

Summary of the AIDA workshop 2003

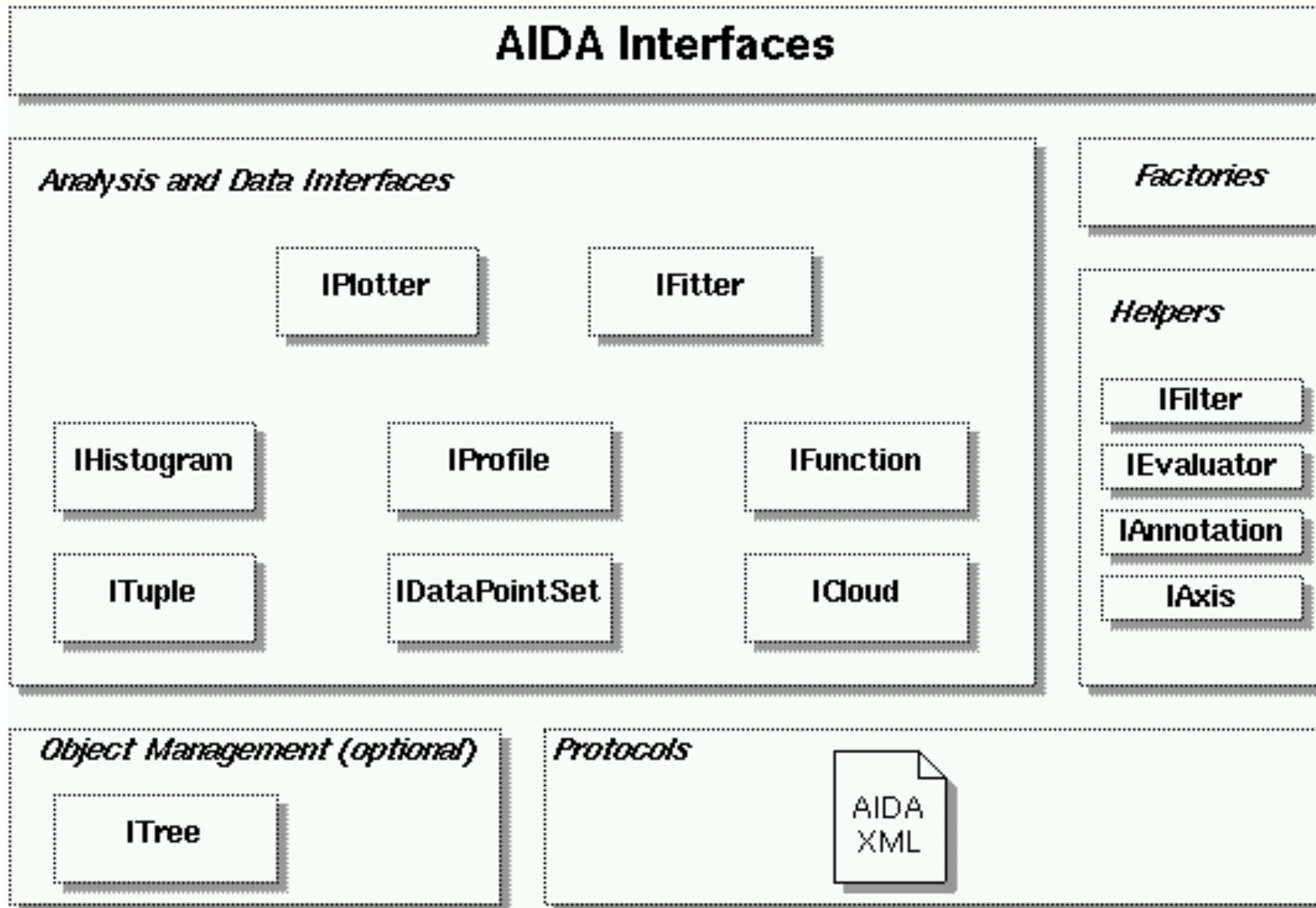
Presentation to the LCG Application Area
July 9th, 2003

Andreas Pfeiffer, CERN, EP/SFT

What is AIDA

- ❖ **AIDA defines today *user level* interfaces for some common analysis data objects**
 - ❑ **IHistogram, ICloud, IDataPointSet, ITuple**
- ❖ **Management of these objects**
 - ❑ **IAnalysisFactory, IFactories, ITree (IManagedObject)**
- ❖ **Some common facilities**
 - ❑ **IFitter, IPlotter**
- ❖ **ITree is user interface to *management* and the *storage* of objects**
 - ❑ Hiding complexity
 - ❑ Supporting multiple different storage formats (in various implementations)

AIDA - Abstract Interfaces for Data Analysis (user level)



Items presented/discussed

- ❖ **Status and demos from the three existing implementations**
- ❖ **Presentations from users**
 - ❑ Geant-4 advanced examples
 - ❑ PI Proxy layer
 - ❑ Statistical testing project
 - ❑ Demo of Panoramix/DaVinci (LHCb)
- ❖ **Discussions between developers**
 - ❑ Prototypes on stores, interchange formats
 - ❑ Concentrating on feature requests on AIDA version 3 and component level interoperability
 - ❑ Very intense, very fruitful
 - ❑ Resulted in concrete workplan

AIDA development process

- ❖ **Starting by defining Abstract Interfaces to allow interoperability on the user-level**
 - ❑ Users can interchange implementations without recompiling their code
 - ❑ Ability to read/write various different storage formats
 - AIDA-XML, HBOOK, ROOT, SQL, ...
- ❖ **Defining a common set of functionality as an agreed superset of the existing implementations**
 - ❑ Not just the “least common denominator”
- ❖ **Discuss with others the Abstract Interfaces and agree on a common set**
 - ❑ Starting from existing implementations, possibly with abstract interfaces

AIDA development process (II)

❖ Next step: improve interoperability at *component level*

- ❑ Mix components from different implementations
 - mix histogram and plotters based on different technologies
 - mix histogram and fitters from different implementations

❖ Will give users even more freedom

- ❑ Can choose “bits and pieces” as required
- ❑ Choice by “strings” can allow run-time selection without need for recompilation

AIDA Implementations

❖ Three groups, two languages

- ❑ Anaphe – C++
 - CERN
 - No longer in active development
- ❑ JAS / JAIDA – Java
 - SLAC
 - AIDA_JNI to use from C++
- ❑ OpenScientist – C++
 - LAL

❖ Full implementations of AIDA 3 interfaces

- ❑ All classes are there
- ❑ some methods may still be dummy

❖ Three similar bindings to Python exist

- ❑ Not yet standardized

CERN reorganisation for LHC and relationship to AIDA

- ❖ **Anaphe team dissolved**
 - Only bugfixes/maintenance
- ❖ **LHC Computing Grid (LCG)**
 - Experiments *and* CERN management together
- ❖ **New LCG project: “Physicist Interfaces” (PI)**
 - Includes some people of old Anaphe team

- ❖ **==> New Interlocutor from CERN**

PI group proposals

- ❖ **Present to users proxy classes for value semantics and to hide the management**
 - ❑ Unmanaged objects may limit user functionality in tools with GUIs
 - ❑ Creation of unmanaged objects is needed in other frameworks
- ❖ **Implementation of the proxy classes over the SEAL software**
 - ❑ Using the SEAL plugin-manager
 - ❑ Better to base this on developer level (Abstract) Interfaces for decoupling
- ❖ **Begin of implementation of some AIDA data interfaces using ROOT classes**
 - ❑ IHistogramxD using ROOT::THx
 - ❑ IProfile1D using TProfile1D

AIDA Workplan

- ❖ **Concentrate on fine grain Interoperability issues**
- ❖ **Be able to use “subpackages” from different impl.**
 - Needs DevIFs
 - e.g., to restore an OpenScientist histo from a Anaphe-HBOOK store
 - Needs agreed “loading mechanism”
 - No longer AIDA_createAnalysisFactory()
 - Common naming schedule for creating the (dev-) factories (FactoryProvider)
 - AIDA_<subpkg>_<impl> (default)
 - AIDA_<subpkg>_<impl>_<subType>
 - Using XML file for discovery

Priorities for interoperability

- ❖ Histograms
 - ❖ Plotting
 - ❖ Fitting (with Function)
 - ❖ DataPointSet
-
- ❖ ==> **By end October have a concrete demonstration of component level interoperability**

Next Workshop end October

- **Finalizing demonstration of component level interoperability**
 - Mix Histo/Plot/Fit from different implementations
 - Document on web in addition to examples
- **Discuss ITuple, IStore, ITree with POOL and SEAL developers**
 - Get feedback on requirements and improve IFs
- **Discuss observer/notification system**
 - On developer level

Users

❖ **AIDA users**

- ❑ BaBar online
- ❑ Gaudi/Athena users
- ❑ Geant-4 advanced examples users
- ❑ Linear collider users
- ❑ PI project of LCG
- ❑ Various individuals

❖ **User attendance at workshop was small**

- ❑ Late announcement, user part of workshop was “add-on” on developer workshop (no free coffee ?)

❖ **Need more advertisement, dedicated user-workshop**

- ❑ Since Anaphe at CERN has been stopped, hard to attract people here
- ❑ PI should now lead the promotion of AIDA

Summary

- ❖ **Very fruitful discussions between AIDA developers**
 - ❑ Small changes only to existing 3.0 IFs
 - ❑ Started work on developer level interfaces for component level interoperability
- ❖ **Quite some discussions with users**
 - ❑ Mostly “developer-users” with concrete feedback
- ❖ **Concrete workplan to achieve component level interoperability**
 - ❑ Through new developer level interfaces
- ❖ **Role of PI project in AIDA**
 - ❑ Represent the interests of the LHC experiments within the AIDA project, and will focus its work on those LHC interests
 - ❑ Encourage and spread the use of AIDA in the LHC community along the line of the blueprint architecture