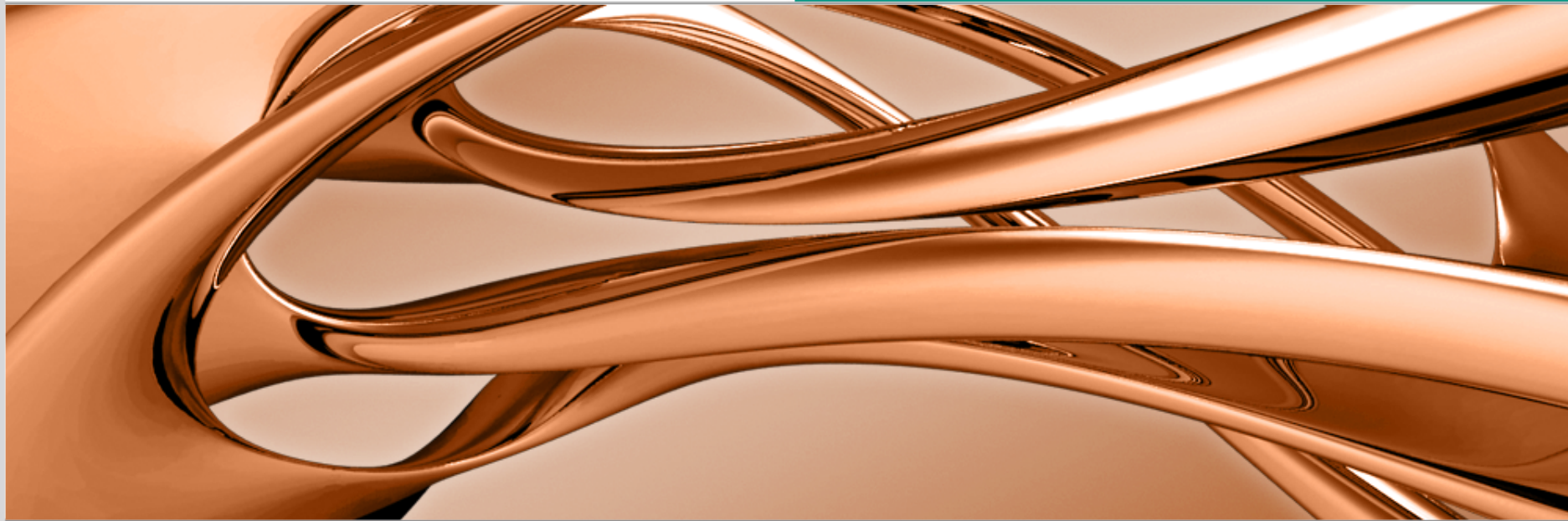


# g-Eclipse tutorial

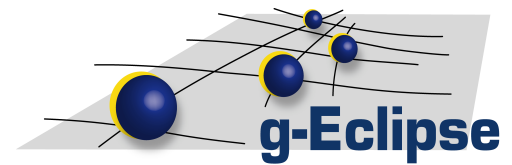
RESPECT tools training session

A. Garcia, S. Girtelschmidt, S. Mueller

EGEE'09 – Barcelona – 20 Sept. 09



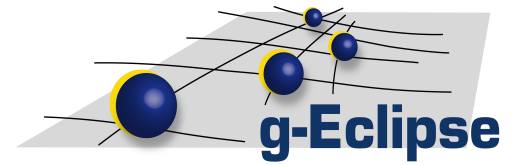
# Outline



- Introduction
  - Why the hell did they do that?, the motivation
  - The two faces of g-Eclipse, the projects
- One client to fit them all?, different Middlewares
- Different usecases?, different Perspectives
- Supported gLite functionality
- Outlook
  
- Developing with g-Eclipse

# Why the hell did they do that?

# The motivation



## ■ The problems

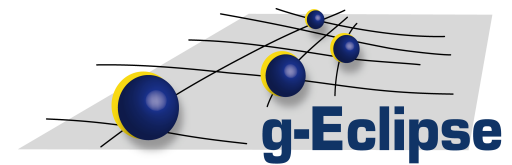
- Dealing with Grids is **difficult**
- The **barrier** for newbies is quite **high**
- There exists no "The Grid" but **different Grids**
- **No generalized** client-side **solution** is out there

## ■ g-Eclipse as a solution

- **User-friendly** graphical client
- **Independent** of underlying Middleware
- Supports **different user groups** (Users, Operators, Developers)

# The two faces of g-Eclipse

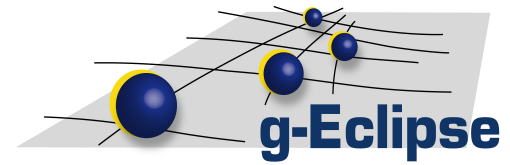
# The g-Eclipse EU project



- R&D funded by European Commission
  - <http://www.geclipse.eu>
- 2 ½ years
- 8 partners in 5 countries
- Around 20 developers
- Monthly release cycle
  - ~ 20 releases
- Final EU-project release in January '09



# The open source project

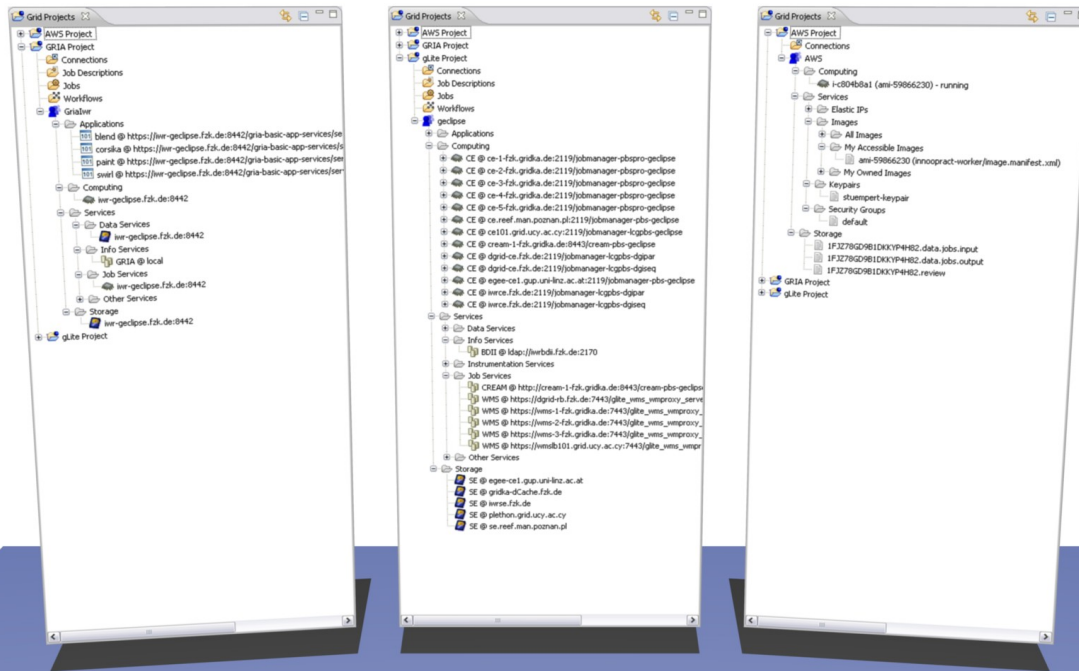
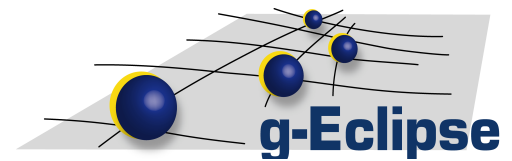


- Eclipse Foundation project
  - <http://www.eclipse.org/geclipse>
  - Eclipse: Java IDE → whole framework, ecosystem
- Since end '06
- **Open source**
- Currently at 1.0 release track
  - **intellectual property** issues being cleared (AWS support)
  - “official” 1.0 release soon

# One client to fit them all?

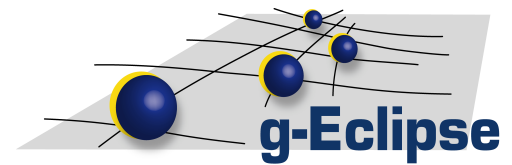


# Middleware



**One single user-interface for several  
Middleware systems!**

# Middleware



- gLite – you know it already :-)



- GRIA

- Service-oriented infrastructure for industry and commerce
- Fully integrated



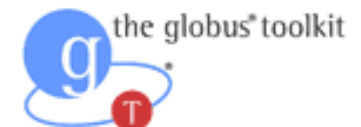
- AWS

- Cloud computing, Amazon
- elastic compute cloud **EC2** and **S3** storage integrated



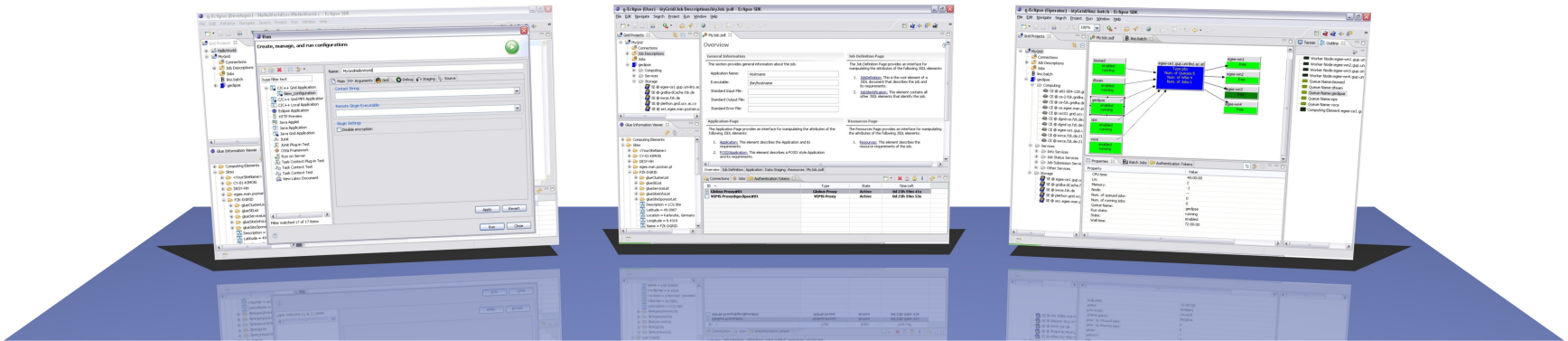
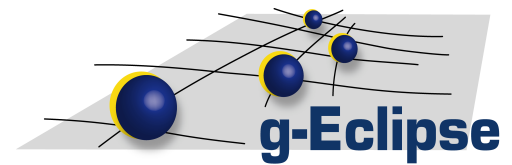
- Globus Toolkit

- GT2 partly integrated
- GT4 in preparation (DORII project)



# Different use-cases?

# Perspectives



## Grid Developer

- Code Applications
- Compile Apps.
- Debug Apps.
- etc.

## Grid User

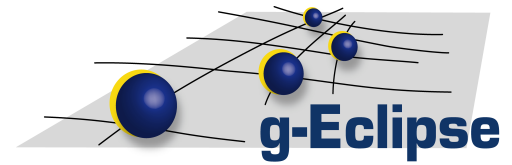
- Data Management
- Job Management
- Visualization
- etc.

## Grid Operator

- Site Administration
- User Administration
- Job Management
- etc.

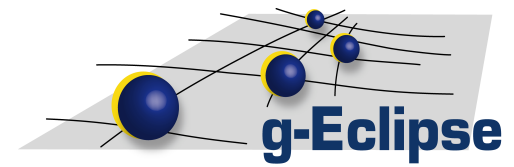
# Supported gLite functionality

# Glite functionality I



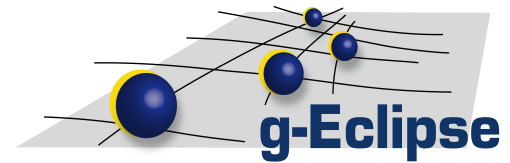
- AAI support
  - Globus proxy creation and management
  - VOMS proxy creation and management
  - CA certificate management
  
- File transfer protocols
  - GSIFTP, based on cog-kit
  - SRM
  - LFC
  - Transfer manager (resume, restart, ... transfers)
  - 3rd party transfers (server-2-server)

# Glite functionality II



- Job description languages
  - JSDL and JDL supported
  - Parametric jobs supported
  - Workflows supported
  
- Job submission services
  - WMS
  - Cream
  
- Job monitoring
  - L&B

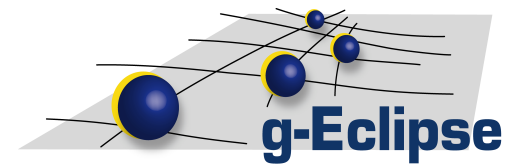
# The gLite user perspective



- Data management
  - Files/Folders create/edit/save/copy/move/delete
  - 3rd party transfers (in preparation)
- Job management
  - Job description creation and editing
  - Job submission
  - Job monitoring
- Workflows
  - Creation, using dedicated workflow editor
  - Submission and status, just like an ordinary job
- Data visualization
  - Using VTK

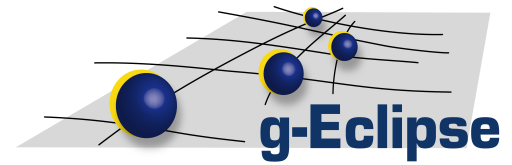


# The gLite operator perspective



- Site administration
  - Queue management
  - Job management
  - Infrastructure monitoring
  - Infrastructure testing
  - Infrastructure benchmarking
  - Service level agreements
  
- User administration
  - VO management
  - Access Control

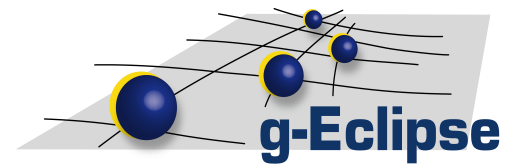
# The gLite developer perspective



- Application development
  - Remote compiling
  - Remote debugging
  - Analyzing MPI applications
- Application deployment
  - VO based deployment
  - Binary deployment
  - Source code deployment

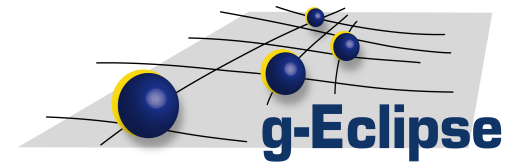
# Outlook

# Outlook



- g-Eclipse **keeps going as an Eclipse project**
- Developers reassigned tasks, less manpower
- BUT developers stay Committers of the Project
- Several developers assigned to projects which also deal with g-Eclipse (e.g. DORII)
- **ongoing support**
  
- Next project proposals ongoing, ICT call 2009,...
- **open source**, anybody can join
  - continue gathering community, users & developers

# Finally ...



## ■ EVERYBODY can contribute ...

- We welcome reuse and new developments
  - use of g-Eclipse as a library
    - for RCP applications
    - for server-side services which need Grid access
    - ...
  - new middleware implementations
  - new components

## ■ YOU can contribute ...

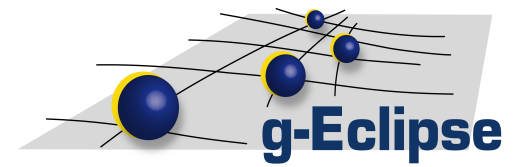
- Report bugs or request features
  - <http://bugs.eclipse.org>
- Contact the team
  - [geclipse-user@eclipse.org](mailto:geclipse-user@eclipse.org)
  - [geclipse-dev@eclipse.org](mailto:geclipse-dev@eclipse.org)
  - [contact@geclipse.eu](mailto:contact@geclipse.eu)
- Contribute source code
  - bug fixes
  - improvements
  - new features

# Any questions?

## Let's start the hands-on!

# Developing with g-Eclipse

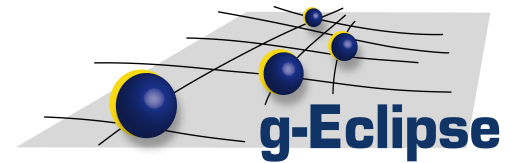
# g-Eclipse domains



- User interface / Grid client
  - graphical user interface for accessing Grid infrastructures
- Framework / API
  - collection of pure Java classes for developing client- and server-side applications for the Grid



# Developing on g-Eclipse

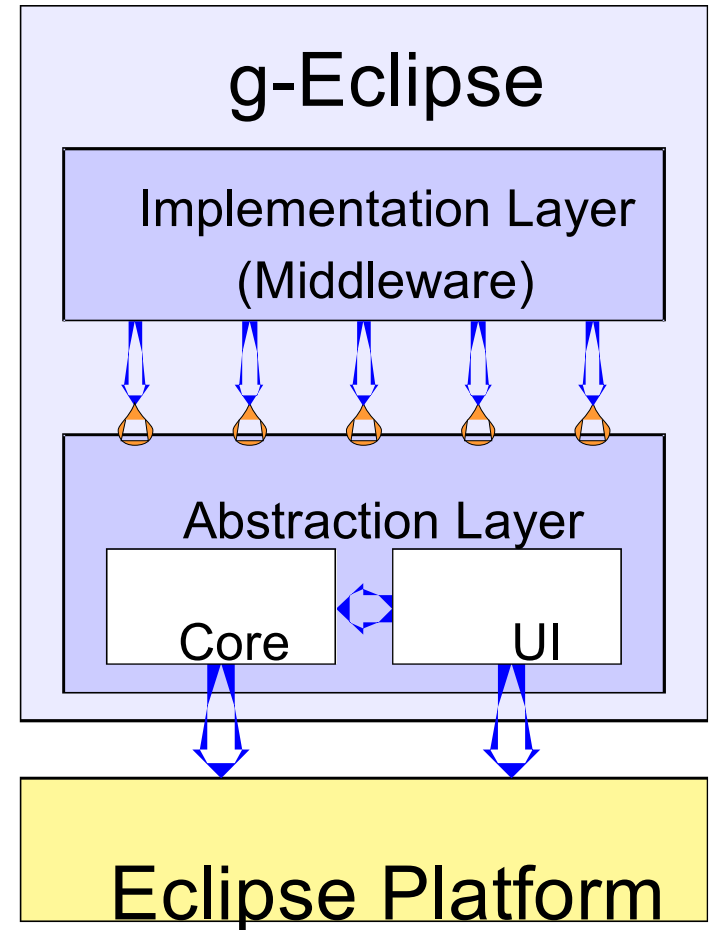


## ■ Abstraction layer

- core functionality, e.g.
  - authentication/authorization
  - VO management
  - data management
  - job submission
- common user interface, e.g.
  - views
  - wizards
  - dialogs
  - preference pages

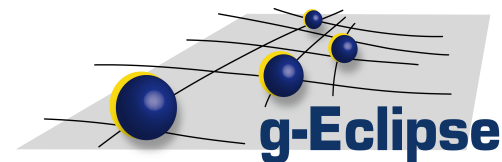
## ■ Implementation layer

- implementing core functionality
- middleware specific functionality

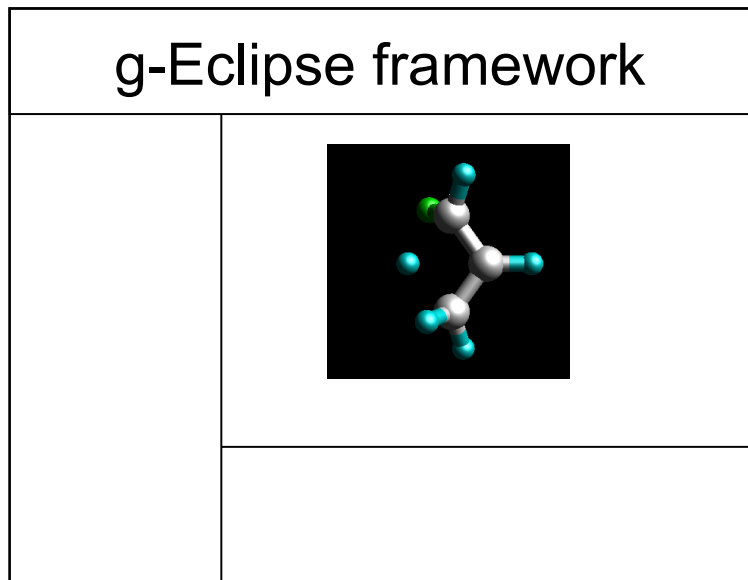


Eclipse Extension Point

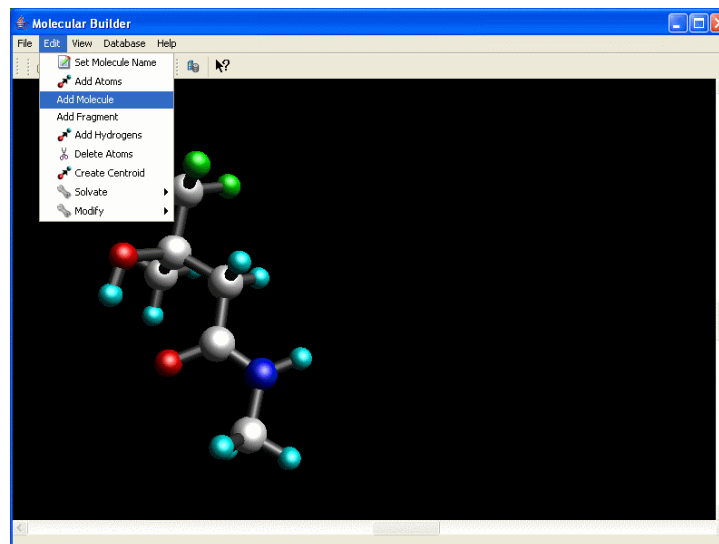
# Applications with g-Eclipse



- Application inside g-Eclipse



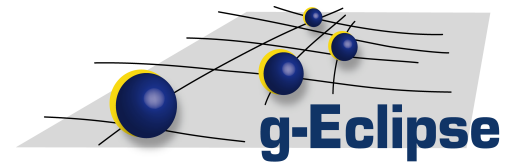
- Application on top of gEclipse



g-Eclipse API

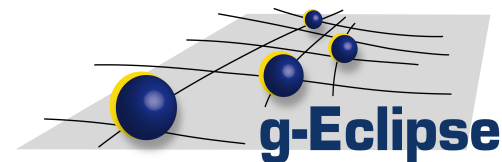
**GRID**

# Application inside g-Eclipse



- Application is plugged into g-Eclipse framework
- Benefit from the user-friendly graphical interface for accessing Grid infrastructures
- Only few optional elements should be added:
  - editor for input files/parameters
  - submission support
  - data visualisation

# Application inside g-Eclipse



The screenshot displays the g-Eclipse application window. The top menu bar includes File, Edit, Navigate, Search, Project, Run, Window, and Help. The main workspace is divided into several panes:

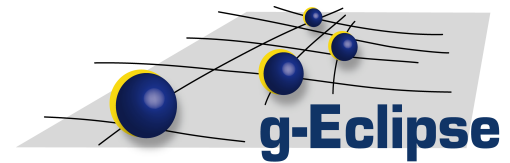
- Grid Projects:** A tree view on the left showing the project structure, including Gaussian, Connections, Job Descriptions (with CH2F.gjf selected), Jobs, Workflows, and Applications.
- GJF text editor:** A central text editor displaying the contents of CH2F.gjf, which includes a Gaussian job description for a frequency calculation on a CH2F-CH=CH2 molecule. The text includes the molecule name, a title, and a list of Cartesian coordinates for the atoms.
- 3D visualization:** A large window at the bottom showing a 3D ball-and-stick model of the CH2F-CH=CH2 molecule, with carbon atoms in white, hydrogen in light blue, and fluorine in green.

Red arrows point from text boxes to these components:

- G-Eclipse framework:** Points to the top menu bar.
- GJF text editor:** Points to the central text editor.
- 3D visualization:** Points to the 3D molecular model.

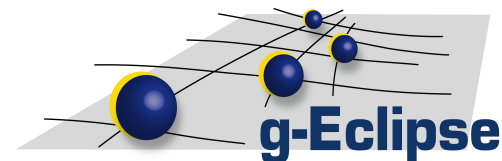
- Grid job description is created from GJF file on the fly

# Application on top of g-Eclipse



- Enhance existing applications with Grid support
- Application has its own GUI and calls g-Eclipse API for accessing Grid resources.
- Application is started as Eclipse Rich Client Application
- Can access other bundles provided by g-Eclipse

# Application on top of g-Eclipse



JMOL application

Data provided by G-Eclipse libraries

Action will be delegated to g-Eclipse libraries