



汎用データ解析ソフト生成ツール ~~ROME & ARGUS~~

Data analysis software generator, ~~ROME & ARGUS~~



東大ICEPP, PSI,ETH

R. Sawada, S. Ritt, M. Shneebeli

Mar, 30, 2006

The Physical Society of Japan

61st Annual Meeting

Ehime University



Introduction



What is ROME ?



- ROME was developed in **MEG** collaboration in 2004.
- ROME is independent of experiments. It is a **general purpose tool**.
- Several groups are using it in **U.S.**, **Canada** and **Europe**.
- ROME works under **Linux**, **Windows** and **Mac**. (Maybe under other UNIX like system with minor modification.)
- ROME is neither an analysis software nor a library. ROME **generates analysis software** from an experiment definition XML file.
- Generated software is written in **C++**.
- Generated software works on the basis of **ROOT** library.



Concept



- **Automatic** code generation
 - **Less work** of physicists. (Physicists must spend time for physics, instead of programming !)
 - **Less bugs** than hand written code
 - Better **readability** of code.
 - Always **compatible document** with code.
- Separation of calculation and data container classes (**tasks** and **folders**)
- **Flexible** control of calculation. For instance, you can try different re-construction algorithms without re-compiling.
- Sharing experiment independent part with other groups
 - **Improvements** from other experiments reflect on **your analysis**. (vice versa)



How your analyzer looks like



```
dhcp157 % ./xyzanalyzer.exe -i romeConfig.xml
reading configuration from romeConfig.xml
*****
*
*           XYZAnalyzer           *
*
* generated by the ROME Environment *
*
*           Version 2.4           *
*
*****

Program steering
-----
q : Terminates the program
e : Ends the program
s : Stops the program
r : Restarts the program
c : Continuous Analysis
o : Step by step Analysis
g : Run until event #
i : Root interpreter

Run 1 started
Run 1 stopped

1001 events processed

Writing Root-File ./data/DataTree00001.root

run times :
-----
program..... : 00:00:19:142
DAQ none..... : 00:00:00:022  00:00:00:007
Task PrintADCValues..... : 00:00:16:945  00:00:16:912

dhcp157 % █
```

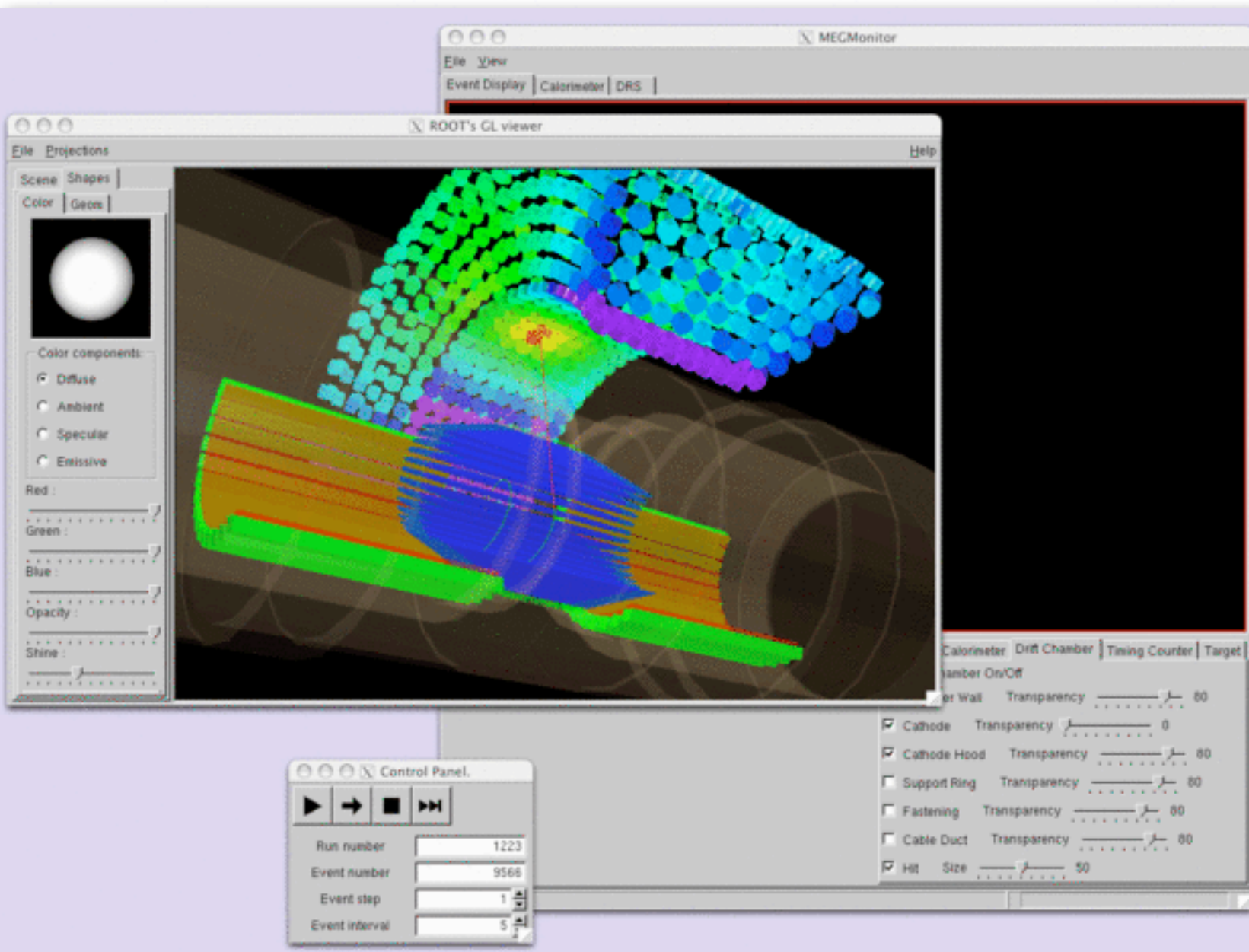
- Analyzer is basically a command line program which reads input files and write result in output files.
- It allows you to go into interactive session.
- It runs also in batch mode.
- You can make GUI which shows result of analysis. (ARGUS is GUI part of ROME)



How your analyzer looks like



- Analyzer is basically a command line program which reads input files and write result in output files.
- It allows you to go into interactive session.
- It runs also in batch mode.
- You can make GUI which shows result of analysis. (ARGUS is GUI part of ROME)





Code generation



How does ROME work ?



Project



MEGANalyzer.xml

Experiment definition XML file

ROME



romebuilder.exe



*Base classes
Utility classes*



How does ROME work ?



Project



MEGAnalyzer.xml
Experiment definition XML file

ROME



romebuilder.exe



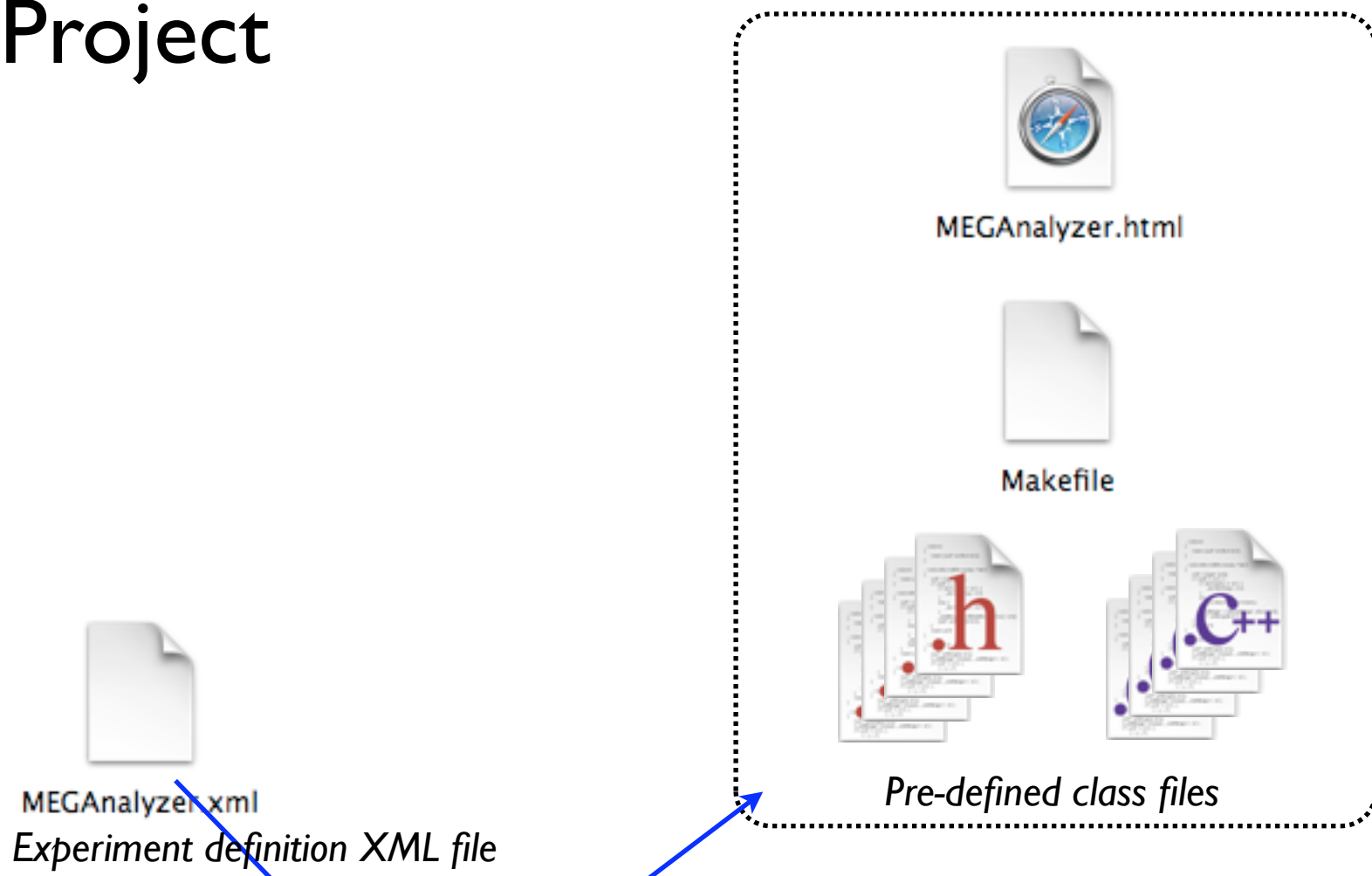
Base classes
Utility classes



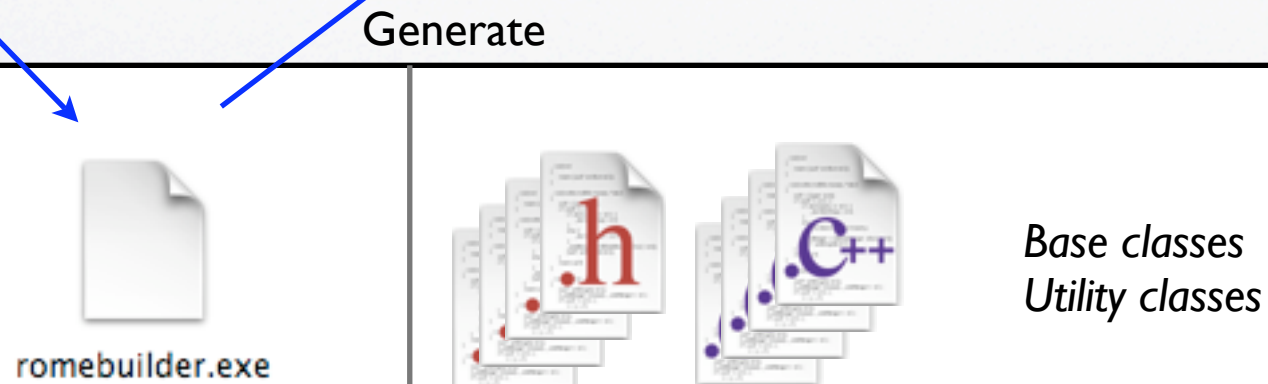
How does ROME work ?



Project



ROME

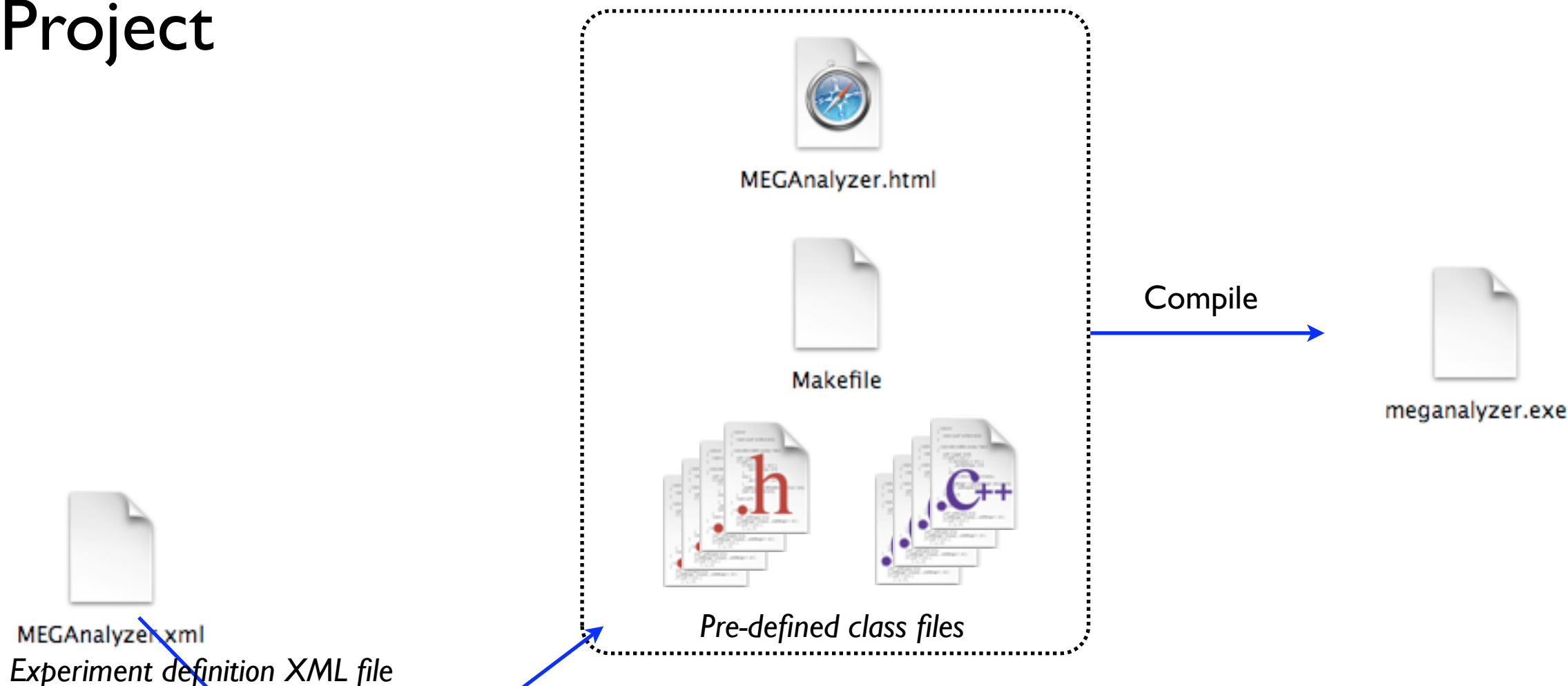




How does ROME work ?



Project



ROME



Generate



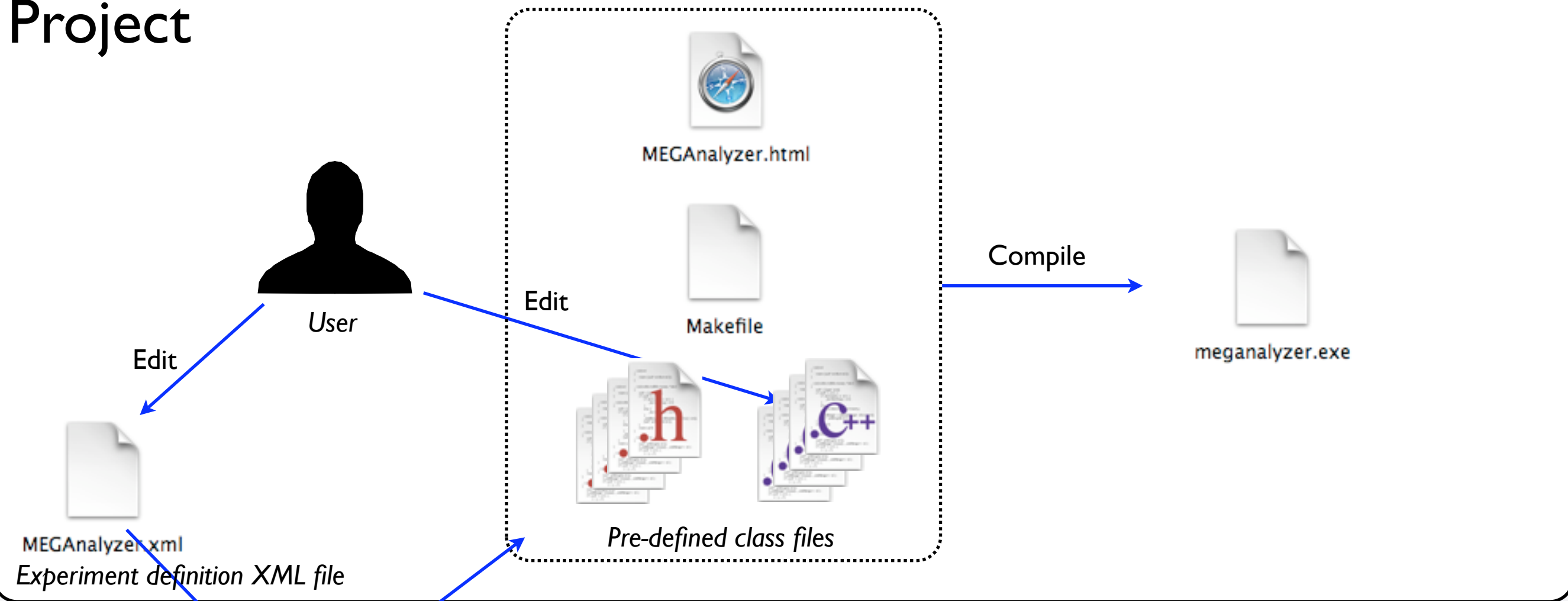
Base classes
Utility classes



How does ROME work ?



Project



ROME





Folder definition



Experiment definition file

```

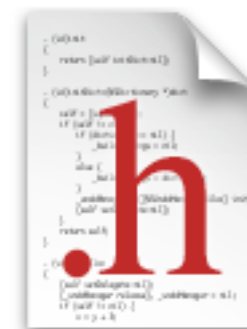
<Folder>
  <FolderName>XEPMT</FolderName>
  <FolderDescription>Data for each PMT.</FolderDescription>
  <Field>
    <FieldName>QE</FieldName>
    <FieldType>float</FieldType>
    <FieldComment>Quantum efficiency</FieldComment>
  </Field>
  <Field>
    <FieldName>Gain</FieldName>
    <FieldType>float</FieldType>
    <FieldComment>PMT gain</FieldComment>
  </Field>
</Folder>

```

updates framework

generates folder class

updates document



MEGAnalyzer.h



MEGXEPMT.h

Always same naming convention of methods to access folders.

`gAnalyzer->GetXEPMT()->GetGain()`

global object

Folder name

Field name

MEGAnalyzer Manual

XEPMT
Data for each PMT.

Fields

Name	Type	Description
QE	float	Quantum efficiency
Gain	float	PMT gain

CMPMTData
Folder Class for Calorimeter Photo Multiplier data.

Fields

Name	Type	Description
------	------	-------------



Task definition



Experiment definition file

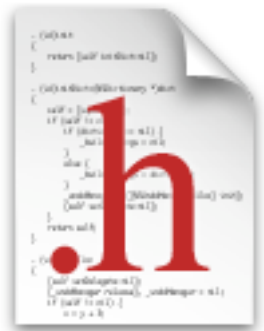
```
<Task>
  <TaskName>XEEnergyRec</TaskName>
  <TaskDescription>Reconstruct energy of gamma.</TaskDescription>
</Task>
```

updates framework



MEGAnalyzer.h

generates task class

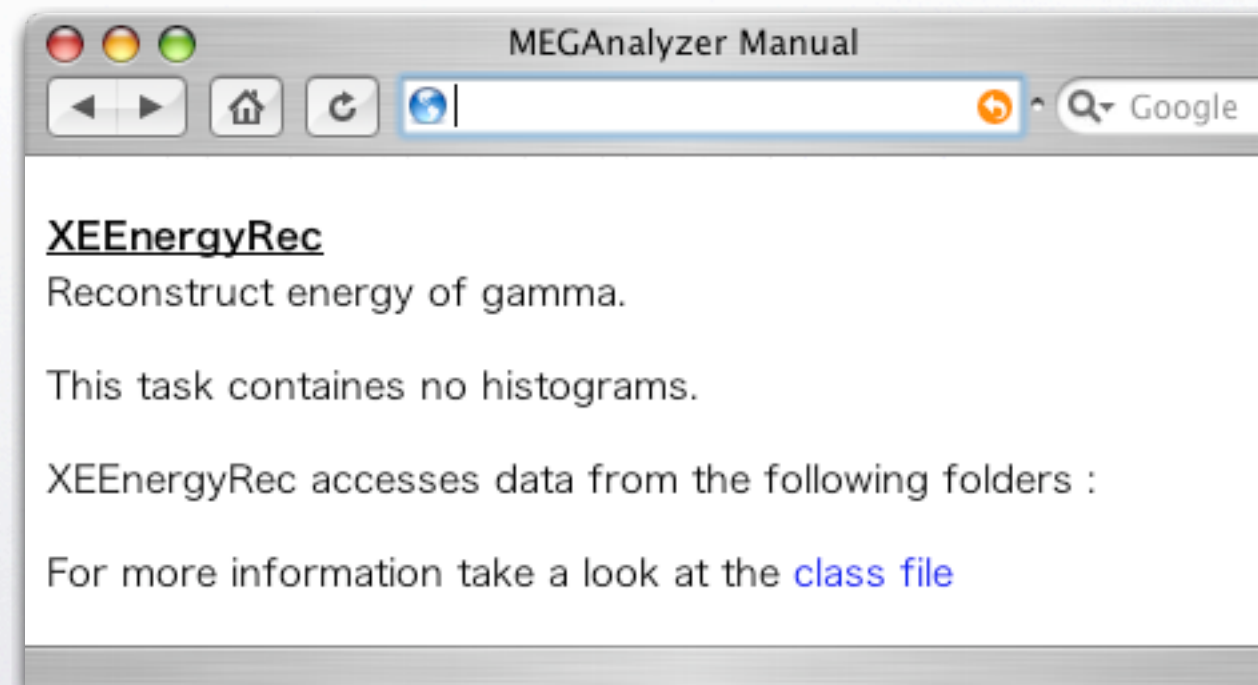


MEGTXEEnergyRec.h



MEGTXEEnergyRec.cpp

updates document



```
void MEGTXEEnergyRec::Init()
{
}

void MEGTXEEnergyRec::BeginOfRun()
{
}

void MEGTXEEnergyRec::Event()
{
}

void MEGTXEEnergyRec::EndOfRun()
{
}

void MEGTXEEnergyRec::Terminate()
{
}
```

User implements actual calculation into generated skeleton.





Auxiliary



Run time configuration



Configuration file allows you to change analyzer's behavior without re-linking.

online/offline ?
input data format.
batch mode or not.

database connection

controlling tasks

controlling trees. (I/O)

user defined steering parameters

```
<Modes>
  <AnalyzingMode>offline</AnalyzingMode>
  <DAQSystem>midas</DAQSystem>
  <BatchMode>>false</BatchMode>
</Modes>
<DataBases>
  <DataBase>
    <Name>MyDataBase</Name>
    <Type>sql</Type>
    <Connection>mysql://username:passwd@server/database</Connection>
  </DataBase>
</DataBases>
<Tasks>
  <Task>
    <TaskName>PrintADCValues</TaskName>
    <Active>>true</Active>
  </Task>
</Tasks>
<Trees>
  <Tree>
    <TreeName>DataTree</TreeName>
    <Read>>false</Read>
    <Write>>true</Write>
    <Fill>>true</Fill>
  </Tree>
</Trees>
<GlobalSteeringParameters>
  <SteeringParameterField>
    <SPName>OutputOnOff</SPName>
    <SPValue>>true</SPValue>
  </SteeringParameterField>
</GlobalSteeringParameters>
```




User defined objects



ROME has flexibility which allows user to customize software.

- User editable [Makefile](#)
- User defined [I/O class](#) for user's input files
- User defined [database](#) class
- [Tasks and folders are editable](#) by users. (ex. adding data members or methods)
- Linking [user defined classes](#)
- [Steering parameters](#) in configuration file. [Command line options](#)

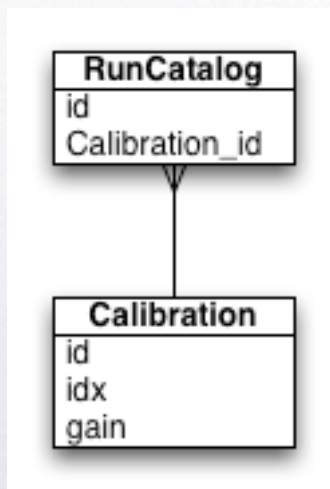


Database



- SQL database is getting popular to store calibration constants and so on.
- ROME has built-in functionality to access SQL database.
- ROME supports MySQL, PostgreSQL, SQLite and more.
- You need not learn SQL programming. We introduced path system to specify variables.

Example: Reading gain of 10 PMTs for run number 100.



```
Read(pContainer, "/RunCatalog[id=100]/Calibration(0,9)/gain");
```

↑
pointer to container to hold values

↑
path

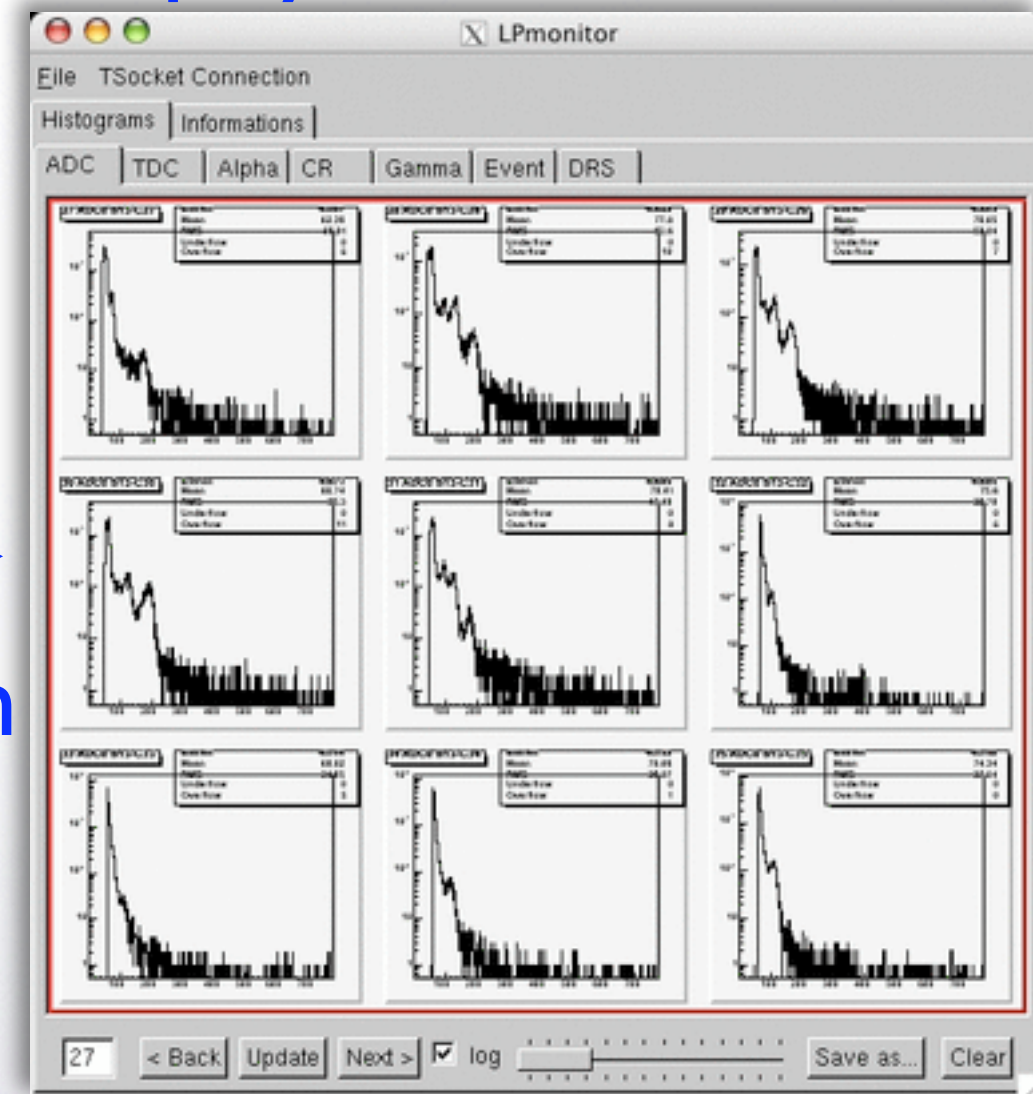


Socket interface



Any ROOT object can be published over network. This feature is useful for online monitor. GUI display is also a ROME project.

display



```

ryu@dhcp157: ~/Analysis/rome_svn/rome/examples/stepbystep
*
* generated by the ROME Environment
*
* Version 2.4
*
*
*****
Program steering
-----
q : Terminates the program
e : Ends the program
s : Stops the program
r : Restarts the program
c : Continuous Analysis
o : Step by step Analysis
g : Run until event #
i : Root Interpreter

Run 1 started
Step by step mode
75 events processed
  
```

analyzer



Summary



- ROME is developed as a **general purpose** software generation tool for elementary particle physics. (Actually, it is currently used not only HEP)
- ROME takes care experiment independent part. What users have to do is basically implementing calculation. It allows you to make analysis software like writing macros.



ROME is introduced in ROOT page

Please visit <http://midas.psi.ch/rome>

The ROOT System Home Page
<http://root.cern.ch/>

- Roadmap
- Mission Statement
- Architecture
- Main Features
- CINT
- Coding Conventions
- Benchmarking
- Picture Gallery
- Publication List
- The ROOT Team

- License
- Register as User
- Download Binaries
- Install from Source
- CVS
- ViewCVS
- LXR
- Nightlies

- User's Guide
- Reference Guide
- Tutorials
- HOWTO's
- RootTalk Forum
- RootTalk Digest
- Example Applications**
- BaBar Tutorials
- FNAL Tutorials
- MINOS Tutorials

- PROOF

Production Version
The production release
LHCb.
Tar files for the source
Version 5.
<http://root.cern.ch/>
The CVS tag for this ve
The AFS versions of 5.

Applications Using ROOT
<http://root.cern.ch/root/ExApplications.html> Google 日本

- **IMV**: Neutrino Interaction Model Validator. Contact person: [Costas Andreopoulos](#)
- **GENIE**: Neutrino Monte Carlo Generator. Contact person: [Costas Andreopoulos](#)
- **INDRA**: ROOT-based simulation/analysis tools developed for charged particle multidetectors such as INDRA, and the study of nuclear multifragmentation. Contact person: [John Frankland](#) **NEW**
- **ROME**: is a framework generator for event based data analysis. It has been developed for the [MEG Experiment](#) at PSI Switzerland. In the ROME environment, the experimenter defines the analysis framework for his experiment in a very clear and compact way in a XML file. Out of this framework definition file, ROME generates all experiment specific classes of the framework. The experimenter only needs to add the calculation code to pre-generated event methods. . Contact person: [Matthias Schneebeli](#) or [Stefan Ritt](#) **NEW**
- **ARGUS**: is a framework generator for graphical display of experiments. It has been developed for the [MEG Experiment](#) at PSI Switzerland. It obtains histograms from MIDAS analyzer or ROME, and obtains data from SQL, XML, ODB database and ROME. ARGUS was designed to be a general purpose framework for online monitor of high energy physics experiment. Once user defines tabs and folders in an XML definition file, ARGUS builder generates prototype of source codes which already have main window creation, connection to histogram/database server and update function of folders. User can design own application with putting graphical objects such as canvases, buttons, sliders and so on. . Contact person: [Ryu Sawada](#) or [Stefan Ritt](#) **NEW**
- **ROODY**: is an application based solely on ROOT for histogram display. It is meant for display of .root files and .hbook files. If running this application in conjunction with either the [Midas analyzer](#) or the [ROME analyzer](#), it will allow to visualize *online* data retrieved through a socket port. . Contact person: [Joe Chuma](#) or [Stefan Ritt](#) **NEW**
- **RooCARDS**: A C++/ROOT interface to SNNS. Steve Sekula (MIT/BABAR?), Andrew Eichenbaum, and Bill Quavle