

# FREE

# HELP

<http://java.freehep.org>



- Qu'est-ce que c'est ?
- Qu'y a-t-il dedans ?
- Où est-il employé ?
- Comment est-il développé ?

# Mission

Le but de la bibliothèque de FreeHEP est d'encourager le partage et la réutilisation du code de Java dans la physique de particules.

Bien qu'une partie du code soit assez spécifique à HEP, l'autre code est plus générique et pourrait être employé par n'importe qui.

Pour maximiser la réutilisation que nous essayons de maintenir les dépendances entre de divers paquets dans la bibliothèque de FreeHEP au minimum, ainsi vous pouvez employer n'importe quelles pièces qui vous intéressent sans être forcé d'employer la bibliothèque entière.

# Qu'y a-t-il dedans ?



## Components

HEP	HEP IO	HepRep	JAIDA	Math	Tools	Vector Graphics	Maven Plugins
<ul style="list-style-type: none"><li>● <a href="#">Java3D</a></li><li>● <a href="#">Physics</a></li><li>● <a href="#">YaPPI</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">HBook</a></li><li>● <a href="#">MCFIO</a></li><li>● <a href="#">ROOT IO</a></li><li>● <a href="#">StdHEP</a></li><li>● <a href="#">XDR</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">HepRep</a></li><li>● <a href="#">HepRep1</a></li><li>● <a href="#">JHepRep</a></li><li>● <a href="#">JHepRep1</a></li><li>● <a href="#">HR1-Adapter</a></li><li>● <a href="#">CHepRep</a></li><li>● <a href="#">HepRepJNI</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">JAida</a></li><li>● <a href="#">Remote</a></li><li>● <a href="#">FMinuit</a></li><li>● <a href="#">JMinuit</a></li><li>● <a href="#">HBook</a></li><li>● <a href="#">Root</a></li><li>● <a href="#">AIDAJNI</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">JMinuit</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">AID</a></li><li>● <a href="#">AID-Plugin</a></li><li>● <a href="#">RTTI</a></li><li>● <a href="#">PSViewer</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">Graphics2D</a></li><li>● <a href="#">GraphicsIO</a></li></ul> <p><b>Formats</b></p> <ul style="list-style-type: none"><li>● <a href="#">CGM</a></li><li>● <a href="#">EMF</a></li><li>● <a href="#">LaTeX</a></li><li>● <a href="#">Java</a></li><li>● <a href="#">PDF</a></li><li>● <a href="#">PostScript</a></li><li>● <a href="#">SVG</a></li><li>● <a href="#">SWF</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">CharTable</a></li><li>● <a href="#">Filter</a></li><li>● <a href="#">IDLJ</a></li><li>● <a href="#">JarJar</a></li><li>● <a href="#">JAS</a></li><li>● <a href="#">NAR</a></li><li>● <a href="#">One</a></li><li>● <a href="#">RMIC</a></li></ul>

## Base

<ul style="list-style-type: none"><li>● <a href="#">Application Framework</a></li><li>● <a href="#">ArgV</a></li><li>● <a href="#">Command Dispatcher</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">Command Line</a></li><li>● <a href="#">Export</a></li><li>● <a href="#">I/O</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">Parameter Database</a></li><li>● <a href="#">Record</a></li><li>● <a href="#">Swing</a></li></ul>	<ul style="list-style-type: none"><li>● <a href="#">Util</a></li><li>● <a href="#">Web Util</a></li><li>● <a href="#">XML</a></li></ul>
---	---	---	---

# Application Framework

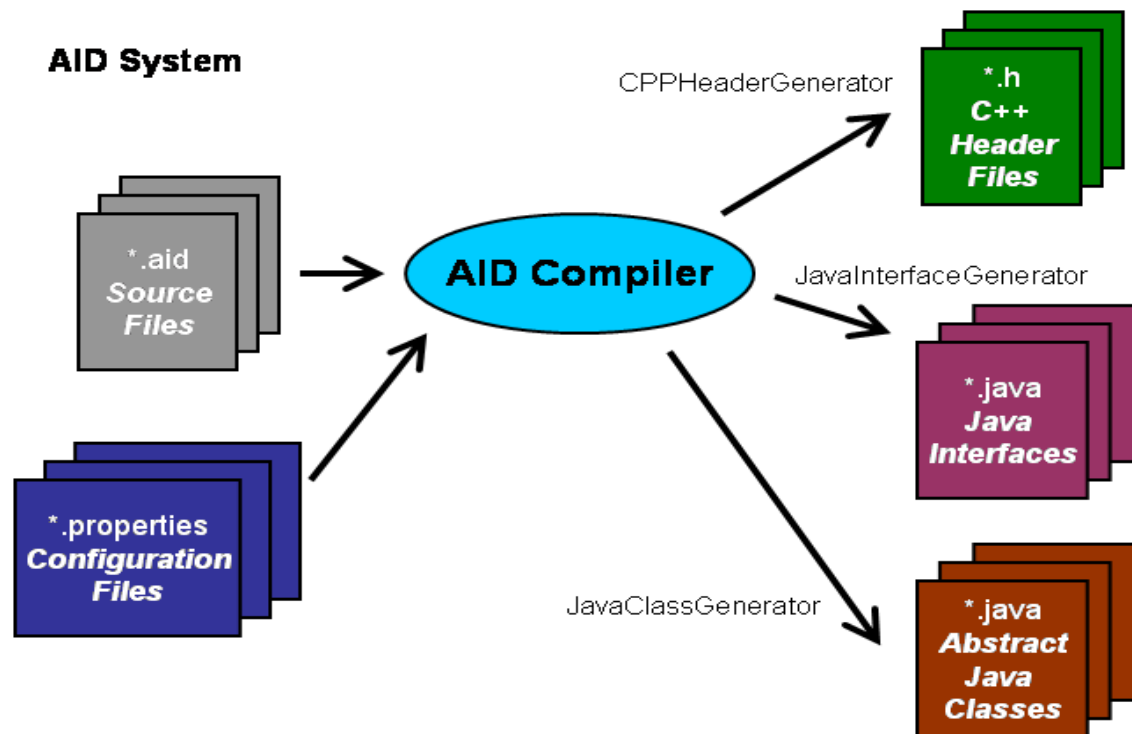


- Ensemble de classes de framework pour construire des applications.
- 4 Couches:
  - Application
  - MDIApplication
  - Studio
  - DataStudio
- Utilisé par JAS.

# AID

## Abstract Interface Definitions compiler

- Outil pour produire des interfaces de langues et des classes spécifiques (Java, C++...) à partir des fichiers source abstraits de définition d'interface (aid).
- Utilisé par AIDA et HEPRep.



# AIDA

## Abstract Interfaces for Data Analysis



- Les buts du projet d'AIDA sont à définir les interfaces abstraites pour les objets communs d'analyse de physique, tels que les histogrammes, les ntuples, les fitters, le E/S etc.. L'adoption de ces interfaces devrait faciliter aux physiciens l'utilisation des outils différents sans avoir besoin d'apprendre de nouvelles interfaces ou changer tout leur code. Les avantages additionnelles seront l'interopérabilité des applications conformes d'AIDA (par exemple en permettant des applications d'échanger des objets d'analyse par l'intermédiaire de XML).
- AIDA dans FreeHEP:
  - JAIDA: implémentation native en Java
  - AIDA-JNI: C++ wrapper of JAIDA
- Autres réalisations:
  - Open Scientist (LAL): C++
  - PI (CERN): C++
  - PAIDA: Python

# AIDA TLD

## Web Service access to AIDA

- La bibliothèque de JSP (Java Server Pages) permet d'accéder à AIDA en pages standards de Service de Web.
- Cela fonctionne avec des données accessibles par AIDA (les bases de données SQL, les fichiers Root...).

**MULTIPLICITY - WEIGHTED**

Entries : 110908
OutOfRange : 0
Mean : 10.079
Rms : 4.2559

**TEST OF HBOOK2**

Entries : 5000
OutOfRange : 0
Mean : 0.48060
Rms : 0.24781
Mean : 0.48094
Rms : 0.24768

Image: Width: 600 Height: 300 Update

Download: [pdf](#) [svg](#) [eps](#) [swf](#) [png](#) [gif](#) [jpg](#)

```
IAnalysisFactory = class hep.aida.IAnalysisFactory
af = IAnalysisFactory::create()

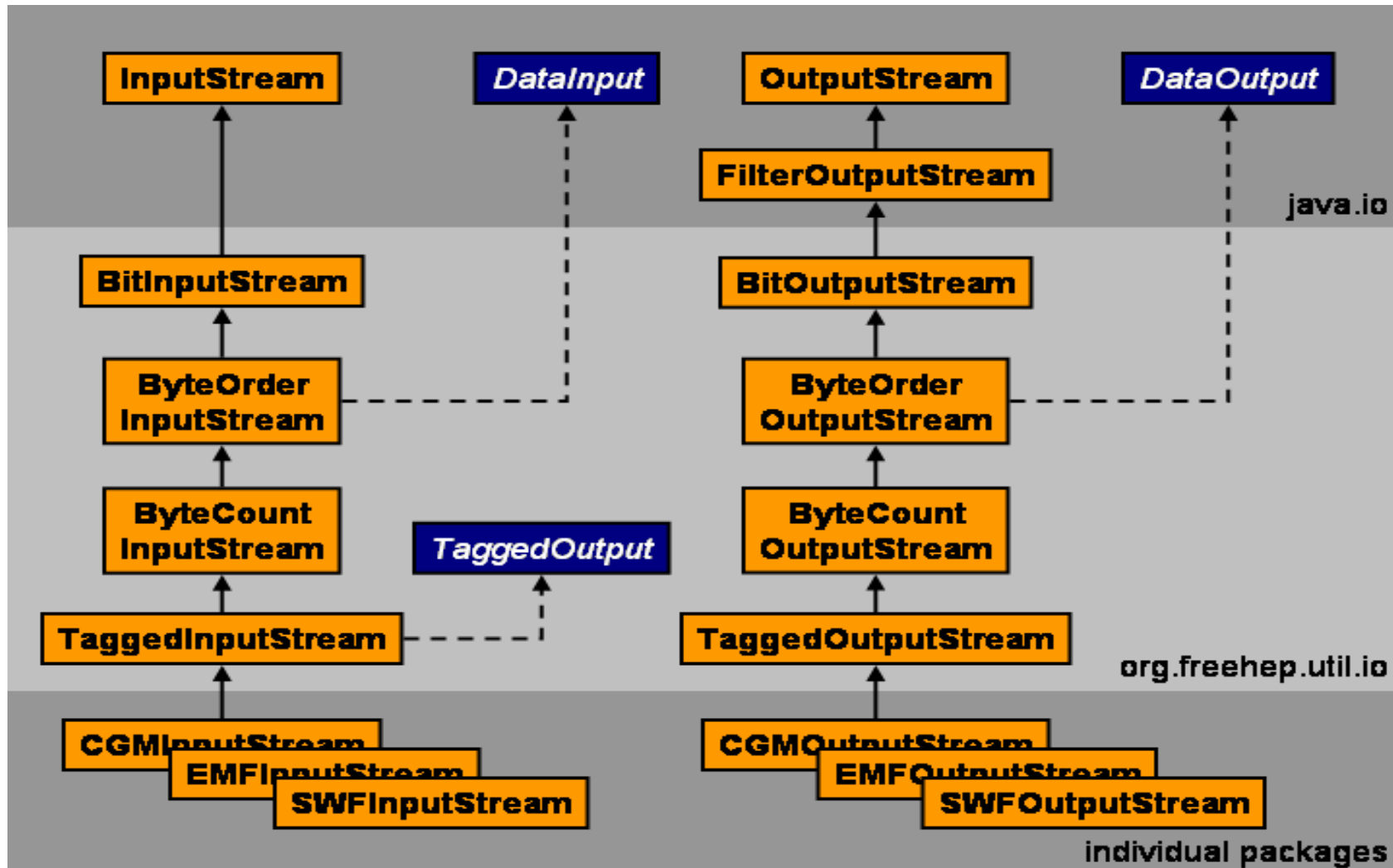
h100 = aidaMasterTree.find("/h100")
h210 = aidaMasterTree.find("/h210")

plotter = af.createPlotterFactory().create()
plotter.createRegions(2, 1)
plotter.region(0).plot(h100)
plotter.region(1).plot(h210)
```

Execute Save As MyPlots

# Generic IO

- Extensions de Java standard E/S.

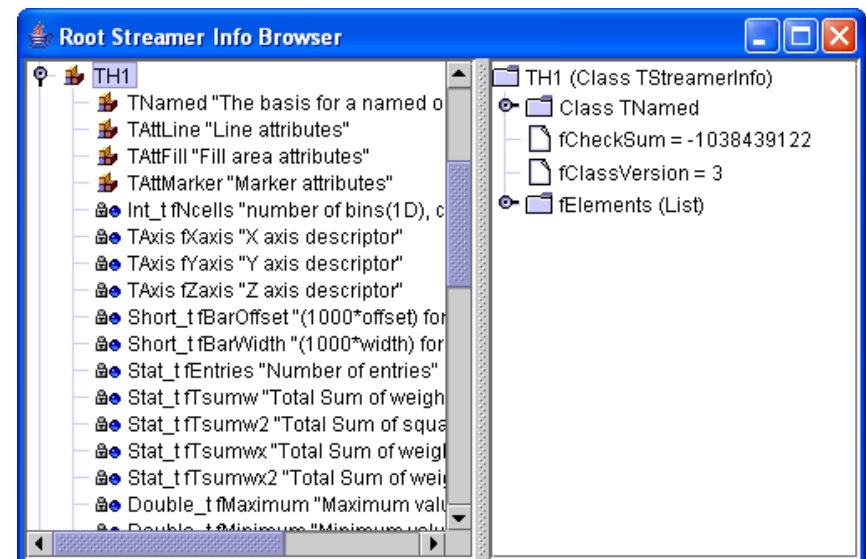
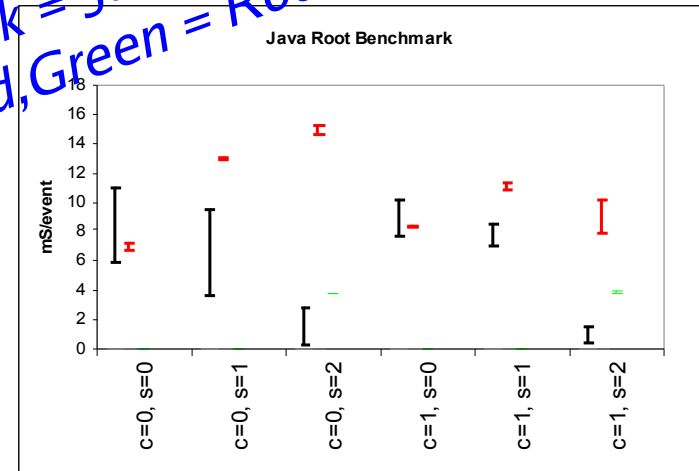




# Root IO

- Implémentation pure Java de Root IO.
- La plupart des classes produites dynamiquement de StreamInfo.
- Accès au serveur Rootd.
- Interprète des objets d'analyse de Root comme des objets d'AIDA.
- L'écriture des fichiers Root dans un prototype autonome.
- Problèmes avec des fichiers Root non standard.

Black = Java  
Red, Green = Root



# Vector Graphics

- Exportation des données graphiques vers beaucoup de formats de graphiques de vecteurs et formats bitmaps de graphiques:
  - Formats de vecteurs : PostScript, PDF, EMF, SVF, SWF, CGM,...
  - Formats bitmaps : GIF, PNG, JPG, PPM,...

# YaPPI

## Yet Another Particle Property Interface



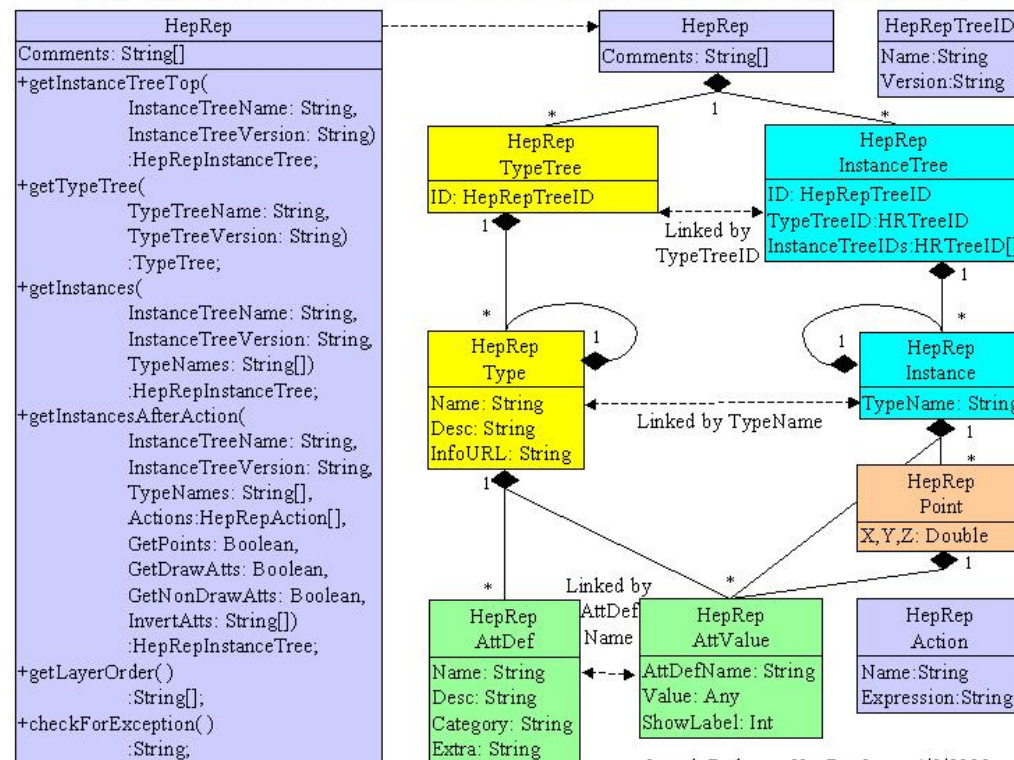
- Une base de données de XML de propriétés de particules.
- Basée sur les données PDG.
- Java API.
- L'interface Web.

# HEPRep

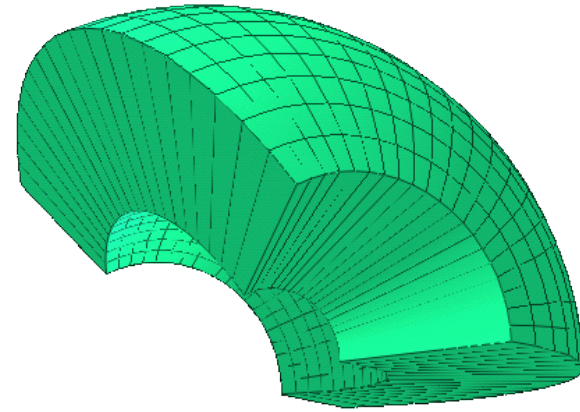
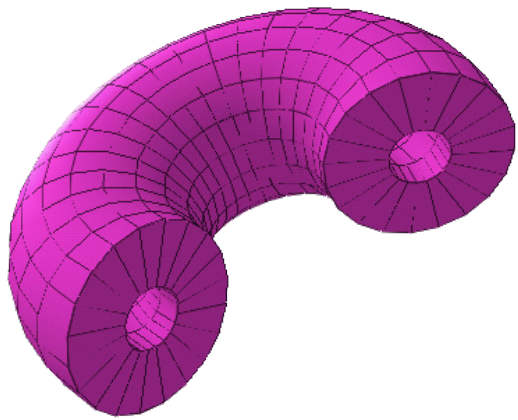
## Generic Interface for Event Display

- Client-Serveur API (Java, C++).
- XML IO.
- Utilisé par : WIRED, Fred, Geant4.

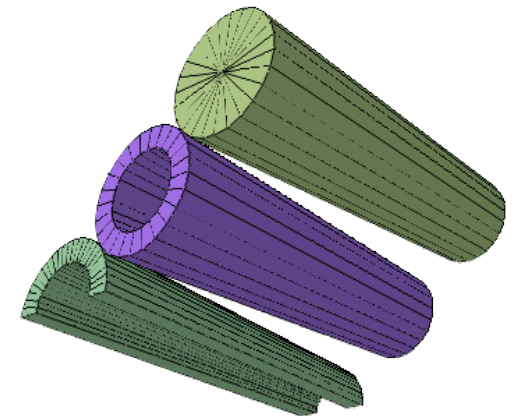
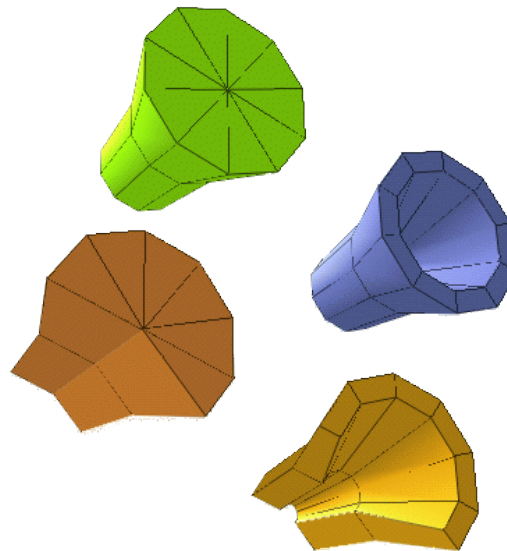
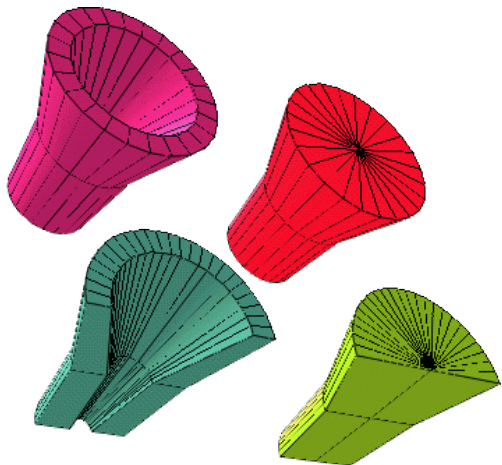
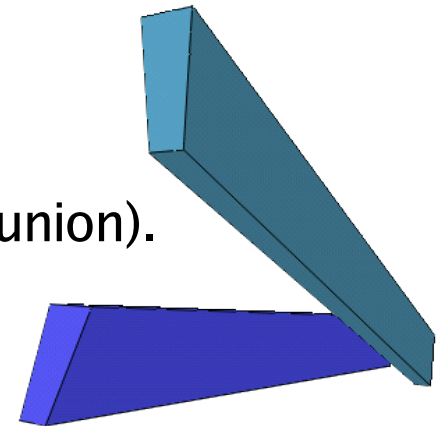
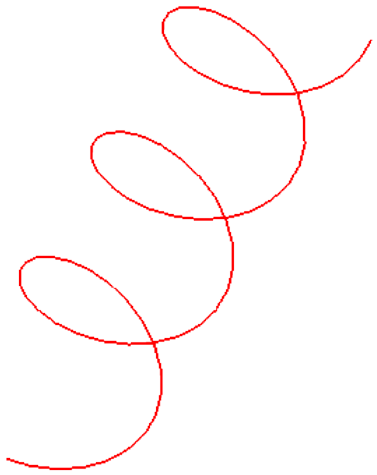
HepRep: a Generic Interface Definition for HEP Event Display Representables



# HEP3D



- Solides 3D optimisés (tous G4 + d'autres).
- 3D Outlining (wireframe).
- Opérations booléennes (intersection, différence, union).
- D'autres objets 3D HEP (Helix,...).
- G4-like API.



# JAS

## Java Analysis Studio



- Les histogrammes 1d, 2d et 3d, les plots XY, les scatterplots etc..
- Impression de haute qualité.
- Fitting (binned ou unbinned) en utilisant un ensemble extensible d'optimiseurs comprenant Minuit.
- Exportation des plots dans une variété de formats comprenant PS, EPS, PDF, SVG, GIF, PNG.
- GUI pour exécuter l'analyse commune (plotting, ntuple-analyse, fitting, etc..) facile à apprendre .
- Une analyse plus complexe peut être définie en utilisant une variété des langues du scripting (pnuts, jython, etc.), ou en écrivant des modules d'analyse de Java.
- Capable à lire des données dans une variété de formats comprenant : Bases de données de SQL, texte, Root,...
- Système d'analyse conforme à AIDA.
- Éditeur et compilateur intégrés.
- Possibilités simples de spreadsheet.
- Le carnet pour enregistrement d'analyse, et en option de montrer des résultats au WWW.
- La structure fortement modulaire permet à l'addition, au remplacement ou au déplacement des modules et adaptation aux besoins du client de domaine particulier de problème.

GUI

Tree of Objects

Integrated Help with Executable Examples

Java Class

Python/PNuts Script

Python/PNuts Command Line

Graphical/Textual Object Representation

The screenshot displays the Athena JAS3 Plugin interface. At the top, the menu bar includes File, Edit, View, Tuple, Loop, Window, Athenaem, and Help. The main window is titled "Athenaem - Athena JAS3 Plugin" and features a logo of Athena.

**Tree of Objects:** A hierarchical tree on the left shows the current state of the application, including folders like "tree-0" and "tuple (tupleLabel)", and objects like "IFlat", "fGauss", and "fFlat".

**Integrated Help with Executable Examples:** A central panel titled "Athenaem - Athena JAS3 Plugin" contains a list of help topics such as "How to Use IAS", "How To Start Athena Python Scripts", and "How to Connect to Athena Python".

**Java Class:** A window titled "Fit.java" shows the source code for a Java class that generates a histogram and a fit. The code includes imports for "hep.aida" and "java.util.Random", and defines a "Fit" class with a "main" method.

**Python/PNuts Script:** A window titled "Tuple.py" shows a Python script that uses the Athena API to create a tuple, fill it with random data, and generate histograms. The script includes comments and code for creating factories and histograms.

**Python/PNuts Command Line:** A terminal window at the bottom left shows a simple Python command: `>>> print "Hello"` followed by the output "Hello".

**Graphical/Textual Object Representation:** The right side of the interface displays several visualizations: a table of data for the "tuple" object, a histogram labeled "h2d" with a Gaussian fit, and a scatter plot. The table shows columns for "IFlat", "fGauss", and "fFlat" with numerical values. The histogram shows a distribution of data points with a fitted curve. The scatter plot shows a distribution of data points in a 2D space.

# MultiLanguage

**FREE** HEAD application

The screenshot displays the JAS3 application window. On the left is a file explorer showing a project structure with folders like 'tree-0', 'tree-1', and 'tree-2', and files like 'Fit', 'Histogram 1D', and 'tuple'. The main area is divided into several panes:

- Styles.pnut plot:** A histogram plot titled 'h1 - h2 - h3'. The y-axis is labeled '# Evt' and ranges from 0 to 450. The plot shows a distribution of data points (blue diamonds) with a fitted curve (green line). A tooltip for 'h1' shows: 'Entries : 1000', 'OutOfRange :', 'Mean : 9.0788E-', 'Rms : 1.005'.
- Tuple.py:** A Python code editor containing the following code:

```
1 from hep.aida import *
2 from java.util import Random
3 from java.lang import *
4
5 af = IAnalysisFactory.create();
6 tree = af.createTreeFactory().create();
7 hf = af.createHistogramFactory(tree);
8 tf = af.createTupleFactory(tree);
9
10 r = Random()
11
12 columnNames = [ "fFlat = 0", "fGauss = 3.", "fFlat = -2." ]
13 columnClasses = [ Integer.TYPE, Float.TYPE, Float.TYPE ]
14
15 tuple = tf.create( "tuple", "tupleLabel", columnNames, columnClasses, "" );
```
- Fit.java:** A Java code editor containing the following code:

```
1 import hep.aida.*;
2 import java.util.Random;
3
4 public class Fit
5 {
6     public static void main(String[] args)
7     {
8         // Create factories
9         IAnalysisFactory analysisFactory = IAnalysisFactory.create();
10        ITreeFactory treeFactory = analysisFactory.createTreeFactory();
11        ITree tree = treeFactory.create();
12        IPlotter plotter = analysisFactory.createPlotterFactory().create("Fit.java Plot");
13        IHistogramFactory histogramFactory = analysisFactory.createHistogramFactory(tree);
14        IFunctionFactory functionFactory = analysisFactory.createFunctionFactory(tree);
15        IFitFactory fitFactory = analysisFactory.createFitFactory();
16
17        IHistogram1D h1 = histogramFactory.createHistogram1D("Histogram 1D",50,-3);
18
19        Random r = new Random();
20
21        for (int i=0; i<100000; i++) {
22            h1.fill(r.nextGaussian());
23            h1.fill(r.nextDouble()*10-5);
24        }
25
26        IFunction gauss = functionFactory.createFunctionFromScript("gauss",1,"background");
```
- Styles.pnut:** A Pnuts code editor containing the following code:

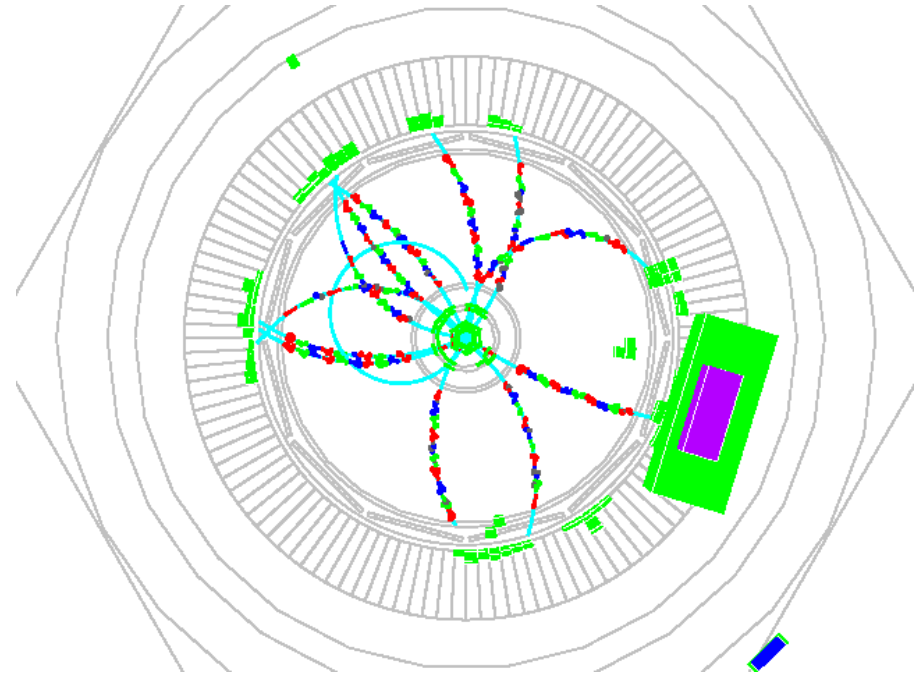
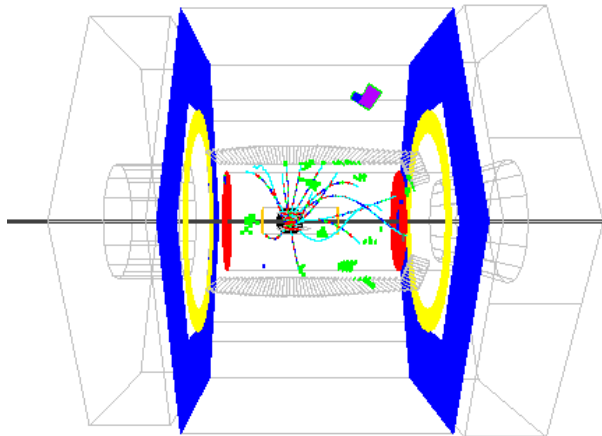
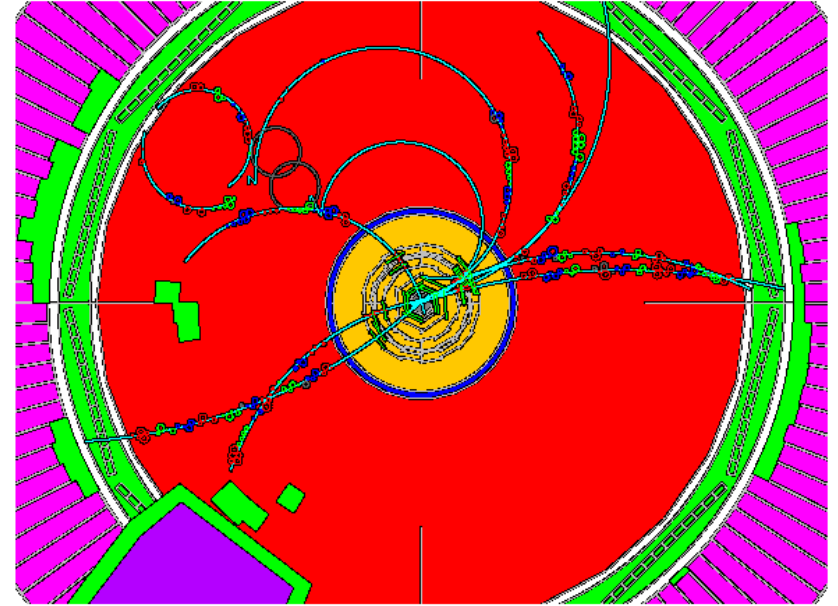
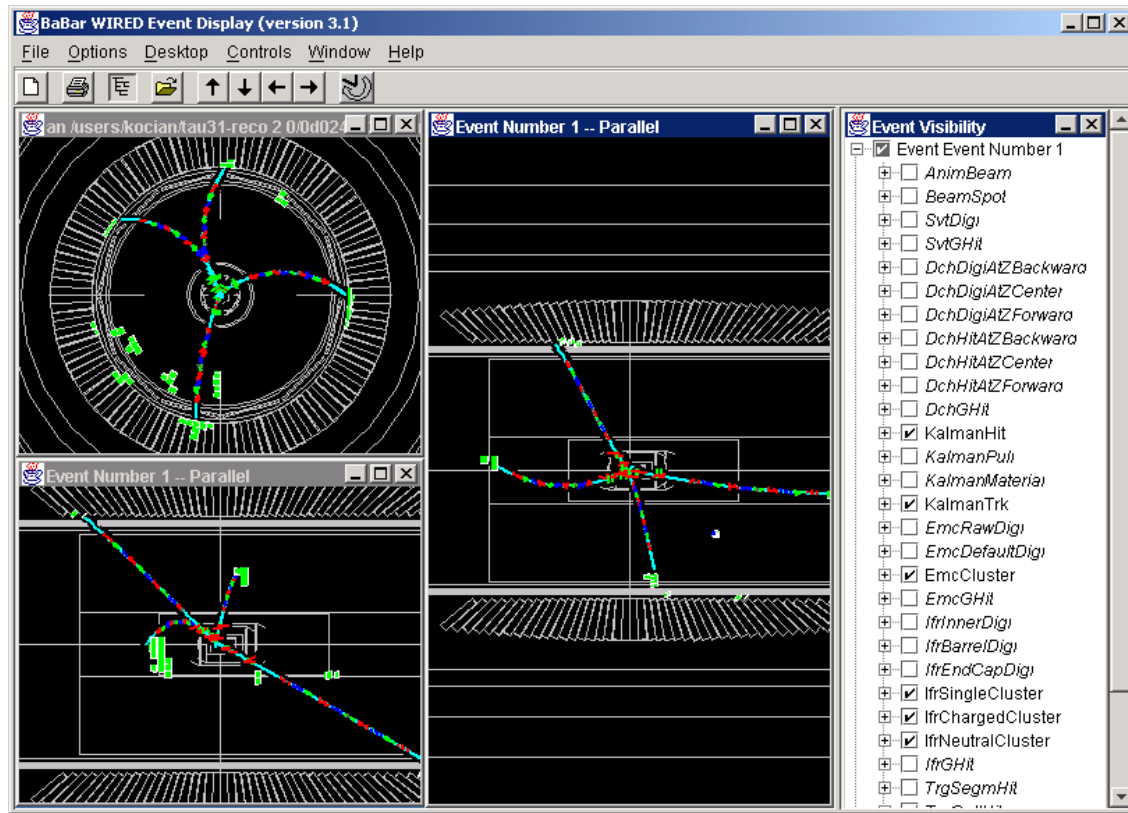
```
1 IAnalysisFactory = class hep.aida.IAnalysisFactory
2 af = IAnalysisFactory.create()
3 tree = af.createTreeFactory().create()
4 hf = af.createHistogramFactory(tree)
5
6 h1 = hf.createHistogram1D("h1",100,-5,5)
7 h2 = hf.createHistogram1D("h2",100,-5,5)
8 h3 = hf.createHistogram1D("h3",100,-5,5)
9
10 Random = class java.util.Random
11 r = Random()
12
13 for (i = 0; i<10000; i++) {
14     h1.fill(r.nextGaussian())
15     h2.fill(r.nextGaussian())
16     h3.fill(r.nextGaussian())
17 }
18
19 plotterFactory = af.createPlotterFactory()
20 plotter = plotterFactory.create("Styles.pnut plot")
21
22 //This are the default styles for the region
23 regionStyle = plotter.region(0).style()
24 regionStyle.dataStyle().setParameter("fillHistogramBars","false")
```

Red handwritten text 'Python' is overlaid on the Tuple.py code, and 'PNuts' is overlaid on the Styles.pnut code. Red handwritten text 'Java' is overlaid on the Fit.java code. The bottom status bar shows 'Pnuts version 1.0rc2 (20030326134609)' and a taskbar with 'Compiler', 'Fit', 'Jython', and 'Pnuts' tabs. The system tray shows '6.25 / 9.14 MB'.



# WIRED

## WWW Interactive Remote Event Display



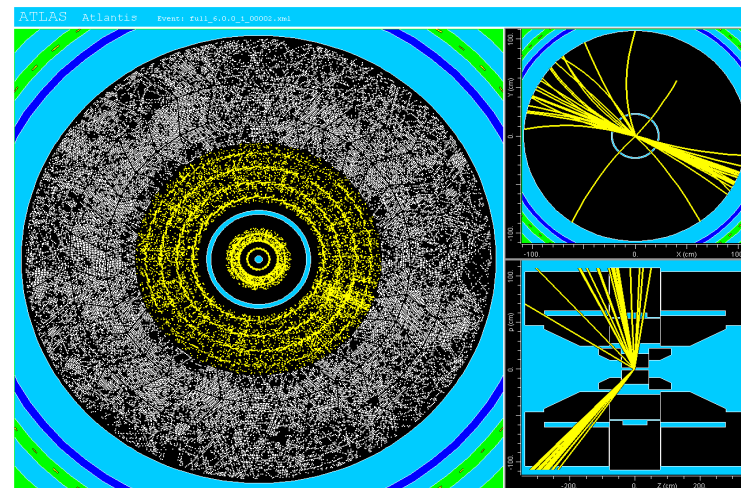
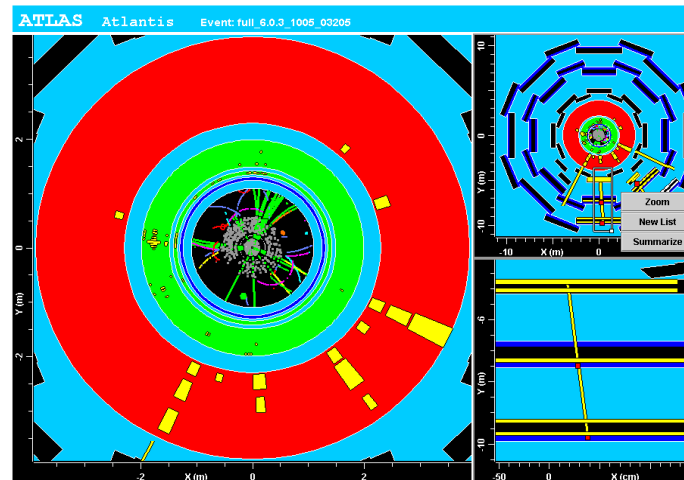
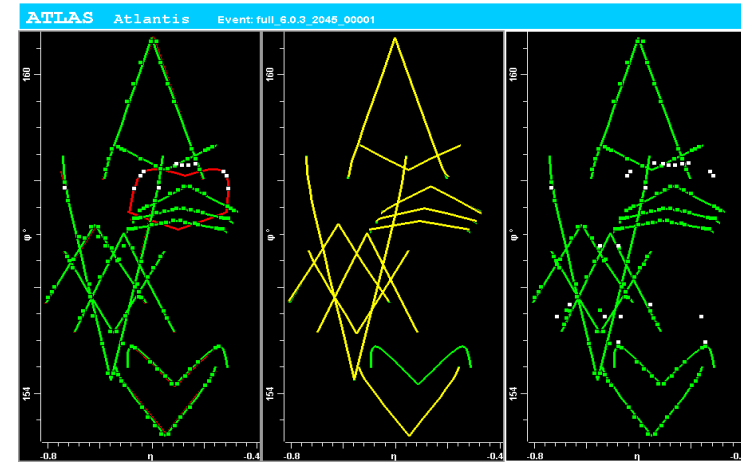
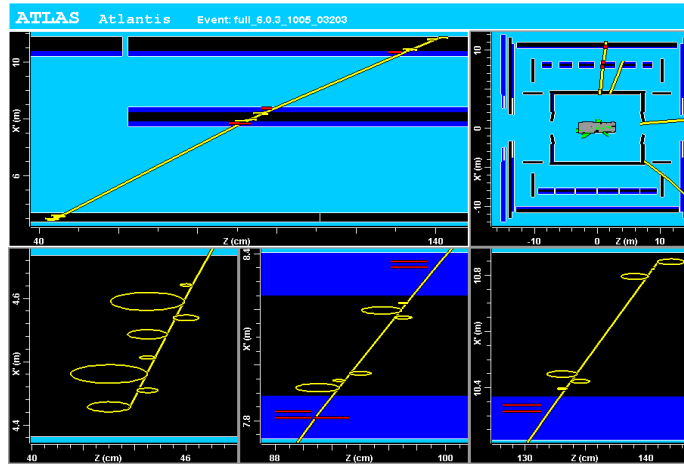


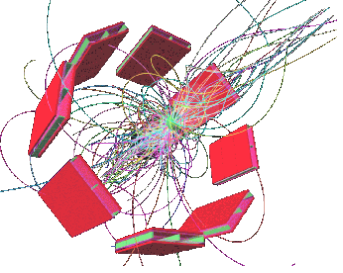
# Atlantis

## Atlas Event Display



- Affichage d'événement d'Atlas basé sur DALI d'Aleph.
- Recherche visuelle et arrangement physique de tous les types d'événements.
- Aide à développer des algorithmes de reconstruction et d'analyse.
- Facilite la correction pendant commisioning.
- Création des images et des animations pour des présentations, des publications et des expositions.



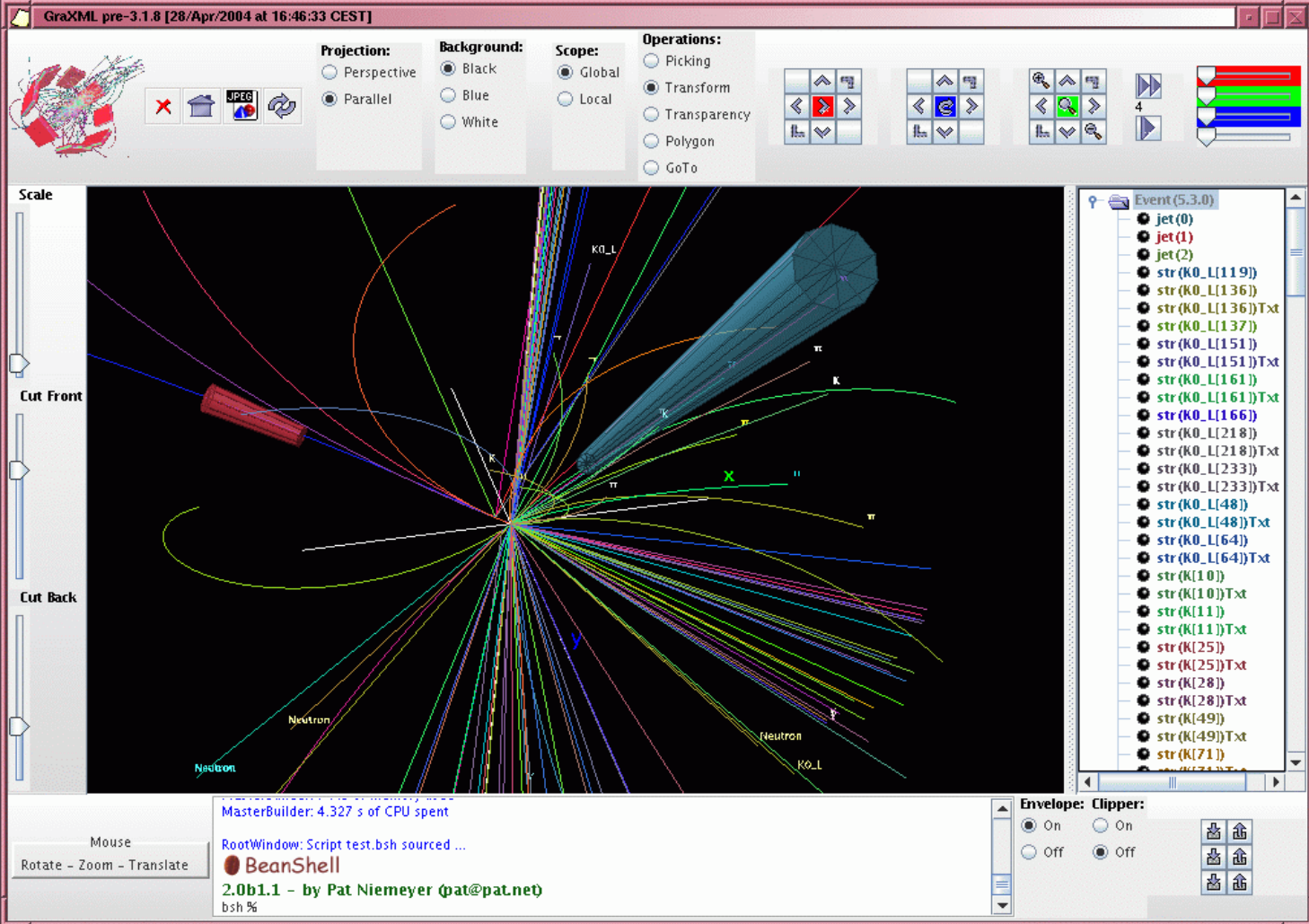


# GraXML

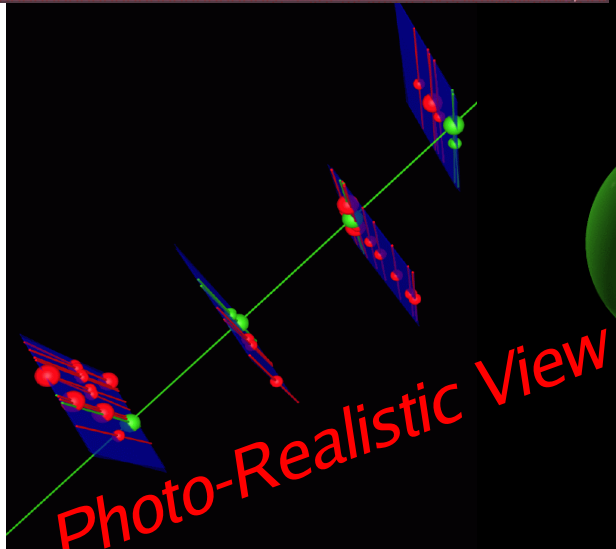
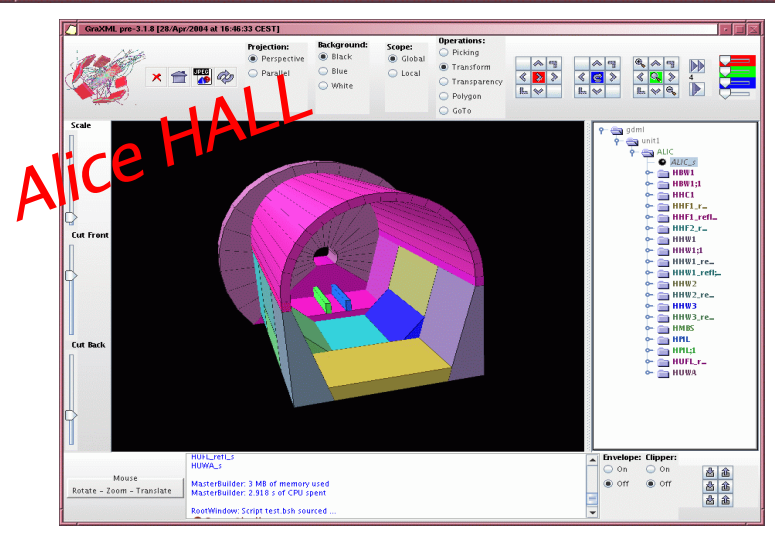
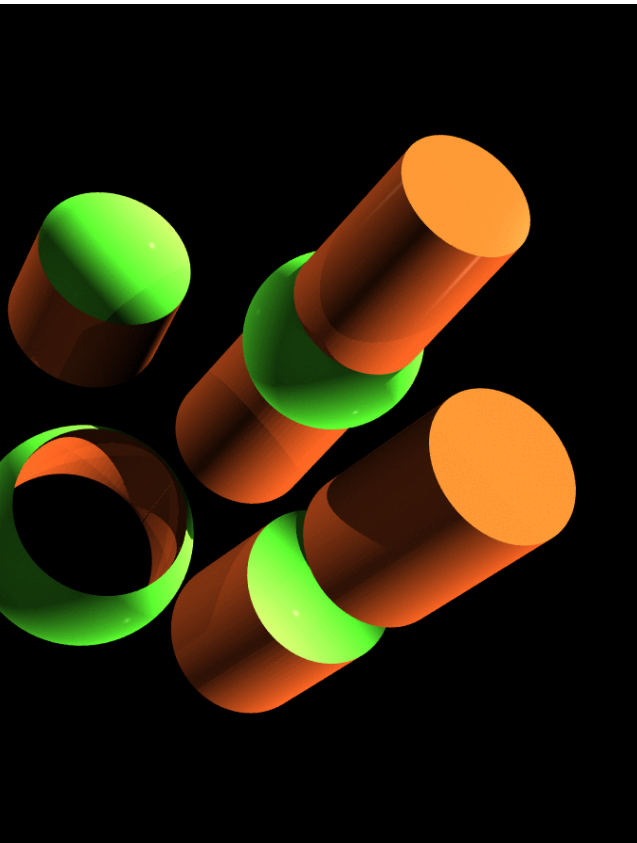
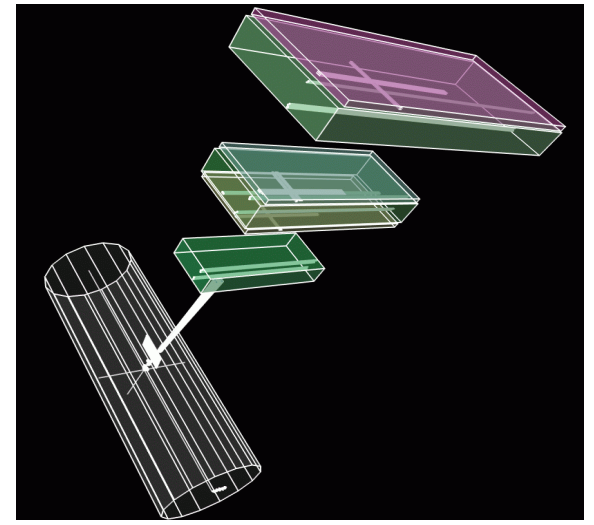
Framework for manipulation and visualisation of  
geometrical objects in space



- La Toolkit GraXML fournit une base pour des opérations sur des objets 3D (éléments, événements de détecteur...).
- Chaque source extérieure de données 3D (**GDML**, **AGDD**, ...) est traduite automatiquement en modèle générique qui est alors analysé et traduit en modèle géométrique avec des modules de GraXML.
- La construction de ce mode géométrique est paramétrée par plusieurs paramètres (niveau d'optimisation, niveau de qualité...) de façon qu'elle puisse être employée dans les applications avec des conditions différentes.
- Le modèle géométrique peut alors être directement employé par des applications.
- Deux applications de visualisation sont fournies par GraXML: L'affichage interactif de GraXML et le convertisseur dans des formats divers de géométrie (**VRML**, **X3D**, **Root TGeo**,...). D'autres applications peuvent être facilement développées.



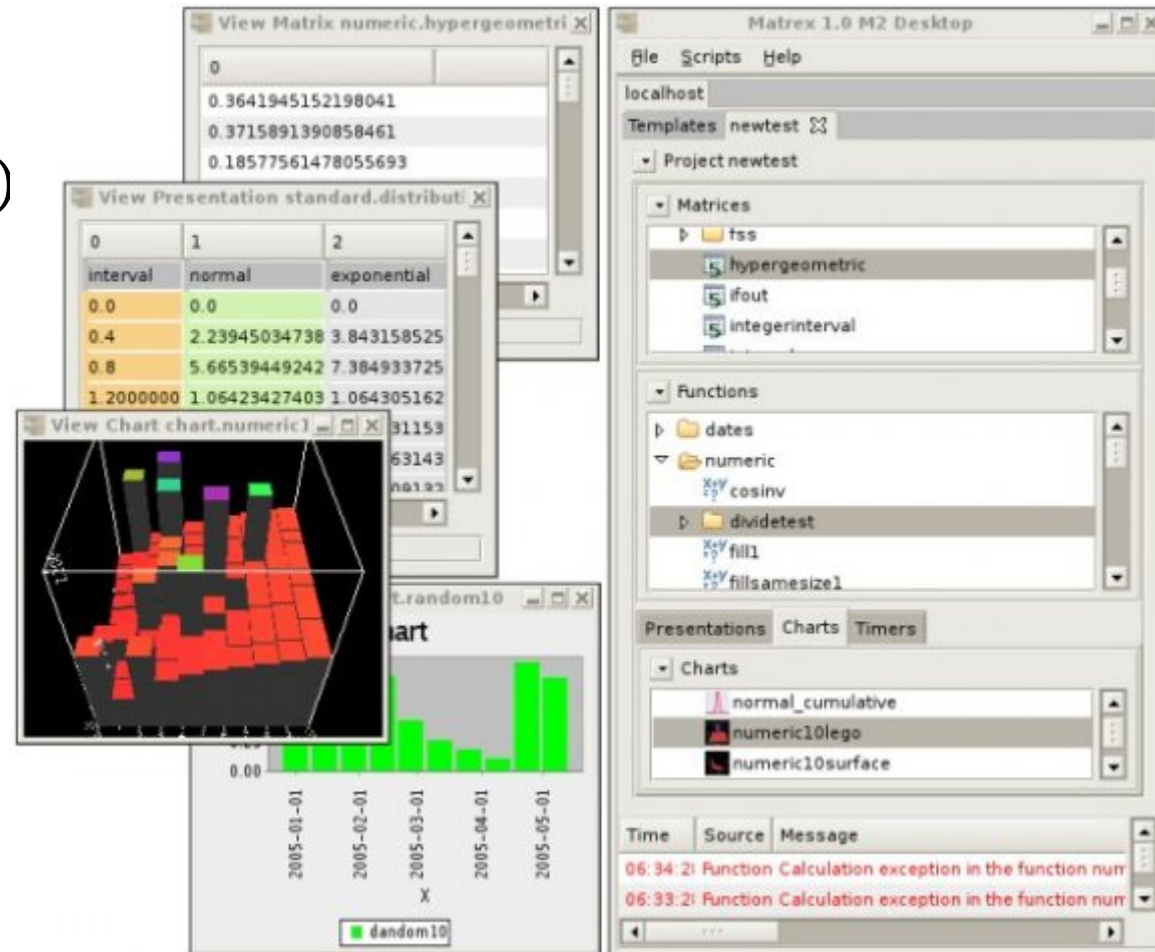
**FREE** **HEP**  
application



# Matrex

Graphical Tool to Show, Organize and Calculate Big Amounts of Data

- Les fonctions (formules) sont appliquées aux blocs entiers de données (appelées les matrices) pas aux cellules.
- Les présentations et les diagrammes sont visibles seulement quand vous voulez, non tout le temps.
- Les calculs ne bloquent pas l'application et peuvent être exécutés en parallèle (multithread).



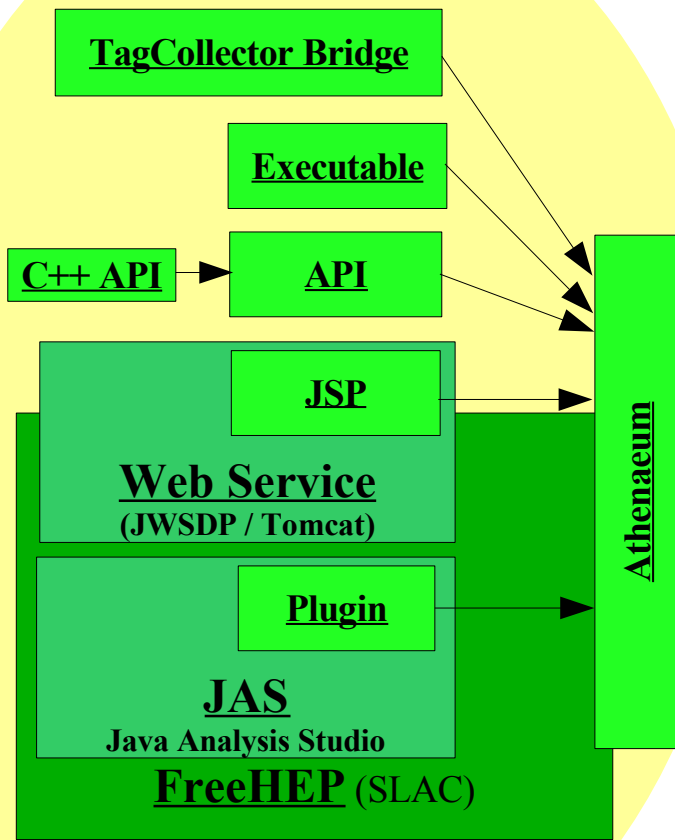


# Athenaeum

Remote Client to  
Atlas Offline Framework

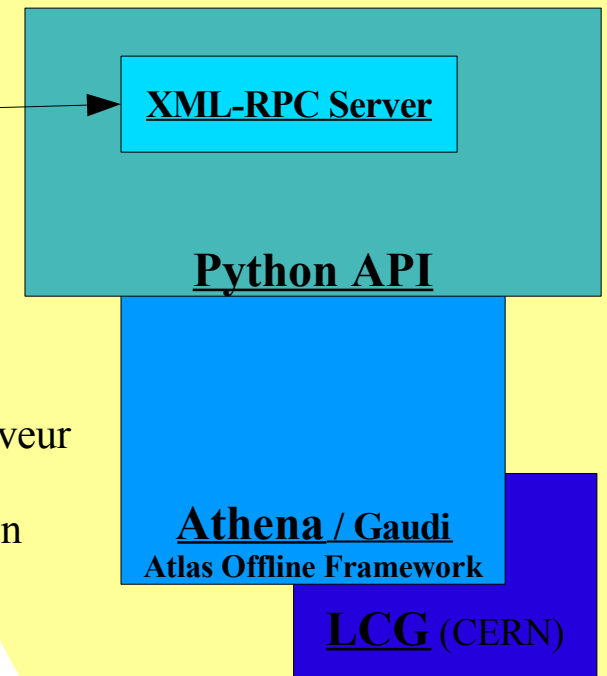


## Local Client

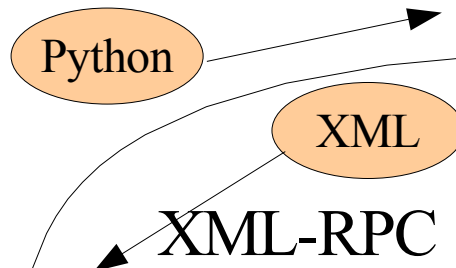


Java / xMB  
(partout)

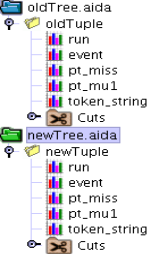
## (Remote) Server



C++ / xGB  
(seulement sur le lxplus)



- Athenaeum se connecte au serveur d'Athéna (extérieur).
- N'importe quel script de python (d'Athéna) peut être envoyé directement à partir du client à Athéna.
- Les résultats (habituellement dans XML) sont renvoyés.
- Des scripts spéciaux de python sont fournis pour présenter quelques données d'Athéna.
- Plusieurs clients existent.

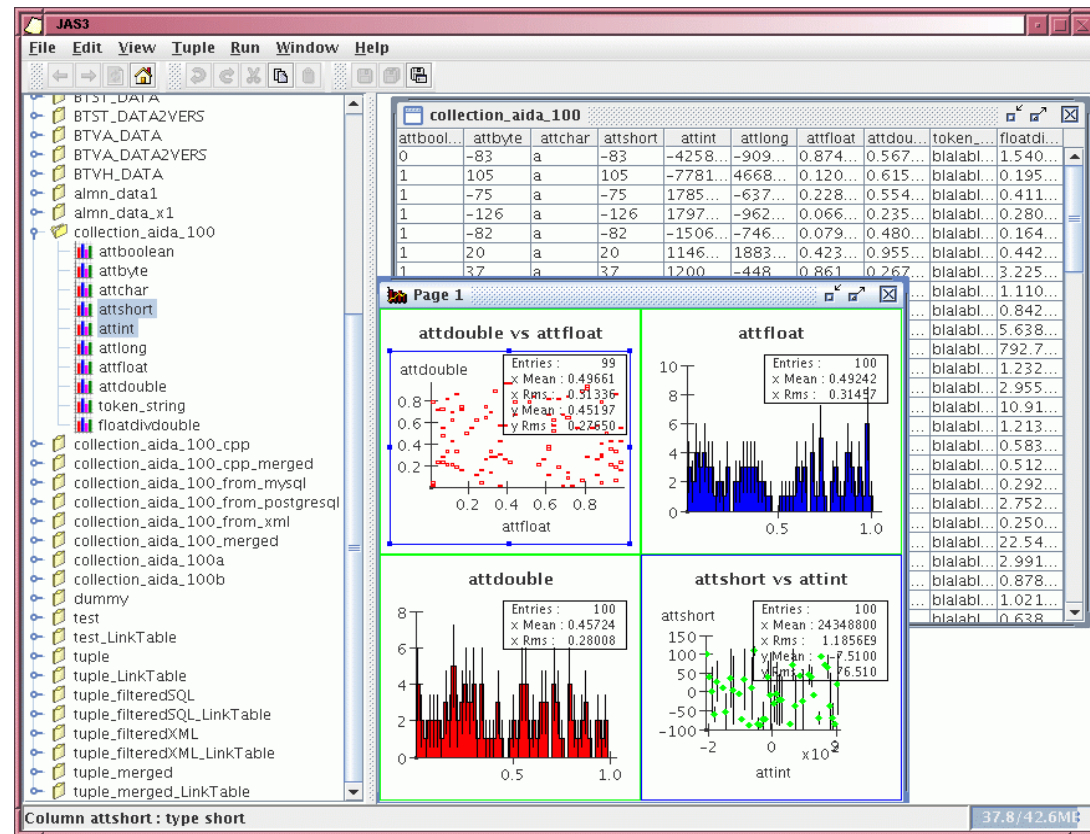
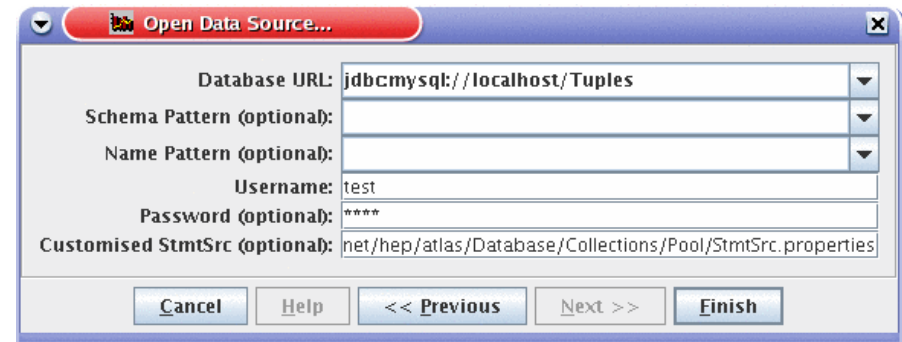


# SQLTuple et ColMan

## AIDA SQL binding



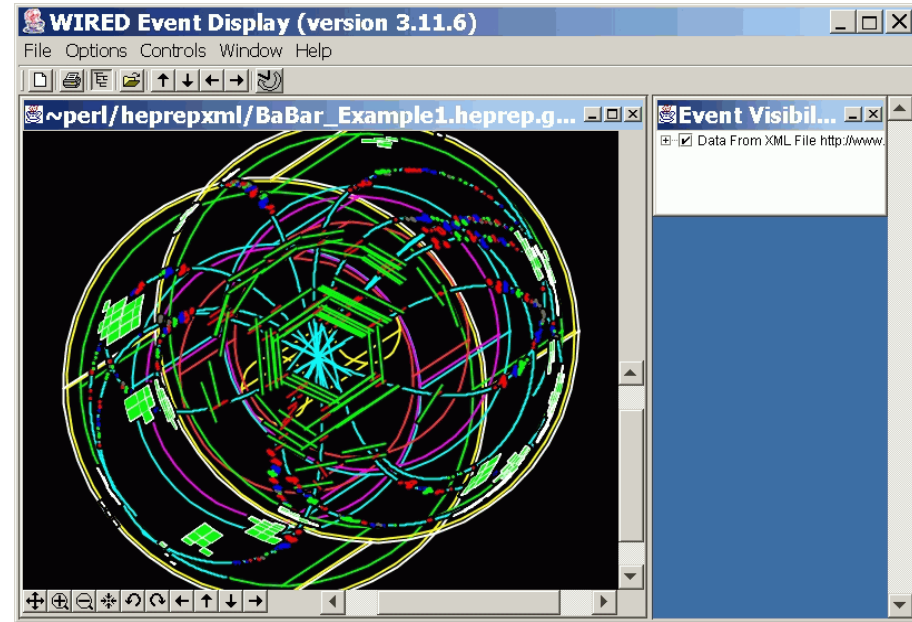
- SQLTuple fournit AIDA ITuple avec stockage de SQL.
- ColMan fournit des utilités de gestion de collection pour AIDA ITuples (dans tout format soutenu, incl. SQL, XML, Root...) :: merging, filtering, analysing.
- Utilisable dans JAS.
- Fournit des couches de C++ et de WebService.
- Supporte tout RDBMS, la distribution est configurée pour MySQL, Postgres et McKoi.



# Geant4

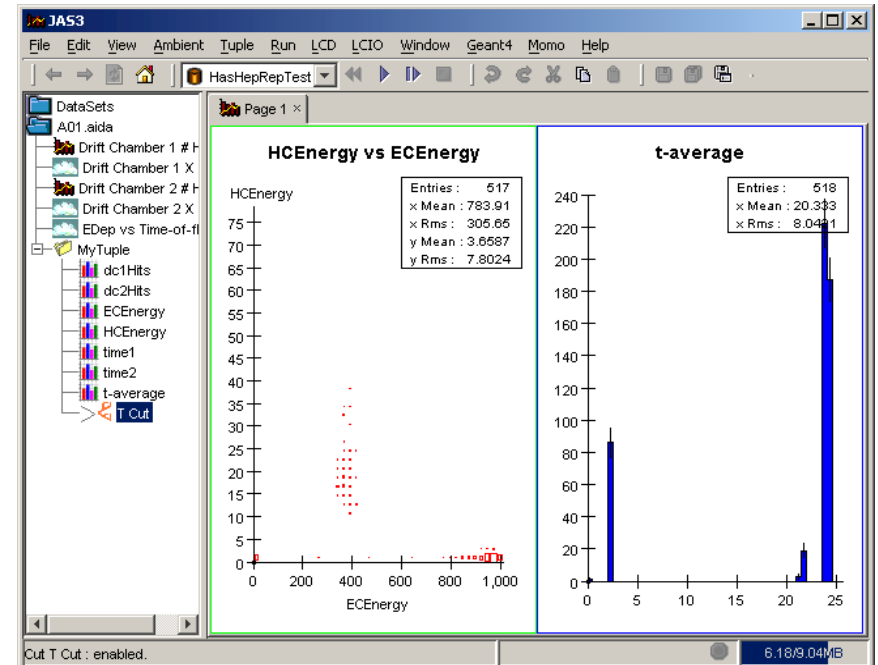


## Emploie AIDA et HEPRep.



The screenshot shows the JAS3 software interface with a data table titled "MyTuple". The table has 7 columns: dc1Hits, dc2Hits, ECenergy, HCenergy, time1, and time2. The data consists of 1100 rows of numerical values. The interface includes a menu bar (File, Edit, View, Ambient, Tuple, Run, LCD, LCIO, Window, Geant4, Momo, Help) and a toolbar with navigation icons. A sidebar on the left shows a tree view of data sets and tuples.

	dc1Hits	dc2Hits	ECenergy	HCenergy	time1	time2
5	5	5	934.6909244396825	0.0	4.987278665919724	43.39693418429103
5	5	5	891.9379165006716	0.0	4.986818705037497	43.232656432740285
7	5	5	351.0629634377849	21.80128748945547	5.014467163590249	43.212292359567904
8	5	5	357.3895289224486	22.26857568972258	5.015349752020167	43.46988179788557
5	5	5	981.5995190718681	0.0	4.9870220186780525	43.06011516361901
9	5	9	397.51028712893105	17.018390800317	5.01567551116749	43.622354806275354
5	5	7	381.3048518661324	15.884693991171186	5.013820908635788	43.35654388913619
5	5	5	964.3255883733482	0.0	4.986793214308921	43.15679674219883
5	5	5	968.1717945208153	0.0	4.987234881027987	43.02189859463945
5	5	5	383.63013626685233	18.746095950351375	5.014415449055382	43.23269110084189
5	5	5	345.8298611919274	16.762355865002128	5.015409454919504	43.2646986225857
6	5	5	367.2333715722816	21.203486000528123	5.0141192104373955	43.38606972316038
5	5	5	952.6560980058448	0.32394775224066397	0.0	43.027276453717754
5	5	5	942.2616446076202	0.0	4.9872837678871855	43.40951686157182
5	5	5	958.679824957826	0.0	4.986881864680074	43.09170120216978
6	5	5	986.8788228306686	0.0	4.9867897615832995	43.21384273984369
7	5	7	956.9453703255346	0.0	4.986795837865437	43.23816598616564
5	7	7	371.6142114922393	40.346880048161694	5.013502826674904	43.543072666258254
6	5	5	945.8280973565053	2.186296860728863	4.9871812573220655	0.0
5	5	5	954.2950526538824	0.0	4.9871826459884865	43.03133057042492
9	5	5	393.77146505285964	22.498974534172415	5.015255116056872	43.244234615214786
7	5	5	956.3925645324418	0.0	4.9872768360152255	42.98306995852938
5	8	8	957.6303429544486	0.0	4.9868526914287035	43.1192277099023
6	5	5	975.5067882160424	0.0	43.33728173796454	0.0
5	7	7	412.1093182844084	13.810608984485208	5.014825621180121	43.59279707744945





# Development Process



- Identifier les composants réutilisables dans un logiciel HEP en Java et entrer les dans la bibliothèque commune:
  - Hiérarchie hep.\* pour le code HEP-spécifique
  - Hiérarchie org.freehep.\* pour le code général
- Distribué sous le **LGPL**.
- Sources stockés dans **SVN** (<svn://svn.freehep.org/svn/freehep/trunk> freehep).
- Construction en utilisant **Ant** et **Maven** d'Apache.
- Les « releases » réguliers, l'intégration continue (employant le **Continuum** d'Apache).
- Documentation complète.
- **JIRA** Bug-tracking, Forum et Mailing list.
- Basé au SLAC, contributions de CERN, LAL,...
- Participation aux projets HEP: AIDA, Grid, Geant4,...

Forum

Web

Bug Reporting



FreeHEP Forum: Welcome to the forum - Mozilla (Build ID: 2004042110)

http://forum.freehep.org/index.php?t=index&c=2:0\_1:1\_3:1\_4:1&rid=(

Members Search F.A.Q. Register Login Home

Show: Today's Messages :: Unanswered Messages :: Show Polls :: Message Navigator

Forum	Messages	Topics	Last message
<b>FreeHEP Java Library</b> - Discussion, comments and questions about the FreeHEP Java Library			
Announcements Announcements of new releases etc. Moderator(s): duns	1	1	Mon, 08 March 2004 By: turri
Vector Graphics Discussion of the vector graphics package.	8	1	Tue, 16 March 2004 By: duns
AIDA Discussion of the FreeHEP implementation of AIDA (JAIDA + AIDAJNI + CERNLIBAdapter)	26	7	Wed, 05 May 2004 By: zapzarap
General General discussion on topics for which no specific forum exists.	12	5	Thu, 01 April 2004 By: tonyj
Developer Developer's forum	3	1	Wed, 17 March 2004 By: jacek

JAS - Discussion of JAS Analysis Studio

WIRED - Discussion of World-Wide Web Interactive Remote Event Display

Test Category - Just a test category

Login Password  Use Cookies?

Logged in users list [show what people are doing] [Today's Visitors]

There are 0 members(s), 0 invisible members and 1 guest(s) visiting this board. [Administrator] [Moderator]

Our users have posted a total of 50 messages inside 15 topics.  
We have 25 registered user(s).  
The newest registered user is ashiganjum  
Last message on the forum: Re: IHistogram1D with no error bars

**Legend**

New messages since last read No new messages since last read

**Current Time:** Thu May 07:51:31 Pacific Daylight Time 2004

Total time taken to generate the page: 0.02468 seconds

... Contact :: Home ...

Powered by: FUDforum 2.6.2  
Copyright ©2001-2004 Advanced Internet Designs Inc.

Browse Project - FreeHEP - Mozilla (Build ID: 2004042110)

http://bugs.freehep.org/secure/BrowseProject.jspp?id=10012

HOME BROWSE PROJECT FIND ISSUES CREATE NEW ISSUE QUICK SEARCH:

**Projects : FreeHEP Java Library & Tools** (Key: FREEHEP)

**Project Lead:** Mark Donszelmann  
**URL:** http://java.freehep.org/  
**Description:** The FreeHEP Java Library and its Tools. This library contains many components on which bugs can be reported independently.

Create a new issue in project FreeHEP Java Library & Tools

**Reports:**  
Open Issues | Road Map | Change Log | Release Notes | Popular Issues

Component/s	Versions	
(with open issues in each component)	(with open issues due to be fixed per version)	
AID Compiler	16	10
FreeHEP IO	1	38
HEP IO	6	139
Infrastructure	5	
JHepRep	13	
Others	2	
PS Viewer	40	
Tools	2	
Vector Graphics	86	
YaPPI	7	
No Component	9	

**Preset Filters**

- ALL
- Outstanding
- Resolved recently
- Added recently
- Updated recently
- Most important

**Project Summary**

- Open: 172
- In Progress: 15
- Reopened: 0
- Resolved: 75
- Closed: 182

**Open Issues**

**By Priority**

- Blocker: 2
- Critical: 4
- Major: 81
- Minor: 96
- Trivial: 4

**By Assignee**

- Mark Donszelmann: 82
- Tony Johnson: 10
- Unassigned: 95

Powered by Atlassian JIRA™ (Professional Edition, Version: 2.6.1-#65) - Bug/feature request  
Contact Administrators

# Taille, Vitesse



- Cca 2000 classes Java (300 kloc), aussi C++, Python, Aid, Ant, Xml,...
- Temps de construction : moins 2 minutes sur le PC commun.
- Taille des bibliothèques compilées : moins 7 MB (dans 40 bibliothèques).
- Utilisable sur toutes les plateformes (sans recompilation).
- Aussi rapide (ou plus rapide) que bibliothèques semblables C++.

Members: Mark Donszelmann, Tony Johnson, Max Turi, Victor Serbo

Contributors: Gary Bower, Julius Hrivnac, Cal Loomis, Joseph Perl, Peter Armstrong, Simon Fischer, Andre Bach, Patric Hellwich, Sarni Kama, Paul, Spence

<http://java.freehep.org>

**FREE**

**HEP**

*FreeHEP a déjà aujourd'hui la plupart  
des fonctionnalités qui sont  
promises pour quelques futurs  
projets dans C++.*

