (Figure 28) la localisation de deux éléments: le SFI (Sub Farm Input) et SFO (Sub Farm Output) qui simulent respectivement le port d'entrée du filtre, en injectant des événements dans le flot des données, et le port de sortie du filtre.

Certaines options de débogage peuvent également être définies dans le fichier de configuration. On trouve notamment la possibilité de création de fichiers d'information pour chaque composant, ou la possibilité d'utiliser un système de report d'information distant tel que NetLogger<sup>19</sup>.

La génération d'identificateurs s'effectue de façon automatique.

---

<sup>18</sup> Extensible Markup Language: Format de données utilisant des balises.

<sup>19</sup> NetLogger : Produit permettant d'implémenter un service d'information dans un code à des fins de débogage

The screenshot shows a graphical configuration interface with a menu bar (File, Machines, Global variables, Location, Defaults, Utilities, Help) and a cyan background. The sections are as follows:

- General stuff:** Run number (1), Geometry file (GeomRun.cz), Naming Service (machine), eff004, port (7200), Log level (No log), NetLog level (No log), NetLog host (maracuja).
- FIFOs Main Dataflow:** Exec (Fifo), SFI, Exec (sfi), SFO, Exec (sfo).
- Distributor:** Exec (D1d2), D1 FIFO depth (5), D2 FIFO depth (5), # event types (1), Use D3 & C1 (no).
- Processing Tasks:** Input FIFO depth (5), Output FIFO depth (5), Exec event type # 0 (PT), rejection (%) (0), # iter min (1), # iter max (1), seed (1).
- Collector:** exec (Fifo), FIFO depth (5).
- Sub-farm 0:** A yellow header bar.
- Machines:** SFI (eff002), SFO (eff002), Distributor (eff002), DGB (no), Collector (eff002).
- Processing Tasks:** A table with two rows:
 

|        |   |                |
|--------|---|----------------|
| eff001 | 2 | PT for type #0 |
| eff002 | 2 | PT for type #0 |

Figure 28 : Interface graphique du Configurateur

Le recours à un fichier de configuration général comprenant l'ensemble des informations nécessaires au flot des données et à la supervision de la ferme s'avère très pratique. On évite ainsi la dispersion et donc la perte de cohérence des informations de base concernant le filtre d'événements. La définition d'un ensemble de configurations du filtre peut être également définie à l'avance à travers différents fichiers de configuration.

### 5.3.5.2 LES AGENTS MOBILES

Les agents mobiles sont des éléments qui peuvent « voyager » de machine en machine. Une fois dans l'une d'entre elles, ils peuvent agir comme un utilisateur local et par exemple, communiquer avec un processus ou tout simplement démarrer ou tuer un processus. Ces agents mobiles sont générés à partir d'un serveur vers lequel ils retournent après avoir parcouru l'ensemble des machines auxquelles ils avaient été affectés. Cette technologie présente tous les avantages désirés pour être implémentés dans le supervi-

seur.

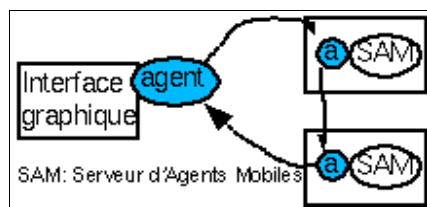


Figure 29 : Principe des agents mobiles

Dans notre cas des agents mobiles sont envoyés dans les machines hébergeant les processus du flot des données, la granularité du nombre de machines par agent étant paramétrisable. Ces agents démarrent les processus du flot des données et éventuellement les arrêtent, et, de façon régulière, ils parcourent les différentes machines pour entrer en communication locale, via un protocole UDP<sup>20</sup>, avec l'élément de contrôle de chaque composant du flot des données. Ainsi ils peuvent contrôler ces éléments mais aussi récolter des informations qu'ils ramènent au niveau du serveur d'agents.

Il existe différents produits proposant des agents mobiles. Notre choix s'est porté sur le produit VOYAGER qui présente l'avantage d'être disponible sous la forme d'une bibliothèque où l'on peut puiser des fonctions, alors que les autres produits apparaissent plus comme des environnements de travail dans lesquels doit s'insérer l'application, ce qui bien sûr est plus contraignant et laisse moins de liberté de conception.

Le choix des agents mobiles s'est avéré être très intéressant, car ils permettent de "mettre" dans chaque machine du filtre un utilisateur virtuel.

### 5.3.5.3 FONCTIONNEMENT

Le schéma (Figure 30) montre l'utilisation qui est faite des agents mobiles au sein de la supervision du filtre

<sup>20</sup> UDP : User Data Protocol : Protocole de communication.

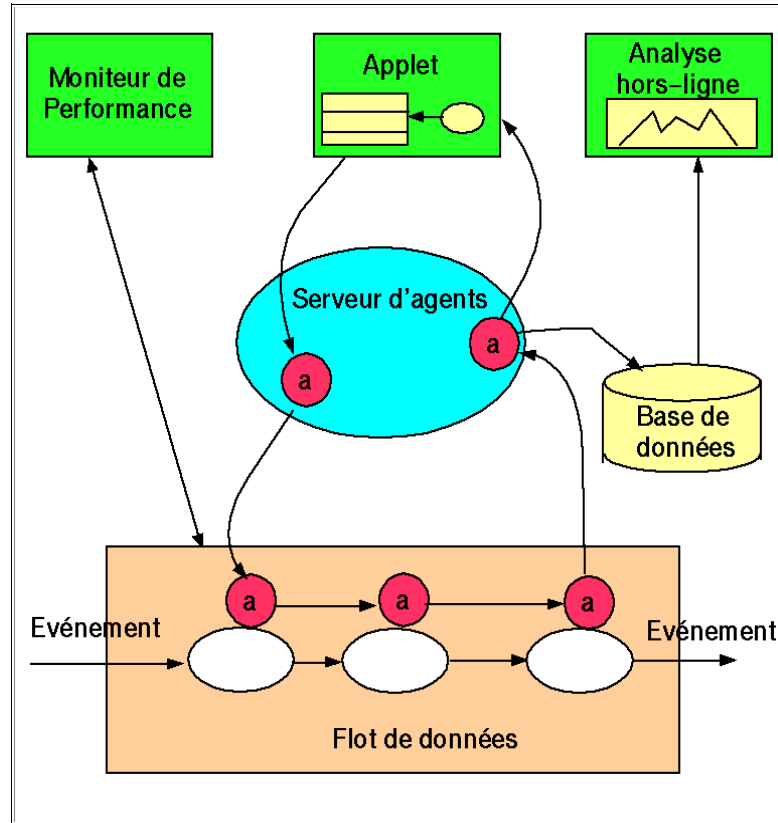


Figure 30 : Déplacement des agents mobiles dans le flot des données

Les informations collectées par l'ensemble des agents peuvent soit être disposées dans une base de données, pour être étudiées ultérieurement ou simplement envoyées dans un *applet* pour être mis à la disposition de l'interface graphique du superviseur. L'enregistrement dans une base de données permet d'avoir l'historique du filtre et donc de pouvoir facilement suivre son évolution au cours du temps.

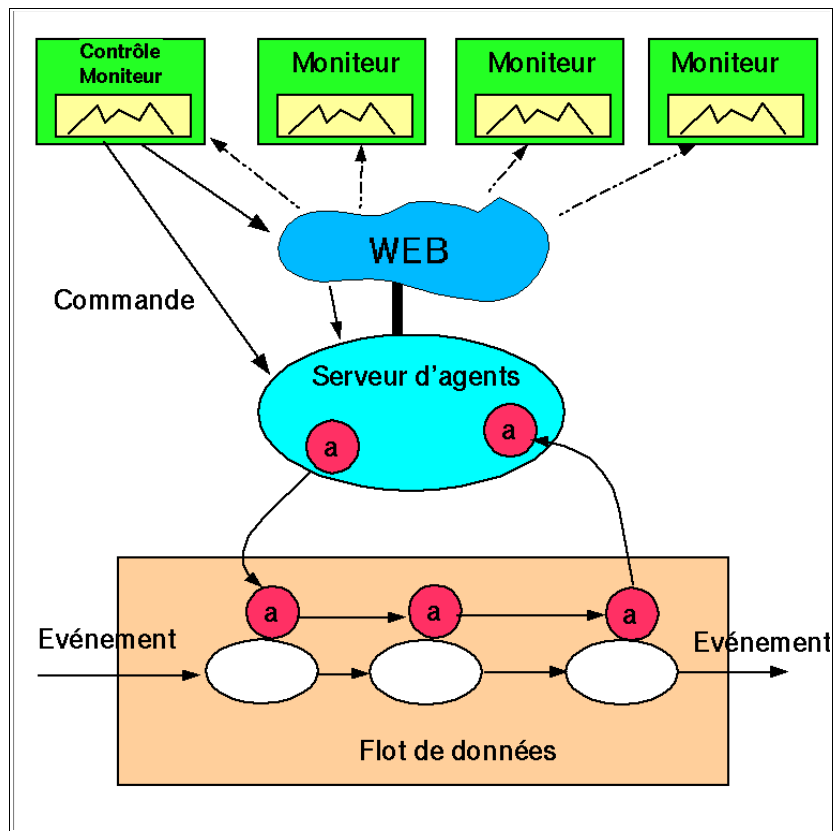


Figure 31 : Contrôle et Surveillance

L'intérêt comme on peut le voir sur le schéma (Figure 31) est de pouvoir distribuer ainsi les informations concernant le filtre sur plusieurs postes de surveillance et cela à travers le WEB.

Cependant, afin d'éviter des conflits, il est absolument nécessaire que seul un poste de surveillance soit à même de contrôler le flot des données. L'implémentation actuelle du superviseur possède cette fonctionnalité ainsi que la possibilité de basculer le mode contrôle à un autre poste de surveillance après identification par un mot de passe. Le poste initialement en mode de contrôle passe alors automatiquement en mode de surveillance.

#### 5.3.5.4 ADEQUATION DES AGENTS MOBILES

L'utilisation des agents mobiles permet de répondre à de nombreux points du cahier des charges.

Tout d'abord, ils assurent un système de communication, que ce soit pour le contrôle des composants ou la collecte des informations, indépendant du flot de données mais aussi de l'architecture matérielle. D'autre part, leur utilisation est indépendante de la taille du système à superviser. Enfin ils sont extrêmement simples d'emploi ce qui est un argument fort pour ce qui concerne la maintenance de leur implémentation.

#### 5.3.6 IMPLEMENTATION

L'implémentation du superviseur, interface graphique comprise, a été réalisée en langage JAVA, garantissant ainsi une portabilité quasi-générale du code. La version des agents mobiles choisie étant également implémentée en JAVA, l'ensemble de la supervision (collecte d'information, contrôle et interface graphique) est indépendante de la plate-forme.

##### 5.3.6.1 INTERFACES UTILISATEURS

L'interface graphique (Figure 32) développée est la même que l'on soit en mode contrôle et surveillance ou en mode surveillance uniquement, la différence étant qu'en mode surveillance l'ensemble des fonctions de contrôle est désactivé.

En mode contrôle, il suffit de charger le fichier de configuration, pour ensuite être à même de lancer, contrôler, surveiller et arrêter le flot des données. Ce panneau principal donne des informations concernant l'état global du flot des données.

On peut voir (Figure 32) que 194 processus sur 195 sont actifs et que le processus nommé *subf0\_pt\_034\_0\_2* normalement présent sur la machine *eff034* n'est pas actif. Ici l'identificateur du processus nous permet de connaître sa nature (PT), sa situation (dans la sous-ferme 0), la machine sur laquelle il devrait être actif (*eff034*).

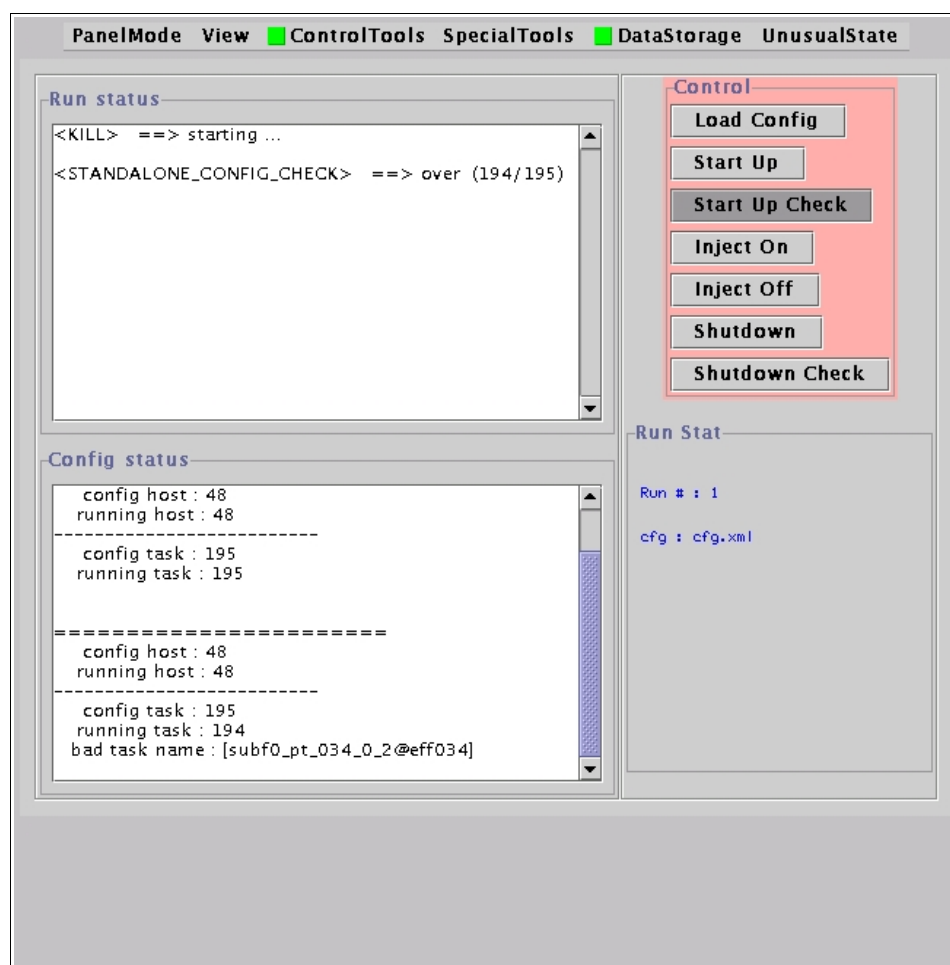


Figure 32 : Panneau principal

Ce panneau principal sert principalement pour le démarrage et l'arrêt du flot des données. Un second panneau (Figure 33) permet de surveiller le système de façon plus globale. Il présente sous forme d'une arborescence les sous-fermes, les machines et les processus en indiquant le nombre de processus actifs sur le nombre qui devraient l'être.

Un code couleur (rouge) signale l'absence d'un processus à la fois au niveau du filtre, des sous fermes, de la machine et du processus. Ce panneau permet de savoir instantanément si l'ensemble des processus de la ferme est actif ou pas. Il correspond au panneau



permettant d'informer l'utilisateur du filtre sur la bonne marche du système.

La représentation sous forme d'arbre permet également de facilement savoir si le problème est localisé à un processus, à une machine voire à une sous-ferme.

| Basic |                  |          |         |         |
|-------|------------------|----------|---------|---------|
|       | Name             | TrueNode | SubTree | Problem |
| EF    |                  |          |         | 1/2     |
| ☐     | subf1            |          |         | 24/24   |
| ☐     | subf0            |          |         | 24/25   |
| ☐     | monarc01         | 1        |         | 8/8     |
|       | subf0_d1         | 1        |         |         |
|       | subf0_d2_0       | 1        |         |         |
|       | subf0_d2_1       | 1        |         |         |
|       | subf0_c2         | 1        |         |         |
|       | subf1_d1         | 1        |         |         |
|       | subf1_d2_0       | 1        |         |         |
|       | subf1_d2_1       | 1        |         |         |
|       | subf1_c2         | 1        |         |         |
| ☐     | eff033           | 1        |         | 4/4     |
| ☐     | eff034           | 1        |         | 3/4     |
|       | subf0_pt_034_0_1 | 1        |         |         |
|       | subf0_pt_034_0_2 | 0        |         | ???     |
|       | subf0_pt_034_1_1 | 1        |         |         |
|       | subf0_pt_034_1_2 | 1        |         |         |
| ☐     | eff035           | 1        |         | 4/4     |
| ☐     | eff036           | 1        |         | 4/4     |
| ☐     | eff037           | 1        |         | 4/4     |
| ☐     | eff038           | 1        |         | 4/4     |
| ☐     | eff039           | 1        |         | 4/4     |
| ☐     | eff040           | 1        |         | 3/3     |
| ☐     | eff041           | 1        |         | 4/4     |
| ☐     | eff042           | 1        |         | 4/4     |
| ☐     | eff043           | 1        |         | 4/4     |
| ☐     | eff044           | 1        |         | 4/4     |
| ☐     | eff045           | 1        |         | 4/4     |
| ☐     | eff046           | 1        |         | 4/4     |
| ☐     | eff047           | 1        |         | 4/4     |
| ☐     | eff048           | 1        |         | 4/4     |
| ☐     | eff049           | 1        |         | 4/4     |
| ☐     | eff050           | 1        |         | 4/4     |
| ☐     | eff051           | 1        |         | 4/4     |
| ☐     | eff052           | 1        |         | 4/4     |
| ☐     | eff053           | 1        |         | 4/4     |
| ☐     | eff054           | 1        |         | 4/4     |
| ☐     | eff055           | 1        |         | 4/4     |
| ☐     | eff056           | 1        |         | 4/4     |

Figure 33 : Présentation arborescente

Il existe un troisième panneau (Figure 34) qui va nous fournir l'ensemble des informations collectées par les agents. Ce panneau présente des informations comme le taux de remplissage des FIFO, les taux de rejet réel par composant PT, le nombre d'événements ayant traversés les composants, la bande passante par composant, etc....

Ce panneau est plutôt un panneau réservé aux "experts" du filtre d'événements permettant d'avoir accès aux informations de base du flot des données.

Basic  UpdateMode HistogramSelect

Total Processes : 195  
 Unlinked Processes : 1  
 Selected Processes : 0  
 Delta Time : 185

| Host     | Process         | reject | cpu% | throughput(n... | inEvents | occupancy | connect |
|----------|-----------------|--------|------|-----------------|----------|-----------|---------|
| eff071   | subf1_pt_071... | 100    | 0    | 24              | 105      | 0         | 1       |
| eff071   | subf1_pt_071... | 100    | 0    | 24              | 107      | 0         | 1       |
| eff070   | subf1_pt_070... | 100    | 0    | 25              | 106      | 0         | 1       |
| eff070   | subf1_pt_070... | 100    | 0    | 21              | 105      | 0         | 1       |
| eff070   | subf1_pt_070... | 100    | 0    | 25              | 105      | 0         | 1       |
| eff070   | subf1_pt_070... | 100    | 0    | 25              | 105      | 0         | 1       |
| monarc01 | subf0_d1        | 0      | 0    | 2076            | 12238    | 17        | 1       |
| monarc01 | subf0_d2_0      | 0      | 0    | 1036            | 6119     | 67        | 1       |
| monarc01 | subf0_d2_1      | 0      | 0    | 1040            | 6119     | 65        | 1       |
| monarc01 | subf0_c2        | 0      | 0    | 0               | 0        | 0         | 1       |
| monarc01 | subf1_d1        | 0      | 0    | 2155            | 9752     | 15        | 1       |
| monarc01 | subf1_d2_0      | 0      | 0    | 1076            | 4876     | 68        | 1       |
| monarc01 | subf1_d2_1      | 0      | 0    | 1080            | 4876     | 68        | 1       |
| monarc01 | subf1_c2        | 0      | 0    | 0               | 0        | 0         | 1       |
| eff039   | subf0_pt_039... | 100    | 0    | 23              | 128      | 0         | 1       |
| eff039   | subf0_pt_039... | 100    | 0    | 23              | 127      | 0         | 1       |
| eff039   | subf0_pt_039... | 100    | 0    | 19              | 129      | 0         | 1       |
| eff039   | subf0_pt_039... | 100    | 0    | 19              | 131      | 0         | 1       |
| eff038   | subf0_pt_038... | 100    | 0    | 19              | 128      | 0         | 1       |
| eff038   | subf0_pt_038... | 100    | 0    | 23              | 127      | 0         | 1       |
| eff038   | subf0_pt_038... | 100    | 0    | 19              | 131      | 0         | 1       |
| eff038   | subf0_pt_038... | 100    | 0    | 19              | 128      | 0         | 1       |
| eff069   | subf1_pt_069... | 100    | 0    | 20              | 106      | 0         | 1       |
| eff069   | subf1_pt_069... | 100    | 0    | 25              | 107      | 0         | 1       |
| eff069   | subf1_pt_069... | 100    | 0    | 20              | 107      | 0         | 1       |
| eff037   | subf0_pt_037... | 100    | 0    | 23              | 128      | 0         | 1       |
| eff037   | subf0_pt_037... | 100    | 0    | 23              | 128      | 0         | 1       |

Figure 34 : Panneau de contrôle des paramètres

Enfin il existe une série de panneaux permettant d'arrêter ou redémarrer un composant en particulier, ou de changer les paramètres d'un composant ou d'un ensemble de composants. D'autres panneaux permettent, de choisir la fréquence à laquelle on envoie les agents collecter l'information, ou d'envoyer les informations dans une base de données pouvant être interrogée par des requêtes de type SQL<sup>21</sup>,...

21 SQL: Langage de spécifications de requêtes qu'utilisent de nombreuses bases de données parmi les plus répandues.

### 5.3.6.2 ADEQUATION DES SOLUTIONS APPORTEES

L'implémentation JAVA qui a été faite du système de supervision répond naturellement au besoin d'indépendance vis à vis de l'architecture matérielle. Les interfaces définies permettent notamment de présenter de façon claire et concise les informations essentielles au monitoring du filtre d'événements. Elles permettent d'identifier de façon rapide et précise l'origine des éventuels problèmes, par la connaissance de la ferme, de la machine et du processus mis en cause.

L'interface de contrôle permet le contrôle général de la ferme mais aussi le contrôle de chacun des processus mis en jeu dans le flot de données. Enfin la possibilité qui est donnée d'accéder à travers le WEB aux informations de monitoring en fait un système ouvert à l'ensemble de la communauté.

## 5.4 TESTS ET MESURES

Le flot des données ainsi que la supervision ont été validés à la suite d'une série de tests portant sur la performance et la fonctionnalité des systèmes.

### 5.4.1 CONDITIONS DES TESTS

Nous avons effectué les développements du flot des données et de la supervision sur une mini-ferme de PC composée de quatre machines bi-processeurs. Afin de réaliser des tests dans des configurations plus réalistes, nous avons ensuite utilisé une ferme de PC mise à notre disposition par la division IT<sup>22</sup> du CERN. Il s'agit d'une ferme composée d'une centaine de PC bi-processeurs. Lors de nos tests, seulement une cinquantaine de machines a été utilisée. Nous avons également à notre disposition une machine SUN quadri-processeurs ULTRA SPARC équipée d'une carte réseau Gigabit servant d'injecteur pour la ferme. Les PC étaient équipés de carte réseau 100 Mb/s et reliés entre eux par un ensemble de commutateurs 3COM. Était également mis à notre disposition un PC

---

<sup>22</sup> Technology Information : Division ayant en charge l'informatique au CERN

bi-processeurs équipé d'une carte Gigabit. L'ensemble des PC utilisait Linux (Red Hat 6.1) comme système d'exploitation et la quadri-processeurs utilisait Solaris 2.6

Pour ce qui concerne la supervision, nous avons testé plusieurs versions de machines virtuelles JAVA, toutes utilisant la distribution SUN « Java Development Toolkit » (JDK) : la JDK 1.1.8, 1.2.2 et 1.3. De même la version du flot des données implementée en JAVA utilisait également ces versions de JDK.

## 5.4.2 TESTS SUR LE FLOT DES DONNÉES

Les tests et les mesures de performances qui ont été menés sur le flot des données s'inscrivent dans deux axes différents. Pour une part, ils avaient pour objectif de tester qualitativement les fonctions du flot des données et pour l'autre part de mesurer les performances, notamment en termes de bande passante.

### 5.4.2.1 TESTS DES FONCTIONS

Dans cette série, les tests sur le flot des données ne sont pas vraiment quantifiables. Ils consistent simplement à vérifier que le design choisi et le code implementé apportaient bien la fonctionnalité qui était demandée. Cette série de tests a notamment permis de vérifier :

- ✓ Le bon démarrage et arrêt des différents composants.
- ✓ Le bon passage des événements à travers l'ensemble du flot des données
- ✓ La possibilité de re-configurer la ferme dynamiquement.
- ✓ La robustesse du flot des données. Des processus ont été arrêtés, démarrés sans pour autant bloquer l'ensemble du flot des données. Le réseau et des machines ont été débranchés physiquement durant un test pour simuler des pannes matérielles.
- ✓ Le contrôle des différents composants, par des changement de paramètres des

composants, et cela de façon dynamique.

- ✓ La fonctionnalité du DGB. Dans ce cas le DGB consistait en une simple sauvegarde sur disque.
- ✓ La fonctionnalité du programme qui génère le fichier de configuration.

#### 5.4.2.2 MESURE DE PERFORMANCES

Les mesures de performances portent essentiellement sur la mesure de la bande passante dans les différents composants. Dans cette série de tests l'injection des événements à travers la ferme est effectuée à partir de la machine quadri-processeurs équipée d'une interface Ethernet Gigabit. Les PC constituant la ferme étaient équipés d'interfaces 100 Mb/s. Cet ensemble était relié par un commutateur 100Mb/s équipé d'un port Gigabit pour la quadri-processeurs. De plus, tous les événements étaient rejetés par la tâche de traitement. Ainsi aucun événement ne passait dans le collecteur, évitant ainsi de perturber l'utilisation du réseau pour l'injection.

-Mesure de bande passante en entrée de filtre

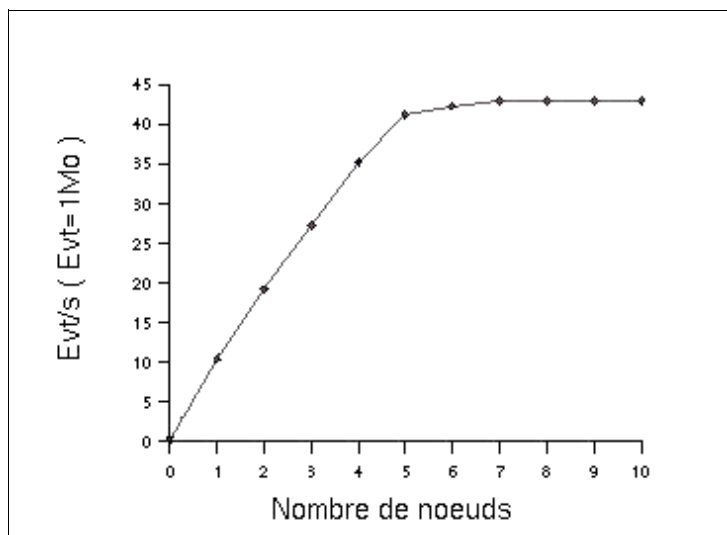


Figure 35 : Bande passante en événement par seconde dans la ferme pour un rejet de 100 % et un temps de calcul nul.

Le graphe (Figure 35) montre la bande passante en nombre d'événements (événements de 1 Mo) par seconde en fonction du nombre de noeuds de calcul utilisés. L'ensemble des processus du distributeur (D1, D2, D3) était sur une même machine ainsi que le collecteur. Sur chaque noeud de traitement étaient disposées deux tâches d'analyse qui n'effectuaient aucun travail (ne consommant pas de temps de calcul), permettant ainsi de mesurer la bande passante d'entrée maximale du filtre.

On constate tout d'abord la parfaite linéarité de l'évolution de la bande passante au fur et à mesure que le nombre de noeuds d'analyse augmente. La saturation obtenue à partir de cinq noeuds provient du fait que c'est l'injecteur qui n'arrive plus à fournir la ferme en événements.

Cette linéarité montre que chaque noeud de la ferme est capable d'absorber autour de 10,4 événements par seconde, chaque événement ayant ici une taille de 1 Mo. Cette limite de 10,4 événements par seconde est proche de la limite de l'interface réseau ( $10,4 \cdot 8 \approx 83,2$  Mb/s).

Le même test réalisé sous le système d'exploitation Windows NT fait apparaître un comportement identique, mais avec un facteur de linéarité plus faible correspondant à

une limite d'interface réseau plus faible. Cette différence entre les deux systèmes d'exploitation s'explique simplement par la différence de performances des pilotes de périphériques (*drivers*).

Le même type de mesure de bande passante a été réalisé avec des tâches d'analyses qui réclament un temps de calcul de l'ordre de ce que l'on attend pour le filtre d'événements, c'est-à-dire de l'ordre de la seconde. On constate dans ce cas (Figure 36) que l'approvisionnement de tâches d'analyses est parfaitement équilibré entre les différents noeuds de la ferme assurant ainsi un *load balancing* correct entre les différentes unités de calcul du filtre.

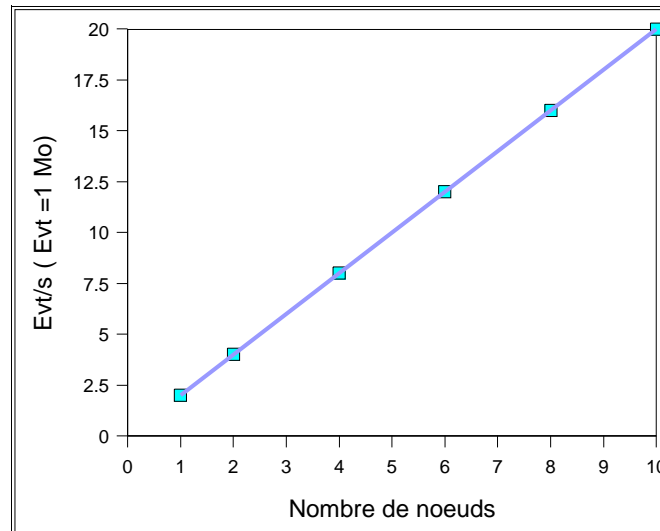


Figure 36 : Bande passante pour un temps de traitement de tâche de 1 seconde avec deux tâches d'analyse par noeud.

Ces mesures de performances ont également été réalisées pour la version JAVA du flot des données, comme on peut le voir sur le graphe (Figure 37). Ici la linéarité est encore présente mais les performances au niveau du réseau sont plus mauvaises que dans le cas de l'implémentation C++. Cela s'explique facilement par le fait que la librairie réseau de JAVA est moins performante que celle du C++. Le développement de la version JAVA a permis de remettre en cause en partie la conception du code, en particulier la définition des objets et de leurs agencements. Ces améliorations ont été ensuite reprises dans

une nouvelle version de l'implémentation C++.

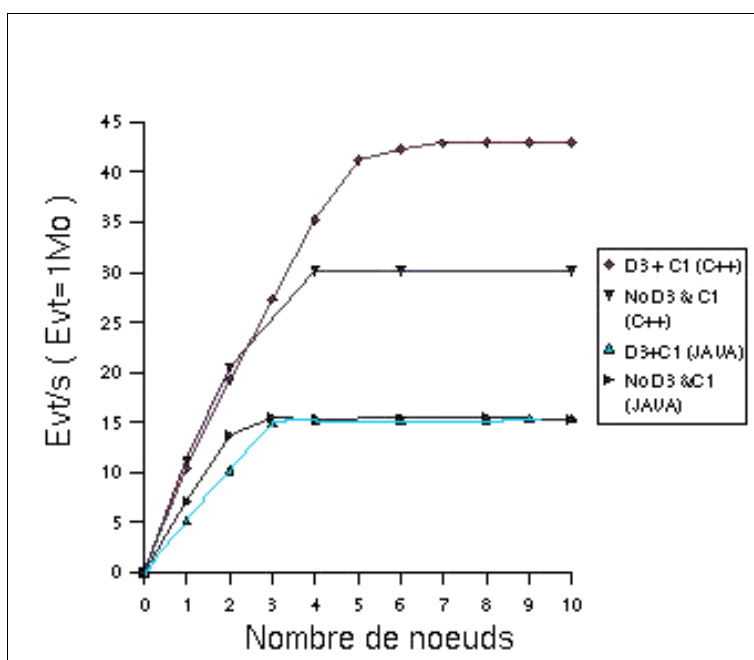


Figure 37 : Bande passante avec et sans présence des composants D3 et C1 et cela pour l'implémentation C++ et JAVA

Cette série de mesures a notamment permis de mettre en évidence l'importance des composants D3 et C1, se situant en amont et en aval des tâches d'analyses. Le graphique (Figure 37) montre la dégradation des performances lorsqu'on omet de mettre ces files d'attente devant et derrière chaque tâche d'analyse. La présence des composants D3 et C1 permet de créer un réservoir d'événements pour chaque tâches d'analyses alimentant celles-ci plus rapidement en événements. Il faut aussi noter que le nombre de connexions vers le composant D2 est notablement diminué puisque les différentes tâches d'analyse d'un même nœud sont connectées au même composant C3.

Ces séries de mesures de bande passante ont été réalisées en se plaçant dans un contexte plus réaliste, c'est à dire en rejetant neuf événements sur dix. Les résultats de linéarité et de bande passante ont été confirmés. Bien sûr, dans ce cas la bande passante maxi-



male au niveau des tâches d'analyses se répartissait entre les événements entrants et les événements sortants vers le collecteur.

Cette série de mesures nous a également permis de mettre en évidence l'importance du processeur au niveau des machines servant d'injecteur. De façon plus générale, les cartes réseaux performantes (Gigabit/seconde), ou du moins celles qui étaient à notre disposition, peuvent se révéler très gourmandes en ressources CPU lorsqu'elles sont utilisées intensivement. L'utilisation de cartes réseau équipées de DMA<sup>23</sup> et de pilotes performants peut s'avérer nécessaire lorsqu'on souhaite avoir à la fois un réseau rapide et une capacité de calcul importante, ce qui est le cas dans les tâches d'analyses du filtre.

#### 5.4.2.3 CONCLUSIONS DES TESTS DE DATAFLOW

Les résultats des tests précédents permettent, dans un premier temps, de mettre en avant le fait que le système de flot de données répond aux attentes. En effet, il devrait être possible d'affecter au filtre une centaine de ports de sortie du constructeur d'événement, ce qui correspond à la gestion par ferme d'un flot de données de 10 événements par seconde. Cette valeur de 10 événements par seconde se trouve exactement dans le domaine (voir la figure 13) pour lequel le système de distribution des événements s'est montré conforme et efficace. Autrement dit, le système de distribution des événements tel qu'il a été déployé sur le prototype PC répond d'ors et déjà à nos besoins.

Cependant il faut bien garder à l'esprit que les évaluations actuelles des besoins (1 seconde de calcul, 1kHz de données entrante, 100Hz de données sortantes, taille des événement de 1 à 2 Mo) peuvent évoluer à la hausse. Il serait donc intéressant de voir comment se comporterait le système si les besoins étaient autres. On a déjà mis en évidence l'importance des drivers réseaux notamment lorsqu'on s'approche des limites hautes pour lesquelles ils sont conçus.

---

<sup>23</sup> Direct Memory Access : Permet l'accès direct à la mémoire des périphériques de données tels que des disques dur ou des cartes réseaux. Permettant ainsi de s'affranchir d'un certains nombres de copies mémoires qui monopolisent le processeur.

### 5.4.3 TESTS SUR LA SUPERVISION

L'essentiel des tests que nous avons effectués au niveau de la supervision sont des tests de fonctionnalité. Nous avons notamment vérifié :

- ✓ La possibilité de contrôler, démarrer et arrêter les composants du flot des données.
- ✓ La bonne collecte des informations.
- ✓ La possibilité d'identifier l'origine, sous-ferme, machine, processus qui a un problème.
- ✓ La robustesse du système, en arrêtant et en redémarrant des processus de supervision sans observer de perturbations dans le comportement de la ferme.
- ✓ La possibilité de contrôler des systèmes de tailles différentes : des fermes de 10 à plus de 1000 processus ont été supervisées.
- ✓ L'utilisation de panneaux de supervision multiples un seul d'entre eux étant affecté au contrôle).

Des mesures de vitesse ont également été faites pour évaluer dans un premier temps quel devait être le nombre de machines affectées à un agent. La solution d'une parallélisation extrême du nombre d'agent (un par machine) n'est pas la meilleure. En effet, un problème d'« embouteillage » se pose au niveau de l'envoi et surtout du retour des agents dans le serveur. Dans le cas d'une ferme de PC de 50 noeuds, nous avons déterminé expérimentalement que la valeur de un agent pour 11 machines est, semble-t-il, la plus intéressante. C'est celle qui offre le meilleur compromis entre démarrage du flot des données, arrêt du flot des données et collecte de l'information. Cependant cette valeur dépend essentiellement du nombre d'agents qui sortent et reviennent au serveur et par conséquent de la taille de la ferme.

#### 5.4.3.1 CONCLUSIONS DES TESTS DE SUPERVISION

Les tests réalisés sur la supervision se sont montrés très positifs. Il ne reste plus qu'à évaluer le comportement du système de supervision pour une taille de filtre plus importante, c'est-à-dire d'évaluer la possibilité d'appliquer des facteurs d'échelle à la supervision. Cette évaluation nécessite l'accès à une, ou des fermes de PC d'un volume conséquent, de l'ordre de plusieurs centaines de machines. Des dispositions sont train d'être prises pour effectuer de tels tests.

#### 5.4.4 INTEGRATION

La phase d'intégration avec le reste du système d'acquisition et de déclenchement a été réalisée.

Le flot des données du filtre d'événements a été connecté à un port de sortie de l'Event Builder et des événements simulés sont passés dans le filtre avant d'être renvoyés dans le flot de données principal. Cette intégration a été d'autant plus facile qu'une API a été définie au niveau de la SFI, facilitant ainsi l'interconnexion des deux systèmes.

La supervision a également été intégrée au système global de *run control* d'ATLAS.

#### 5.4.5 TEST D'UN PROGRAMME DE RECONSTRUCTION

Parallèlement à l'étude du prototype nous avons souhaité évaluer les codes qui devront être implémentés sur le filtre d'événements, c'est-à-dire essentiellement les codes de reconstruction. Il n'est malheureusement pas encore possible de disposer de versions des codes qui seraient représentatives des versions finales. Un travail de développement est en cours de réalisation au niveau de la collaboration, travail qui consiste à écrire ou réécrire les codes.

Nous nous sommes donc orientés sur l'idée qui consiste non pas à étudier les codes à proprement parler mais à acquérir une première expérience en ce qui concerne la valida-

tion de ceux-ci dans le contexte d'une utilisation dans le filtre d'événements. Nous n'avions à notre disposition que des versions préliminaires des codes, et en particulier d'un code de reconstruction de traces : XKALMAN++. En partenariat avec la société Compaq, nous avons ainsi effectué une série de tests visant non pas à évaluer le code choisi, mais plutôt à bien cerner les points fondamentaux auxquels devra répondre un code pour être validé pour le filtre d'événements. Par cette première approche, nous avons pu définir quelles devraient être les méthodes de validation de ces codes.

Dans un premier temps il a fallu porter le code de reconstruction XKALMAN++ dans sa version C++ sur le système d'exploitation TRUE64 Unix. Cela a permis notamment de mettre en évidence des manquements concernant la gestion du champ magnétique au sein du code. Les difficultés liées aux différences entre compilateurs ont dû être abordées, le code initial étant développé sous un compilateur générique (egcs) alors que le portage était réalisé avec un compilateur propriétaire (CXX).

Suite à de nombreux tests, un lot d'événements de type Higgs  $\rightarrow b\bar{b}$  en présence éventuellement de bruit de fond (*pile-up*) et un lot d'événements de type *jets*, ont été sélectionnés parmi les productions faites par la collaboration ATLAS. Ce type d'événements réclame énormément de ressources de la part des codes de reconstruction de traces. Une machine de type DS 20 équipée de deux processeurs Alpha EV6 et de deux 2 Go de mémoire a été mise à notre disposition par le Centre Technique Europe de Compaq à Sophia-Antipolis afin d'effectuer nos mesures [26]. Des outils de profilage et de monitoring propriétaires avait également été mis à notre disposition, permettant de mesurer l'utilisation de la CPU, de la mémoire, etc...

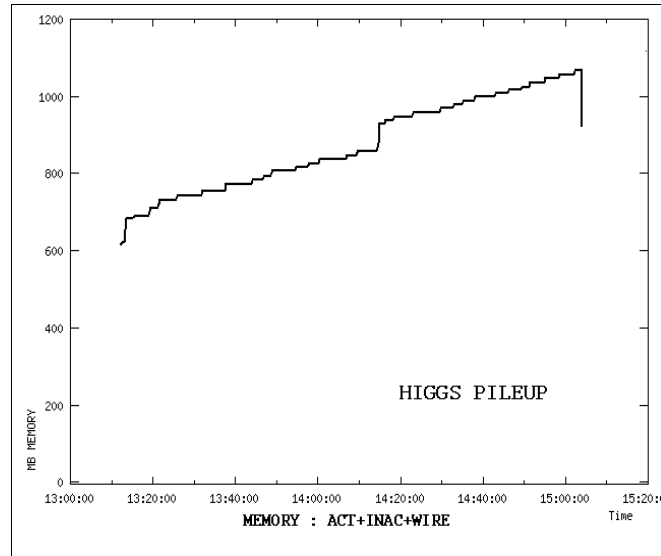


Figure 38 : Consommation de la mémoire par XKALMAN++ pour des événements de type Higgs en  $b\bar{b}$  avec pile-up

La Figure 38 montre l’occupation de la mémoire en fonction du temps (ici, l’heure absolue) pour des reconstructions d’événements de type Higgs en  $b\bar{b}$ . Chaque pas de l’escalier que fait apparaître le graphique correspond à la reconstruction d’un nouvel événement. On remarque tout d’abord que certaines reconstructions réclament des temps de l’ordre de plusieurs minutes, voire dizaines de minutes, ce qui est en désaccord avec le temps de l’ordre de la seconde qui est imparti au filtre pour effectuer sa sélection. Cela est d’autant plus vrai que XKALMAN++ ne représente qu’un des codes de reconstruction qui devront s’exécuter dans le filtre.

La seconde constatation concerne le fait que, après la reconstruction de chaque événement, la mémoire n’est pas libérée, ce qui entraîne une consommation de la ressource mémoire toujours plus importante. Ce type de comportement correspond à une fuite de mémoire dans l’implémentation du code.

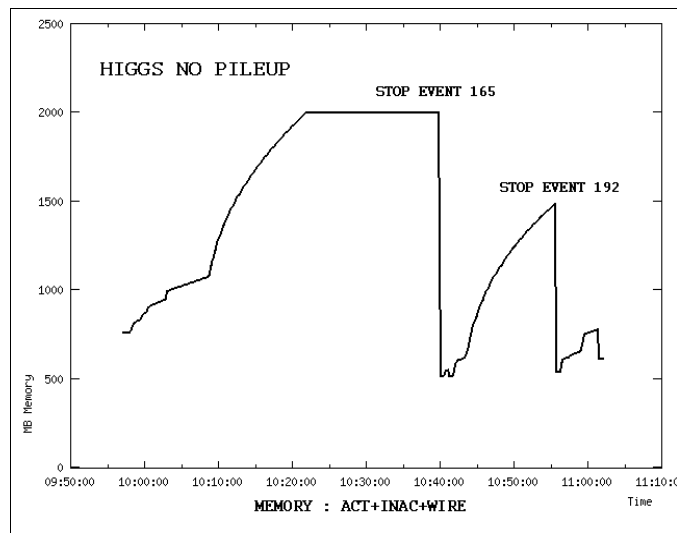


Figure 39 : Consommation de la mémoire par le code de reconstruction XKALMAN++

Le dessin précédent (Figure 39) met en évidence une fuite de mémoire plus importante, mais montre également que pour certains événements (numéros 165 et 192) la consommation de mémoire peut être très importante et aller jusqu'à utiliser l'ensemble des 2 Go de mémoire (événement numéro 165). Dans ce cas, le système va paginer et donc consommer énormément de processeur, ce qui diminuera d'autant les ressources de calcul disponible pour traiter l'événement, aboutissant finalement au blocage du système.

Ces tests ont notamment montré que certaines topologies d'événements peuvent réclamer des temps de traitement très importants et totalement incompatibles avec la contrainte temporelle à laquelle est soumis le filtre. La présence de fuites de mémoire a également été mise en évidence, fuites menant après un certain laps de temps le système dans un état bloqué. Dans le cadre de la collaboration que nous avons établie avec l'auteur du code, des corrections à l'implémentation ont pu être faites rapidement. Les résultats de tels tests permettent de rendre compte à la Collaboration des besoins (ici de fiabilité des codes) du filtre, du point de vue des applications.

Dans le même ordre d'idées, j'ai regardé l'environnement logiciel (Framework) ATHENA dans lequel s'insèrent les programmes de reconstruction. Cet environnement

est conçu pour faire tourner les programmes de reconstruction dans un environnement hors ligne, qui n'est pas celui du filtre. De ce fait, les caractéristiques de cet environnement de travail n'ont pas été optimisées pour répondre aux contraintes du filtre. En particulier, il se caractérise par le fait qu'il n'a pas été prévu pour une utilisation "locale". Ainsi, l'accès à de nombreuses ressources se fait par l'intermédiaire de systèmes de fichiers distribués tels que "afs" dont l'utilisation dans un environnement temps réel n'est pas optimale. Parmi ces ressources, de nombreuses références utilisent des bibliothèques chargées dynamiquement. Lorsque l'on désire effectuer une utilisation locale, on a affaire à des processus de très grande taille qui consomment énormément les ressources mémoire des unités de traitement. Il se caractérise également par le fait que les contraintes de temps et de performance au niveau des échanges d'information n'ont absolument pas été prises en compte, alors qu'elles sont essentielles dans le contexte du système de déclenchement.

La majorité des applications de reconstruction ne sont pas encore finalisées. Il est donc extrêmement difficile d'évaluer les besoins qu'elles vont réclamer au niveau des ressources informatiques à mettre en oeuvre dans le filtre. Cette remarque s'applique également à l'environnement de travail qui est, lui aussi, encore en développement. La conséquence de ce flou autour des besoins réels des logiciels en termes de ressources est qu'il n'est pas encore possible de définir précisément les caractéristiques (puissance CPU, mémoires, ...) des unités de calcul du filtre.

#### **5.4.6 RECOMMANDATIONS**

L'étude de ce code de reconstruction permet cependant de définir quelques recommandations que l'on peut faire concernant l'implémentation des algorithmes. Ces recommandations ont pour but de concevoir au mieux un code utilisable à la fois dans le contexte hors ligne et dans le contexte en ligne. Ces recommandations portent sur la conception globale du code et non pas sur les détails d'implémentation.

La première des recommandations concerne la modularité du code. Dans la mesure du possible, le code doit être élaboré de façon modulaire et compartimentée. Cette modularité permet, en plus de rendre l'implémentation propre, de pouvoir, lors d'une utilisation en ligne, ne pas utiliser certaines fonctionnalités qui peuvent s'avérer gourmandes en

temps et dont la justification n'est pas nécessaire dans ce cas. La modularité permet également de définir proprement les interfaces et de rendre ainsi plus facile la maintenance et l'évolution du code.

La seconde recommandation concerne la validation des codes. Les codes doivent bien évidemment remplir leurs fonctions mais cela ne doit pas être le seul critère de validation. Cela doit être réalisé dans un contexte sain, par-là j'entends que les ressources (mémoire, CPU, etc...) doivent être utilisées au mieux. Le cas du code étudié précédemment est l'exemple d'un code fonctionnel qui utilise mal les ressources (fuite de mémoire), ce qui dans le cas d'une utilisation en ligne peut amener au blocage du système. Le fait de développer un code multi-threadé peut également permettre d'augmenter de façon notable les performances du code suivant l'architecture matérielle sur laquelle il est déployé.

La dernière recommandation que je ferais concerne les langages dans lesquels sont implémentés les algorithmes. Il n'y a pas un langage de programmation meilleur qu'un autre. Ce qui compte, ce sont les interfaces, leur bonne définition et surtout leur respect au niveau de l'implémentation. Le choix du langage de programmation doit venir naturellement selon la structure de l'algorithme, les ressources disponibles, les contraintes et finalement selon le choix du programmeur. Certains types d'algorithmes se prêtent bien à une implémentation de type objets, d'autres non. Certaines contraintes, notamment temporelles, sont fortes, d'autres non, ce qui peut nous orienter vers un langage de programmation moins performant mais nécessitant un investissement en temps moins grand.

## 5.5 RESUME

Dans ce chapitre j'ai décrit le design du flot des données et du système de supervision que nous avons développé dans le cadre du prototype, afin de répondre aux contraintes de conception du filtre. Le design présenté ne se limite pas à ce prototype mais il est également celui retenu pour la version SMP du filtre.



L'implémentation et les choix technologiques (pour le flot des données et la supervision) qui ont été faits sont également décrits dans ce chapitre, ainsi que la campagne de tests qui a permis de valider les choix de design et de technologie.

L'intégration avec le reste du système de déclenchement et d'acquisition de l'expérience a été validée.

Des tests portant sur des logiciels devant être intégrés au sein du filtre ont montré la nécessité d'avoir un code propre et fiable.

## CHAPITRE 6

# PERSPECTIVES ET CONCLUSIONS

Au cours de l'été 2000 la communauté Trigger/DAQ du détecteur ATLAS a produit un document (TP: Technical Proposal) présentant les diverses solutions évaluées durant une période de recherche et développement qui s'est étendue sur 4 ans. Les solutions envisagées pour le filtre d'événements, comme les autres éléments du système de déclenchement et d'acquisition, ont été présentées dans ce document.

La collaboration se trouve actuellement dans une phase qui doit s'étendre jusqu'à l'été 2002 durant laquelle les évaluations, notamment d'intégration des différents systèmes, vont être réalisées. L'aboutissement de cette période correspondra à la publication du TDR (Technical Design Report), document présentant les choix technologiques qui auront été faits. A partir de là, démarrera la phase de production du système de déclenchement et d'acquisition du détecteur ATLAS. Ce planning ne concerne pas le niveau un de déclenchement qui, étant très dépendant de la construction des détecteurs, a déjà produit son TDR au cours de l'été 1998.

### 6.1 CONCLUSIONS

Les études et séries de tests menés sur le prototype PC de filtre d'événements ont notamment permis:

- ✓ De définir les contraintes liées au système et ainsi de proposer un design indépendant de l'architecture matérielle du système, que cela soit pour le système de distribution des événements ou pour le système de supervision du filtre.
- ✓ De proposer une implémentation du système de gestion du flot de données qui réponde à toutes les contraintes qui lui sont imposées. Ce système a par ailleurs montré son indépendance vis-à-vis des choix technologiques, par son déploiement sur deux modèles de filtres d'événement totalement différents du point de vue de leurs architectures matérielles.
- ✓ De proposer et d'implémenter un système de contrôle et de supervision du filtre également indépendant des futurs choix technologiques.
- ✓ D'évaluer et de valider l'utilisation de technologies innovantes telles que celle des agents mobiles JAVA.
- ✓ D'évaluer les caractéristiques d'un filtre d'événements basé sur l'idée d'une ferme de PC.
- ✓ De valider la faisabilité d'une telle réalisation et son intégration au sein du reste du système d'acquisition du détecteur ATLAS.
- ✓ De définir les besoins notamment au niveau des performances des algorithmes qui sont susceptibles d'être utilisés dans le filtre.

Le choix de la technologie qui sera employée dans le filtre d'événements n'est pas encore fixé et le but de cette étude n'était pas de le faire, mais de présenter les diverses solutions possibles et de démontrer la faisabilité du prototype PC. De nombreux autres arguments devront être pris en considération avant de faire un choix.

La conception du filtre a été réalisée en gardant à l'esprit cette volonté d'indépendance vis-à-vis de la technologie permettant de repousser au plus tard la décision du choix d'un système ou d'un autre. Elle donne même la possibilité de ne pas avoir à faire un choix unique en faisant cohabiter les différentes solutions envisagées ou qui pourraient l'être plus tard.

C'est également durant cette période qui précède la production du TDR que doit démarrer un travail de modélisation des différents éléments du Triggers DAQ d'ATLAS et notamment du filtre, et cela afin de disposer d'une source d'information la plus réaliste possible sur les caractéristiques finales du système.

## 6.2 PERSPECTIVES

Le filtre d'événements d'ATLAS présente comme caractéristiques principales d'être, d'un point de vue informatique, un système réclamant des ressources de calcul importantes liées avec un besoin également important de flux d'entrée/sortie. A cela s'ajoute également un besoin très important de stockage d'informations.

Ces caractéristiques se retrouvent dans de nombreux autres systèmes faisant appel à l'informatique. On peut bien sûr penser tout de suite aux filtres d'événements des autres détecteurs de physique des particules qui, à quelques détails près, sont similaires en termes de ressources informatiques au filtre d'événements d'ATLAS.

D'autres domaines de recherche scientifique ou même industriels ont également les mêmes besoins. C'est notamment le cas pour les industries qui sont grandes consommatrices de simulations, comme l'aéronautique. C'est vrai également pour des branches de la recherche totalement étrangères à la physique des hautes énergies. On peut citer par exemple la biologie qui, avec l'expansion massive des recherches sur la génétique, réclame des moyens de séquençage et donc de calcul très important, ainsi qu'un besoin de stockage important. Les sciences de la terre, comme la météorologie, ont recours à des méthodes numériques de simulations très complètes et nécessairement très gourmandes en ressources de calcul.

Ces besoins communs, réclamés par une multitude de disciplines disparates, suggèrent naturellement la création d'un outil commun qui serait à même de répondre à toutes ces requêtes. C'est ainsi que depuis quelques années est apparu le concept de grille (GRID)

informatique.

### 6.2.1 LES GRILLES

Les grilles informatiques font naturellement appel à la notion de réseau. On peut d'ailleurs considérer que le WEB est une grille informatique, grille qui dans ce cas peut être qualifiée de grille d'information, étant donné qu'elle est basée sur la notion de partage uniquement de données et de faible volume.

Les grilles dites de calcul qui sont actuellement en train d'être développées apportent plus de fonctionnalités. On y retrouve certes la notion de partage d'information, mais dans ce cas avec des volumes et des vitesses d'accès bien supérieurs à ce que l'on connaît actuellement. On y trouve aussi la notion de partage de ressources de calcul et de stockage. Les objectifs de tels systèmes sont de partager des informations mais aussi de partager des infrastructures informatiques<sup>24</sup> (bases de données, fichiers, périphériques, ...). Ce concept est également connu sous le nom de World Wide Grid.

L'existence de grilles de calcul nécessite le développement de plusieurs éléments :

#### ✓ LES RESEAUX:

Le point essentiel qui relie les applications et qui est le fondement des grilles de calcul est le réseau. Jusqu'à présent les réseaux informatiques permettaient d'échanger des informations de taille modeste. La notion de grilles de calcul suppose que les différents éléments de la grille, fichiers, applications, bases de données, etc... soient accessibles de n'importe où, et cela aussi facilement que si elles étaient locales, ce qui suppose la mise en place de réseaux de très forte bande passante. Les réseaux utilisés doivent avoir une certaine qualité de service afin de permettre notamment la gestion de la grille et la sécurité de données.

---

<sup>24</sup> La référence au terme grille ou Grid vient de l'idée d'une infrastructure où il serait possible de connecter dessus différents modules hétérogènes permettant ainsi de créer à la demande des fonctionnalités à partir de composants existant par ailleurs.

Ces réseaux seront (et sont) basés sur de nouvelles technologies faisant notamment appel à des systèmes de multiplexage de l'information à travers des fibres optiques, mais aussi par l'utilisation de routeurs et de commutateurs de plus en plus performants (commutateurs optiques,...) et intelligents (réseaux actifs).

#### ✓ LA COUCHE INTERFACE UTILISATEUR SYSTEME

Réseaux, calculateurs, unités de stockage, bases de données ne sont que les constituants de la grille, ils nécessitent, pour former un tout, d'être fédérés.

Comment apporter la puissance de calcul, de stockage, les applications, les données, etc... à l'utilisateur? La réponse à cette question passe par la définition et le développement d'une couche logicielle qui va interfacier d'un côté l'utilisateur et ses outils et de l'autre les ressources de la grille; c'est ce que l'on appelle le *Middleware*.

C'est cette interface qui devra connaître l'état des ressources de la grille: quelles sont les machines (avec leurs caractéristiques), les logiciels, les données disponibles, comment y accéder, ...?

C'est également à travers cette interface que l'utilisateur va accéder aux ressources de la grille. Cela suppose une connaissance des différentes autorisations auxquelles l'utilisateur peut prétendre, tant du point de vue accès aux données que d'accès aux différentes machines ou aux distributions d'applications.

### 6.2.2 GRILLES ET PHYSIQUE DES HAUTES ENERGIES

En physique des hautes énergies, les grilles de calcul trouvent leur place auprès de nombreux domaines.

Les simulations, absolument nécessaires pour évaluer les performances des futurs détecteurs, réclament également des ressources informatiques importantes. Ces simulations que l'on qualifie de production, consistent à générer un ensemble d'événements simulés

sur lesquels porteront des analyses. Le besoin important de statistique implique la génération d'un très grand nombre d'événements de différents types, simulation d'événement signal mais aussi d'événement bruit. La puissance de calcul des grilles pourrait être utilisée à de telle fin.

Au niveau des analyses également les grilles de calcul pourraient être utilisées en physiques des particules. Dans ce cas c'est surtout par leur aspect, accès à des volumes important de données, qu'elles seraient intéressantes. Ainsi à travers une grille les physiciens du monde entier pourraient avoir accès aux informations enregistrées par les expériences et cela de façon naturelle.

### 6.2.3 GRILLES ET FILTRE D'ÉVÉNEMENT

Moyennant des caractéristiques, notamment de sécurité et de disponibilité des ressources en terme, de bande passante, puissance de calcul, etc. etc..., les grilles sont également susceptibles d'effectuer des tâches qui correspondraient à une application temps réel. Par temps réel on entend ici temps réel dans un domaine acceptable et non pas temps réel absolu, ce qui est le cas des filtres d'événements.

Les filtres d'événements présentent toutes les caractéristiques pour être déployés sur une grille de calcul :

- ✓ Ils nécessitent de forte puissance de calcul.
- ✓ Ils manipulent de gros volumes d'informations.
- ✓ Ils utilisent des bases de données diverses.
- ✓ Ils doivent fonctionner dans un milieu hétérogène.
- ✓ Ils mettent en oeuvre des applications réclamant de nombreuses ressources.
- ✓ Ils doivent pouvoir être modifiés dynamiquement.
- ✓ Ils doivent pouvoir être monitorés facilement.

On peut très bien imaginer que les noeuds de calcul du filtre d'événements ne seraient finalement que des unités de calcul situées on ne sait où de part le monde, mais faisant partie d'une même grille de calcul.

La physique des hautes énergies est une des disciplines scientifiques les mieux placées pour définir et surtout permettre l'évaluation des systèmes de metacomputing, dont les grilles de calcul. Les filtres d'événements sont emblématiques car ils intègrent la majeure partie des besoins, tant du point de vue des ressources que des fonctionnalités attendues.





## ANNEXE 1: SPECint95

Le SPECint95 est une unité de mesure des performances de calcul sur des nombres entiers, ce qui correspond aux types d'opérations utilisées dans les programmes de reconstruction utilisés en physique des particules. Le programme de tests de SPECint95 comprend une série de huit applications (compression, décompression, manipulation de chaîne, nombres premiers, ...). Il existe un analogue pour les nombres flottants : le SPECfp95. Le tableau ci-dessous donne quelques valeurs de SPECint95 pour quelques machines (cf. <http://www.specbench.org/osg/cpu95/results/cint95.html>).

| <i>ORDINATEUR</i>              | <i>CPU</i>         | <i>SPECTInt95</i> |
|--------------------------------|--------------------|-------------------|
| Dell XPS 100                   | Pentium 100Mhz     | 3,16              |
| Dell Precision Workstation 420 | Pentium III 800MHz | 38,7              |
| Compaq Alphaserver ES40        | Alpha 21264 833Mhz | 50                |
| HP 9000 Model C3600            | PA RISC 552 MHz    | 42,6              |
| AMD                            | Athlon 1 GHz       | 42,9              |

Table 6 : SpectInt95 de quelques ordinateurs



## ANNEXE 2 : Triggers Physiques.

Les évaluations des taux d'événements après passage dans les différents triggers se révèlent extrêmement importantes ne serait-ce que pour évaluer le flot de données nécessaires à stocker, mais aussi pour définir les besoins en termes de taux d'événements nécessaires pour faire ressortir un signal.

Les résultats de telles études sont sans cesse réévaluées. Dans la collaboration ATLAS, une branche (PESA: Physics and Event Selection Strategy) du groupe Trigger/DAQ est en charge de ce travail. Chacun des éléments de triggers physiques, séparation électron/photon, ET miss, b-tagging... et ainsi évalué pour chacun des deux niveaux de triggers (niveau 2 et Event Filter) qui constituent le HLT (High Level trigger).

A titre d'exemple, le tableau suivant montre les performances en termes de taux d'événements correspondant à la séparation de photons et électrons par les triggers de haut niveau. Les efficacités sont données pour un électron de 30 GeV et pour un photon de 60 GeV. Les résultats sont présentés sous forme de séquence. Au niveau du trigger de LVL2, deux types de reconstructions sont présentées : celle basée sur le détecteur TRT et celle basée sur les détecteurs pixels. Le champ *Matching* identifie les résultats pour la combinaison identification du détecteur de traces plus identification du calorimètre.

Le terme m50 représente la limite en temps, sur un ordinateur de type Pentium II 500 MHz (Linux), en dessous de laquelle 50 % des événements sélectionnés demandent moins de temps de traitement, m95 représente la même chose mais pour un nombre d'événements représentant 95 % du lot.

|   | Triggers            | Luminosité nominale |                |                       | Basse luminosité |                |                       |
|---|---------------------|---------------------|----------------|-----------------------|------------------|----------------|-----------------------|
|   |                     | Taux (Hz)           | Efficacité (%) | Temps $m_{50}/m_{95}$ | Taux (Hz)        | Efficacité (%) | Temps $m_{50}/m_{95}$ |
| E<br>L<br>E<br>C<br>T<br>R<br>O<br>N<br>S | LVL2 Calo           | 3490±160            | 97,1±0,3       | 0,2/0,26 ms           | 1100±30          | 96,0±0,6       | 0,15/0,23 ms          |
|   | LVL2 Pre-<br>cision | 620±70              | 90,3±0,6       | 6,2/12,7 ms           | 150±11           | 92,4±0,8       | 2,4/5,8 ms            |
|   | LVL2 TRT            | 1360±100            | 89,7±0,6       | 0,4/1,2 s             | 360±17           | 89,2±0,9       | 31/210 ms             |
|   | LVL2 Mat-<br>ching  | 460±60              | 85,3±0,7       | --                    | 140±11           | 88,1±0,9       | --                    |
|   | EF Calo             | 313±50              | 83,5±0,8       | 0,39/0,63 s           | 85±8             | 86,4±1,0       | 0,34/0,56 s           |
|   | EF ID               | 149±34              | 79,3±0,8       | 11/71 s               | 57±7             | 82,4±1,1       | 0,31/1,6 s            |
|   | EF Matching         | 117±30              | 77,6±0,3       | --                    | 41±6             | 80,8±1,2       | --                    |
| P<br>H<br>O<br>T<br>O<br>N<br>S           | L2 Calo             | 250±43              | 97,7±0,4       | 0,2/0,26 ms           | 83±8             | 94,0±0,3       | 0,2/0,26 ms           |
|   | EF Calo             | 144±33              | 87,3±0,9       | 0,39/0,63 s           | 57±7             | 84,7±0,6       | 0,34/0,56 s           |
|   | EF ID               | 114±43              | 82,9±0,9       | 19/106 s              | 51±7             | 81,2±0,6       | 0,44/1,2 s            |

Table 7 : Taux d'événements évalués au niveau des triggers de haut de niveau d'ATLAS pour une isolation d'électrons et de photons.

Le paramètre temporel donné ici permet de se référer au temps alloué à chacun des niveaux (LVL2 et EF) pour effectuer sa sélection.

## GLOSSAIRE

**ATLAS** : A Toroïdal LHC Apparatus. L'un des quatre détecteurs mis en place autour de l'accélérateur de particules LHC, qui doit rentrer en fonctionnement en 2005 au CERN.

**ATM** : Asynchronous Time Division : Technologie de réseau permettant le transfert en mode asynchrone.

**CERN** : Centre Européen de Recherche Nucléaire: Centre de recherche international situé à Genève qui regroupe une grande communauté de physiciens et ingénieurs travaillant dans le domaine des physiques nucléaire et de la physique des particules.

**CPPM** : Centre de Physique de Particules de Marseille.

**CORBA** : Common Object Request Broker Architecture: Protocole de communication entre objets distribués.

**CSC** : Cathode Strip Chamber : Chambre à muon de précision se situant dans la partie centrale du spectromètre à muons

**CTP** : Central Trigger Processor : Élément recevant les informations du système de trigger de premier niveau.

**DAQ** : Data Acquisition: système d'acquisition de données.

**DCS** : Detector Control System: Système ayant en charge l'acquisition et la gestion de tout ce qui concerne les détecteurs, exceptés des données liées à l'événement. C'est lui qui en charge l'approvisionnement en gaz, électricité, froid,... de tous les sous-détecteurs et qui s'assure de leur bon fonctionnement.

**DFM** : Dataflow Manager : Élément qui a en charge la gestion du flot de données dans le constructeur d'événement.

**DGB** : Distributor Global Buffer : Système de sauvegarde se situant au point d'entrée du filtre et permettant de garder une copie de chaque événement envoyé dans le filtre, s'as-

surant ainsi de la non perte d'événement au cours du traitement.

**DMA** : Direct Memory Access : mécanisme d'accès direct à la mémoire des périphériques de données tels que des disques durs ou des cartes réseaux.

**EB** : Event Builder: Constructeur d'événement : Élément de la chaîne d'acquisition qui assemble les différentes parties de l'information appartenant à un même événement.

**EF** : Event Filter: filtre d'événements: C'est le troisième et dernier niveau de sélection du système d'acquisition. Il comprend une partie de gestion de flot de données et une partie de sélection.

**EM** : Electro-magnétique : Désigne le calorimètre électromagnétique spécialement développé pour déterminer l'énergie des électrons et photons.

**FE** : Front End : Nom générique des interfaces de sortie des différents sous-détecteurs.

**FEL** : Front End Link : Liens optiques ou électriques reliant les interfaces de sortie des détecteurs aux ROD.

**FIFO** : First In First Out : File d'attente de type premier entré premier sorti.

**FPGA**: Field Programmable Gate Array: Composants électroniques constitués de portes logiques pouvant être activées ou pas par création de champs électriques, permettant ainsi de créer des réseaux logiques qui assurent des fonctions de traitement d'information extrêmement rapidement.

**GRID** : Grille : Nom générique des grilles de calcul informatique.

**GUI** : Graphic User Interface: Interface graphique d'utilisation: Désigne une interface graphique par laquelle on reçoit des informations et/ou on agit sur un système.

**HLT** : High Level Triggers: Désignation générique du trigger de second niveau et du filtre d'événements.

**I/O** : E/S : Désigne les entrées et sortie des systèmes auxquels ils sont rattachés.

**LHC** : Large Hadrons Collider: Grand collisionneur Hadronique: Futur accélérateur de particules hadroniques mis en chantier au CERN.

**LVL1** : Level 1 : Désignation du système de sélection de premier niveau.

**LVL2** : Level 2 : Désignation du système de sélection de second niveau.

**MDT** : Monitored Drift Chamber : Chambre à muons, prévue pour des mesures de grande précision dans la zone à grand  $\eta$ .

**MPI** : Message Passing Interface : Ensemble de fonctions normalisées de communication supportant différents protocoles et types de communication.

**Myrinet** : Technologie de réseau , correspondant à une évolution de Ethernet.

**NS** : Naming service: Serveur de Nom.

**NUMA** : Non Uniform Memory Access : Ordinateur de type SMP dont seulement une partie de la mémoire est commune aux différents processeurs.

**PC** : Personal Computer: Ordinateur personnel: Désigne l'ensemble des ordinateurs dits familiaux.

**PCI** : Peripheral Component Interconnect : Bus d'interconnexion de périphériques parmi les plus répandus dans l'informatique.

**PT** : Processing Task : tâche d'analyse: Représente l'ensemble des applications de reconstruction et de sélection qui tourneront sur le filtre.

**PVM** : Parallel Virtual Machine : Ensemble logiciel permettant de fédérer un ensemble de machines relié par réseau comme si c'était une seule machine parallèle.

**ROB** : Read Out Buffer : Éléments de stockage d'information au sein de la chaîne d'acquisition d'ATLAS. Les ROB sont essentiellement constitués de mémoires tampons permettant de stocker l'information mais également d'une partie permettant leur contrôle.



Les **ROB** sont susceptibles de recevoir de l'information mais aussi de la distribuer sur demande.

**ROD** : Read Out Driver: Éléments de stockage constitués de mémoires tampons. Contrairement aux **ROB** qui sont tous identiques les **ROD** sont spécifiques à chaque sous-détecteurs.

**ROI** : Region Of Interest : Zones du détecteur susceptibles de contenir de l'information utile. Les **ROI** sont déterminées par le trigger de premier niveau.

**ROS** : Read Out System : Système regroupant différents **ROB** et leurs interfaces de communication.

**RPC** : Resistive Plate Chamber : Chambre à muons servant essentiellement pour le système de trigger dans la partie centrale du détecteur

**SCADA** : Supervisory Control And Data Acquisition system: Nom du DCS d'ATLAS.

**SCT** : Semi Conductor Tracker : Détecteur à base de semi conducteur déployé sous forme de ruban, permettant de définir des points de passage de particules chargées.

**SFI** : Sub Farm Input : Point d'entrée du filtre d'événements.

**SFO** : Sub Farm Output : Point de sortie du filtre d'événements.

**SMP** : Symmetric Multi-Processor : Ordinateur dont l'architecture est basée sur la mise en parallèle de plusieurs processeurs ayant accès à une mémoire commune.

**SQL** : Structured Query Language : Langage de spécification de requêtes pour des bases de données relationnelles.

**TCP/IP** : Transmission Control Protocol/Internet Protocol : Protocole de communication le plus répandu sur les réseaux de type Ethernet. Actuellement la version du protocole est connue sous le nom de IPv4 et devra être remplacé d'ici peu par l'IPv6.

**TGC** : Thin Gap Chamber : Chambre à muons servant essentiellement pour le système

de trigger dans la zone à grand  $\eta$ .

**TRIGGERS** : Ensemble de sélections. Peut être utilisé aussi bien pour désigner des systèmes de sélection que les critères de sélections eux-mêmes. On parle des triggers pour désigner les différents étages de sélection de la chaîne d'acquisition mais aussi de triggers physiques pour désigner les sélections qui portent sur la physique contenue dans l'événement.

**TRT** : Transition Radiation Tracker: détecteur par transition de radiation: Un des sous-détecteur du détecteur interne d'ATLAS. Il permet notamment la détermination des traces des particules.

**TTC** : Timing, Triggers and Control : Signal d'horloge servant de référence à l'ensemble des systèmes liés au LHC.

**UMA** : Uniform Memory Access : Ordinateur de type SMP, dont les différents processeurs ont la même priorité d'accès à l'ensemble de la mémoire.

**WAP** : Wireless Application Protocol : Protocole de communication destiné en particulier aux systèmes embarqués. Désigne de façon courante les appareils susceptibles d'utiliser ce protocole comme certains mobiles téléphoniques.

**XML** : eXtended Markup Language : Format de fichiers basé sur l'utilisation de balises et descendant de SGML.



## Bibliographie

- [1] : ALEPH Collaboration, Observation of an excess in the search for the standard model Higgs boson, 11/2000
- [2] : DELPHI Collaboration, Search for the standar Higgs model boson at LEP in the year 2000, 12/2000
- [3] : ATLAS Collaboration, ATLAS Detector and Physics performance, 05/1999
- [4] : Mommsen, R et al, ATL-DAQ-2000-007, Performance Studies for Electron and Photon Selection at the Event Filter, 02/2000
- [5] : Polesello, G et al, ATL-DAQ-2000-016, Event Filter rates for the etmiss+jets trigger at low luminosity, 03/2000
- [6] : Wielers, M et al, ATL-DAQ-2000-015, Performance Studies of Jets in the High Level Trigger, 02/2000
- [7] : Baines, JTM et al, ATL-DAQ-2000-031, B-Physics Event Selection for the ATLAS High Level Trigger, 02/2000
- [8] : ATLAS Collaboration, ATLAS Technical Proposal, 12/1994
- [9] : D0 Collaboration, Phys Rev Lett, Phys Rev Lett 74 2422, 1995
- [10] : CDF Collaboration, Phys Rev, Phys Rev D50 2966 , D51 4623, 1994
- [11] : ALICE Collaboration, CERN/LHCC/95-71, A Large Ion Collider Experiment -Technical Proposal, 12/1995
- [12] : LHCb Collaboration, CERN/LHCC/98, LHCb Technical Proposal, 02/1998
- [13] : CMS Collaboration, CERN/LHCC/, CMS Technical Proposal, 12/1994
- [8] : ATLAS Collaboration, ATLAS Technical Proposal, 12/1994
- [15] : ATLAS HLT/DAQ/DCS Group, CERN/LHCC/2000/17, ATLAS High Level trigger, DAQ,DCS Technical Proposal, 03/2000
- [16] : ATLAS Level 2 Group, ATL-DAQ-2000-040, Results from the Level 2 Pilot Project Testbed, 03/2000
- [15] : ATLAS HLT/DAQ/DCS Group, CERN/LHCC/2000/17, ATLAS High Level trigger, DAQ,DCS Technical Proposal, 03/2000

- [18] : Crone, G et al, ATL-DAQ-2000-053, The Read-Out-Buffer in DAQ/EF Prototype -1, 12/1999
- [19] : Ambrosini, G et al, ATL-DAQ-2000-048, Summary Document of the Event Building, 03/2000
- [20] : Burckhart, D et al, ATL-DAQ-2000-001, Back-End Summary Document, 12/1999
- [21] : Beck, HP et al, ATL-DAQ-2000-005, Event Filter Summary Document, 12/2000
- [22] : Vercesi, V et al, ATLAS DAQ -1 TECHNICAL NOTE 128, Detailed Design and Implementation of a SMP Event Filter Sub Farm, 06/1999
- [23] : Meessen, C et al, ATL-DAQ-2001-001, Event Filter Dataflow Software, 12/2000
- [24] : Meessen, C et al, ATLAS DAQ -1 TECHNICAL NOTE 132, Event filter farm component implementation design Version 3, 06/1999
- [25] : Qian, Z et al, ATL-DAQ-2001-002, PC-based Event Filter supervisor: Design and Implementation, 12/2000
- [26] : Fede, E et al, ATLAS DAQ -1 TECHNICAL NOTE 129, First Results of Offline Code Benchmarking in the ATLAS Event Filter Context, 06/1999