

Integrando Plataformas e Algoritmos para o Segundo Nível de Trigger do Experimento ATLAS

André Rabello dos Anjos e José Manoel de Seixas

COPPE/EE/UFRJ - CP68054, CEP21945-970, Rio de Janeiro, Brasil
Email: Andre.dos.Anjos@cern.ch, seixas@lps.ufrj.br

Resumo

No ano de 2005 entrará em funcionamento o experimento ATLAS, que pretende verificar interações a 16 Tev. A física de interesse é rara e o número de eventos gerados é enorme. Um sistema de filtragem será, então, responsável pela separação da física de interesse da física ordinária. Este sistema é dividido em níveis de processamento. No segundo nível de filtragem estão sendo investigados vários algoritmos e plataformas. Aqui apresentamos desenvolvimentos para o Software de Referência, que visa integrar os estudos de algoritmos com realização tecnológica neste nível de filtragem.

1. Introdução

Em 2005 entrará em funcionamento o detetor ATLAS (do inglês, A Toroidal ApparatuS) que investigará colisões próton-próton a 16 Tev. O detetor ATLAS (vide Figura 1) visa registrar o bóson de Higgs e medir a sua massa com precisão.

A previsão para este bóson é que seja extremamente instável, decaindo rapidamente em partículas de menor energia e mais pesadas, formando, possivelmente, uma cascata eletromagnética. Além disso, como o alvo do experimento é extremamente raro, cerca de quarenta milhões de eventos serão gerados por segundo. Isto faz com que as medições do experimento possam ser livres de tendências introduzidas por uma estatística pobre.

A taxa de dados prevista para o sistema está na ordem de *1 Terabyte por segundo*, o que corresponde a uma taxa de dados impraticável para qualquer sistema atual de armazenamento permanente.

Para reduzir esta taxa de dados, a aquisição de dados do ATLAS contará com um sistema de filtragem que fará uma redução online na taxa de eventos, separando a física ordinária da física de interesse [1]. Este sistema será discutido na próxima seção (seção 2). A seção 3 descreve o software de referência, um workframe para o segundo nível de filtragem do ATLAS. Conclusões e extensões deste trabalho são abordadas na seção 4.

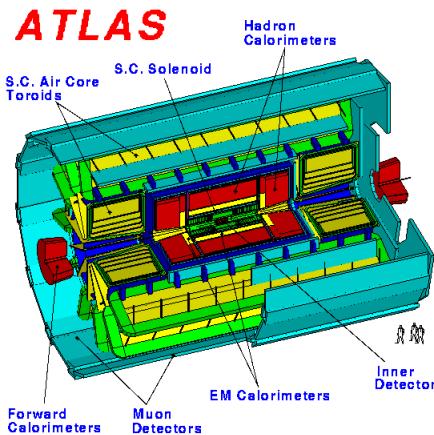


Figura 1. O detetor ATLAS e seus sub-detetores. Repare, em escala e no canto inferior direito, o desenho de pessoas.

2. O sistema de filtragem de eventos do ATLAS (trigger)

Para poder reduzir a taxa de eventos e, consequentemente, o fluxo de dados a serem gravados em mídia permanente, a aquisição de

dados do ATLAS possui um complexo sistema de filtragem. Este sistema é constituído por 3 níveis de complexidade crescente e velocidade de processamento decrescente. A técnica visa cobrir a taxa inicial de eventos (40 MHz) a reduzindo-a a não mais que 100 eventos por segundo.

2.1. O segundo nível de filtragem

O segundo nível de filtragem (LVL2) do ATLAS utilizará processamento local para reduzir a taxa original de eventos de 100KHz para não mais que 1KHz. Este processamento é baseado no conceito de regiões de interesse [2]. As regiões de interesse (do inglês Regions of Interest, ou ROI) são as coordenadas no plano $\eta \times \phi$ de onde há informação interessante, uma vez que nem todas as áreas dos detetores se encontram excitadas num dado evento. O LVL2 recebe do primeiro nível de filtragem tais coordenadas e a indicativa do canal físico pelo qual o evento foi validado. O sistema então, dependendo da estratégia de processamento, pode carregar as informações de todas as ROI's ou analisar seqüencialmente todas as ROI do evento. Também pode optar por carregar os dados de todos os detetores para uma ROI ou somente parte dos dados. Estas estratégias estão sendo discutidas no âmbito da colaboração PESA (Physics and Event Selection Algorithms) [3].

Embora as estratégias de análise estejam sendo discutidas, o modelo de processamento a ser seguido é simples:

1. Para cada região de interesse são extraídas características segundo um ou mais detetores. Estas características abrangem energia transversa, momento, e outras;
 2. A partir das características extraídas é possível identificar a partícula que excitou os detetores neste ponto;
 3. Ao identificar todas as partículas, compara-se o evento com uma tabela contendo os possíveis decaimentos de Higgs a serem aceitos pelo LVL2. Se houver um casamento entre o observado e o que se encontra tabelado, o evento é repassado ao 3º nível de filtragem, de outra forma, é deletado dos buffers.

Um gráfico explicativo, mostrando o relacionamento do segundo nível de filtragem com os demais níveis, assim como o fluxo de dados no LVL2 pode ser vista na Figura 2.

Durante a sua operação no LHC, o segundo nível estará funcionando sobre uma rede de computadores interconectada por um rápido sistema de rede. As tecnologias de rede, plataforma computacional e sistema operacional também estão sendo escolhidas.

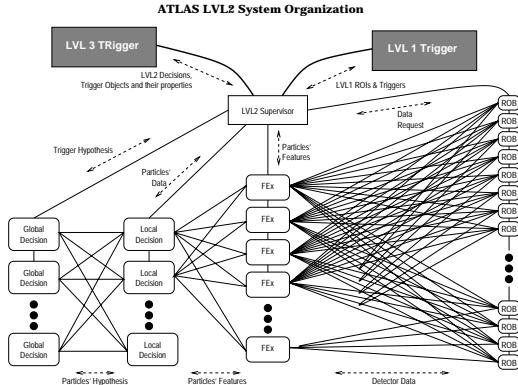


Figura 2. O segundo nível de filtragem do ATLAS visto graficamente.

The Reference Software Structure

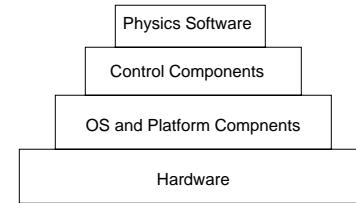


Figura 3. O empilhamento de sistemas para os Programas de Referência. Os elementos na parte de cima da pirâmide são construídos usando os elementos abaixo deles.

3. Os Programas de Referência

Os programas de referência foram criados para integrar o estudo de algoritmos (supervisores, extratores de característica e decisores locais) ao estudo de tecnologias para o segundo nível. Isto é possível graças às técnicas de orientação a objetos, que permitem que uma transparência de operação seja alcançada sem que sejam necessárias mudanças na sintaxe dos algoritmos do segundo nível.

Para atingir tal efeito, é necessário organizar os programas em camadas, como é feito com protocolos de comunicação [4]. No caso dos Programas de Referência, esta pilha será dada como na Figura 3 [5]. É possível, desta forma, isolar os algoritmos da plataforma sendo utilizada.

Para que tal efeito fosse atingido, foi necessário que as interfaces de cada elemento do conjunto fossem cautelosamente projetadas. Este zelo previne re-estruturações que possam vir a intervir no modelo de funcionamento estipulado [6].

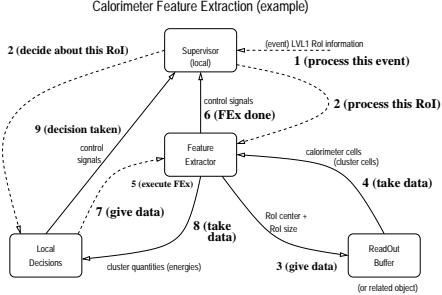


Figura 4. Uma análise gráfica da extração de características.

3.1. Extração de Características de RoI's no Calorímetro

Como exemplo, averiguamos o funcionamento da fase de extração de características para regiões de interesse. Mais especificamente, estaremos olhando para a extração de características para os calorímetros do ATLAS, embora o modelo possa ser aplicado à qualquer sub-sistema de extração de características no segundo nível de filtragem do experimento.

É possível modelar tal extração de características a partir dos seguintes passos (veja Figura 4) [11, 12]:

1. Um extrator de características é alocado, pelo supervisor, para processar uma RoI;
2. O extrator (que encontra-se em um nó de processamento qualquer hipoteticamente) faz o pedido dos dados a serem processados para uma RoI que foi indicada pelo supervisor. Neste caso, são transmitidos às memórias que armazenam a informação dos calorímetros (ROB's, do inglês Read Out Buffers) o centro da RoI, seu tamanho e um identificador do evento;
3. O extrator recebe tais dados e realiza o processamento, extraíndo as características da RoI;
4. Estas características são repassadas a um decisor local que, possivelmente recebendo dados sobre a mesma RoI de outros extratores (para os detetores de traços TRT e SCT ou detetores de muôns), decide sobre a natureza do evento na RoI.

A modelagem orientada a objetos pode ser extraída deste modelo e está denotada na Figura 5.

Esta figura mostra as relações entre os objetos do segundo nível. Note-se que aqui, o supervisor não está sendo modelado. Ainda, é possível reparar que o algoritmo do extrator está desacoplado do

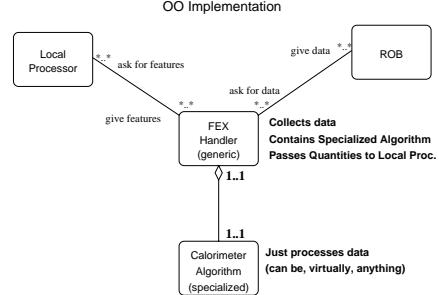


Figura 5. Uma modelagem orientada a objetos da Figura 4. Repare que aqui o supervisor não é modelado. O esquema segue a Unified Modelling Language (UML).

mesmo. Sua união é feita durante a compilação para uma plataforma seguindo opções de *linkagem*. Em outras palavras, o algoritmo somente conforma com o sistema pois conforma com a interface proposta pelo extrator, que na figura funciona como um simples manipulador de dados. É claro que o sistema final não é tão simples, pois em muitas vezes se faz necessário o uso de travas de memória (mutexes) ou semáforos. Estes mecanismos são encapsulados pelo módulo que recobre o sistema operacional. A implementação da interface do objeto *Algoritmo de Extração* é vista a seguir:

```
template<class FT, class DT, class UD>
class BaseAlgorithm
public:
    // definições a serem usadas por
    // outros templates e classes.
    typedef FT Feature;
    typedef DT Data;
    typedef UD UserData;

public:
    BaseAlgorithm() ;
    virtual ~BaseAlgorithm() ;

public:
    // Extract features from Data.
    virtual Feature *extract(LVL1Id id,
        const Region& region,
        const Data& data,
        UserData *param = 0) = 0;

    // Create new instance of this algorithm.
    virtual BaseAlgorithm<FEATURE,
        DATA,USERDATA> *clone() = 0;
;
```

Observe que o método `extract()` deve ser definido como **virtual** e nenhuma implementação deve estar contida nesta interface. A declaração do protótipo deste método a relação entre o módulo manipulador de dados (aqui chamando de FEx Handler) e o módulo que define o algoritmo. Percebe-se que o módulo manipulador de dados (FEx Handler) desconhece o algoritmo com que trabalha. Isto é possível graças às técnicas de orientação a objetos propostas anteriormente, e é implementável usando-se C++.

4. Conclusões

O segundo nível de filtragem (LVL2) do sistema ATLAS foi apresentado e discutido. Uma análise orientada a objetos (OO) foi apresentada como exemplo para um sub-sistema do LVL2, o extrator de características para os calorímetros. A partir desta análise foi realizado um projeto OO e uma implementação em C++ se tornou possível. Particularmente, para os extratores de características, observou-se que é possível desconectar estudos de algoritmos ao estudo tecnológico, ainda que provendo um ambiente de trabalho comum a todos os sub-sistemas.

Este ambiente está sendo utilizado atualmente no desenvolvimento de todos os sub-sistemas do LVL2, que deverá estar pronto em 2005, junto com o restante do sistema ATLAS. Ele permite que os estudos em diversos países e centros possa ser comparado a um padrão de funcionamento, fornecendo ainda uma ponte de integração entre diversas plataformas/sistemas operacionais e os algoritmos no segundo nível de filtragem do ATLAS.

O sistema do calorímetro foi implementado [7] e não houve diferenças marcantes nos tempos de processamentos anteriormente obtidos (com programas escritos em C) e os valores atuais, para a mesma eficiência na seleção de canais físicos.

No ambiente dos programas de referência, testes com a tecnologia dos processadores digitais de sinais (DSP's) rápidos, como processadores atuantes na extração de variáveis, e com redes neurais artificiais (ANN's), como alternativa de algoritmo estão sendo consideradas. A flexibilidade dos programas de referência permite estas implementações.

Referências

- [1] First-Level Trigger Technical Design Report, ATLAS Level-1 Trigger Group, CERN (1998).
- [2] ATLAS DAQ, EF, LVL2 and DCS Technical Progress Report, ATLAS Collaboration, CERN (1998).
- [3] <http://atlasinfo.cern.ch/Atlas/GROUPS/-DAQTRIG/PESA/pesa.html>
- [4] Internet Working with TCP/IP (Volume 1), Douglas E. Comer, Prentice Hall, second edition (1991).
- [5] Software Framework and Testbed Architectures, LVL-2 Reference Software Group, CERN, Testbed Technical Note 1, CERN (1998).
- [6] Object Oriented Analysis and Design with Applications, G. Booch, Addison and Wesley (1994).
- [7] <http://www.cern.ch/Atlas/project/-LVL2testbed/www/doc/physics/calo/-index.html>
- [8] A Neural Online Triggering System based on Parallel Processing, A. dos Anjos *et al.*, IEEE transactions on Nuclear Science, volume 45, number 4, part I (August 1998).
- [9] Study of the FEX subfarms with C40s and Digital Memory Channel Cluster, B. Kastrup *et al.*, CERN, ATLAS DAQ Internal Note 68 (1997).
- [10] What Can Artificial Neural Networks Do for the Global Second Level Trigger, R. Bock *et al.*, CERN, ATLAS DAQ Note 11 (1994).
- [11] Requirements, Constraints, Use Cases and an Object-Oriented Analysis of the Steering and Feature Extraction Systems for LVL-2, A. dos Anjos *et al.*, CERN, Testbed Technical Note 8 (1998).
- [12] Design of the Feature Extraction System, A. dos Anjos *et al.*, CERN, Testbed Technical Note 20 (1998).