



**ETICS**  
*The Grid Quality Process*

# ETICS Service Overview

*Marc-Elian Bégin*  
*ETICS Project*  
*CERN*

[www.eu-etics.org](http://www.eu-etics.org)



- **Introduction**
- **Typical usecase**
- **Service**
- **Conclusions**

# Introduction

- **ETICS stands for e-Infrastructure for Testing, Integration and Configuration of Software**
- **Project started on January 1<sup>st</sup>, 2006 and has a duration of two years**
- **Goal is to provide efficient automated tools for building and testing**
- **ETICS is born from the experience of developing, integrating and testing grid and distributed software**



Build system,  
software  
configuration, service  
infrastructure,  
dissemination, EGEE,  
gLite, project coord.

Software  
configuration,  
service  
infrastructure,  
dissemination

The Condor batch  
system, distributed  
testing tools, service  
infrastructure, NMI

Web portals and  
tools, quality  
process,  
dissemination,  
DILIGENT

Test methods and  
metrics, unit testing  
tools, EBIT



# Usecases

- **Context: A user develops, for example, a Grid service (as a single component)**
- **This service relies on existing software (dependencies)**
- **User performs following actions**
  - For each required dependencies
    - Find documentation
    - Identify correct version to use (release, maintenance, beta, etc)
    - Identify if a binary package is adequate
    - In case building is required (very often the case), find build instructions
    - Build code
    - Install and configure code (using available documentation)
    - Verify that the code is built, installed and configured correctly
    - Find developer contact information in case of problems
  - Inter-dependencies must be understood and version clashes resolved
  - Write a script to build all dependencies
  - Write a script to build the current component (Grid service), including fetching code, build and test instructions/command
  - Store the component in a repository for others to use
- **This above process has to be executed, in full or in part, for each targeted platforms**
- **For each platform, a machine has to be setup and maintained**
- **If regular builds and tests are required, a scheduled task must be setup for each required platform**
- **Each version of the software must be advertised to potential user communities**

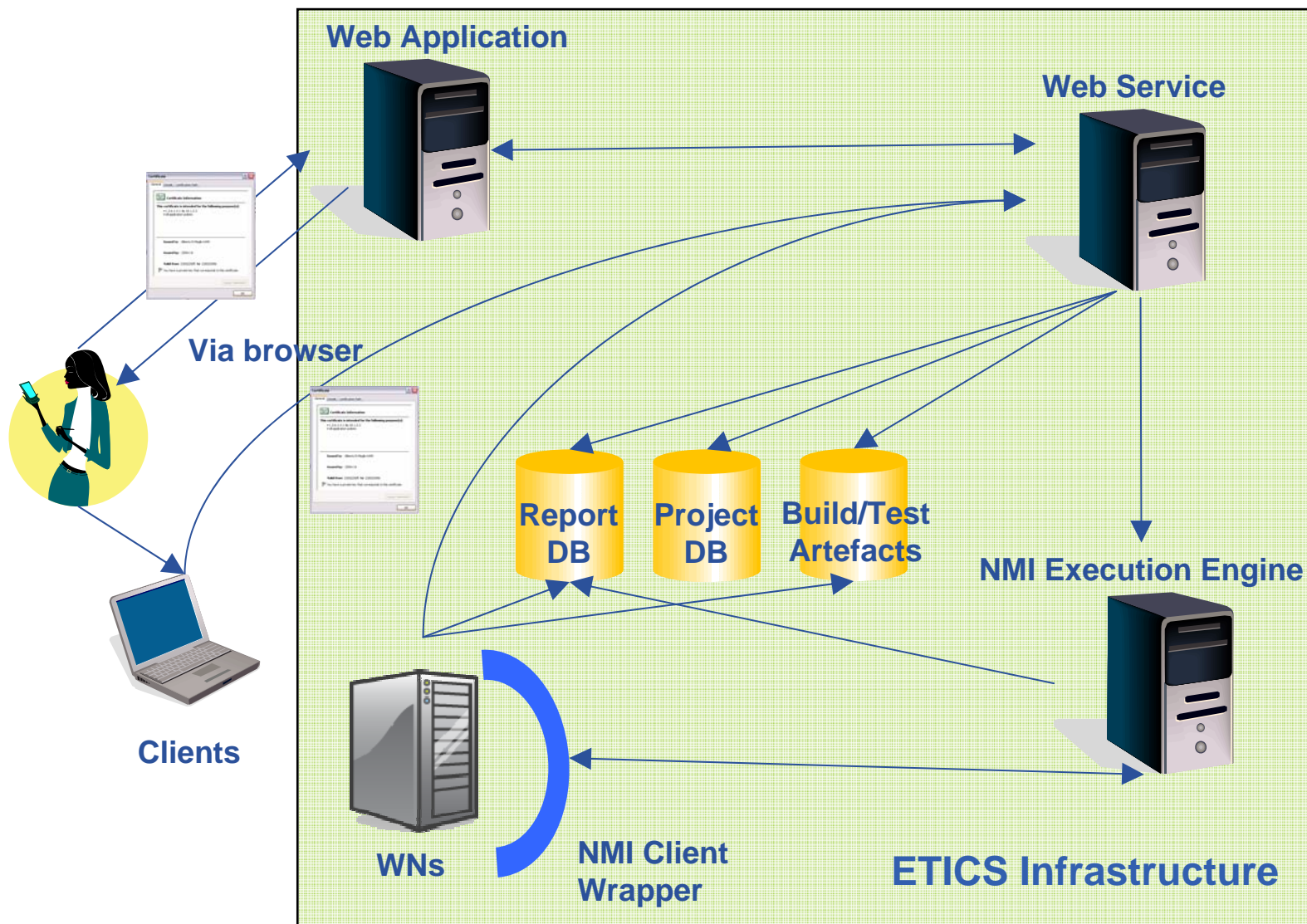
- **The above usecase can take between a few hours to a few weeks, depending on the number of dependencies**
- **When complexity grows, so does the time required to get started**
- **Different software teams will define the required information differently, which takes longer for the developers to integrate their work**
- **Unless well controlled, builds and tests might not be reproducible**
- **Significant cost in maintaining different machines for build and test**



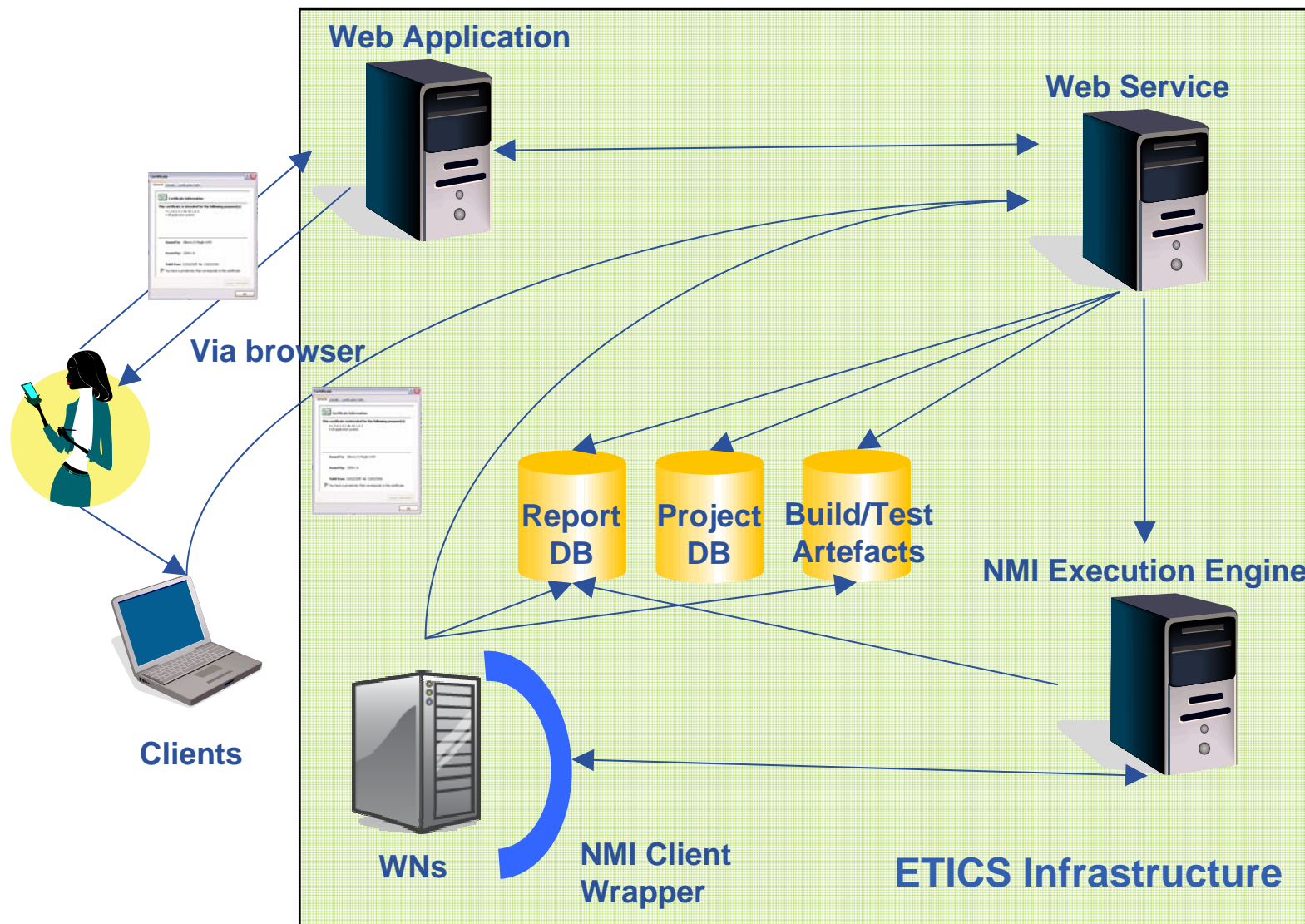
- **Context: A user develops, for example, a Grid service (as a single component)**
- **This service relies on existing software (dependencies)**
- **Using ETICS, user:**
  - Enters the commands to checkout, build and test the component
  - Selects the dependencies this component requires (from the rich catalog of existing components in ETICS)
  - Builds and tests the component locally
  - Submit build and test for parallel execution on all required platforms
  - All build and test artefacts are registered in the ETICS repository for public access

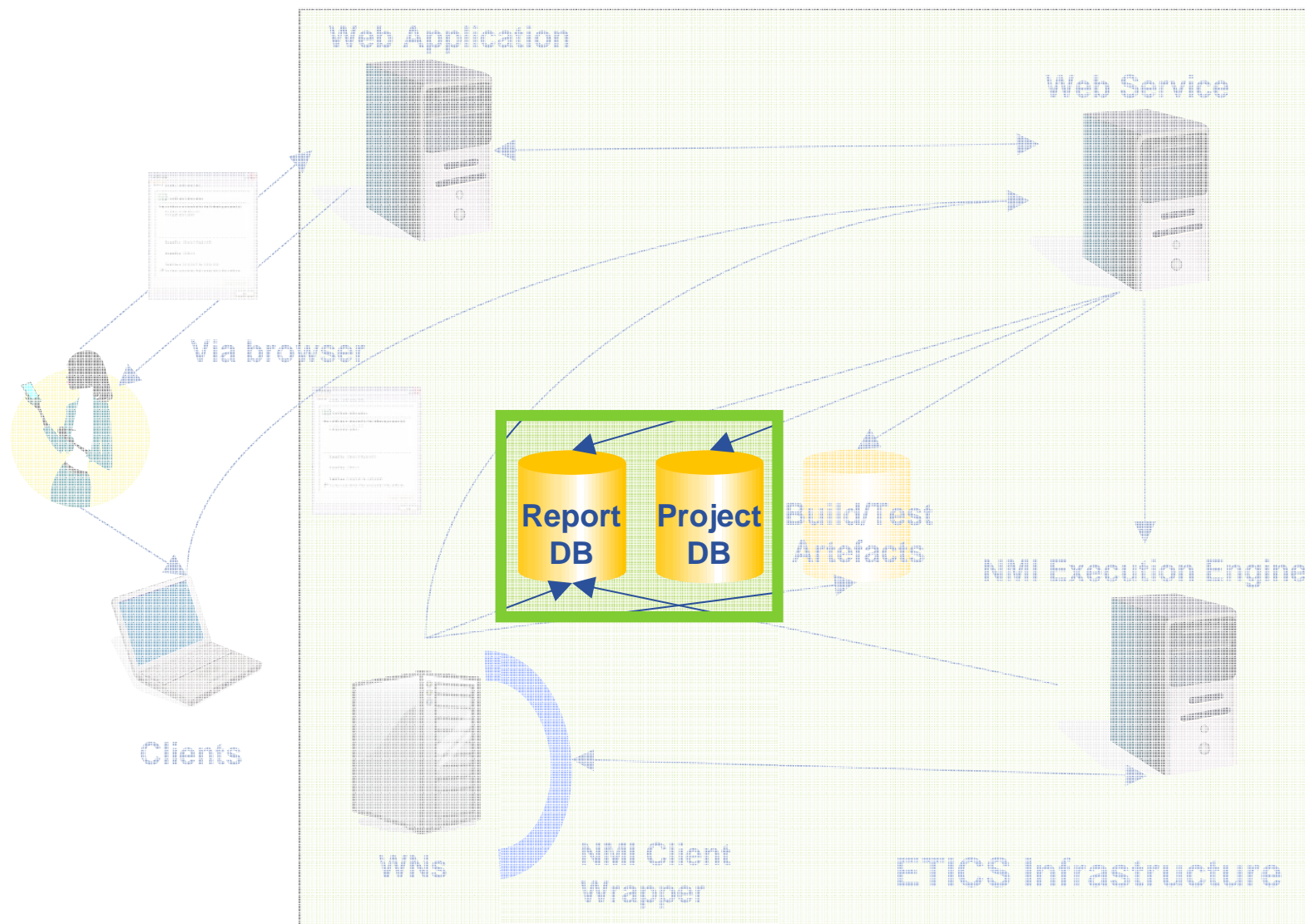
# Service

- Out-of-the-box **automatic build/test** system
- **Multiplatform support** for development and testing
- Powerful and flexible **automatic dependency management** (internal and external dependencies)
- Trigger coding convention checks, unit test, documentation tools and collect and publish the results
- Possible to build everything **from source** or use pre-built **binary packages**
- Designed to **support several Version Control Systems**
- Possible to **register artefacts on the repository**
- **Publishes run-time configuration information** such as environment variables, runtime dependencies, etc ...
- Produce **different package formats** (rpms, tar, msi, debs)
- Automatic **deployment, configuration** and **test execution**
- Produces **rich build and test reports**
- Ability to run **custom plugins** during build and test procedure



- **ETICS Service build as an N-tier architecture**
- **All elements of ETICS are configured, built and tested with ETICS**
- **ETICS Service is Open Source: Apache 2.0 license**

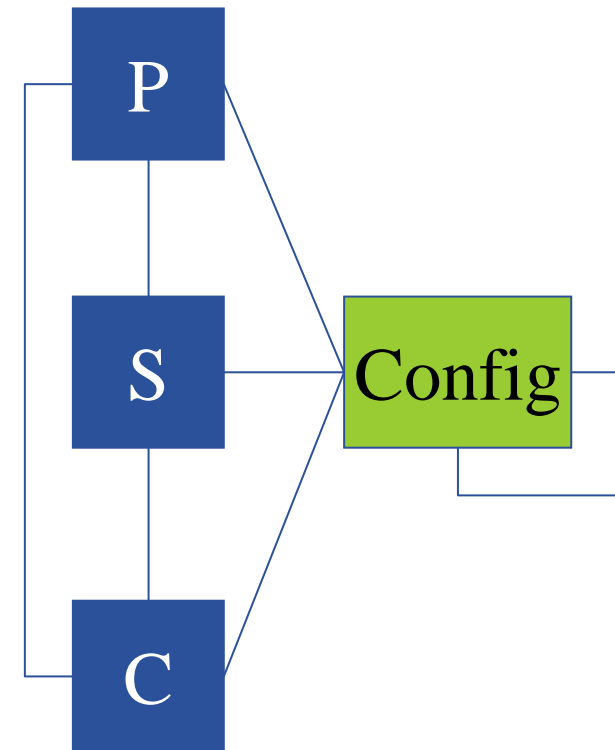




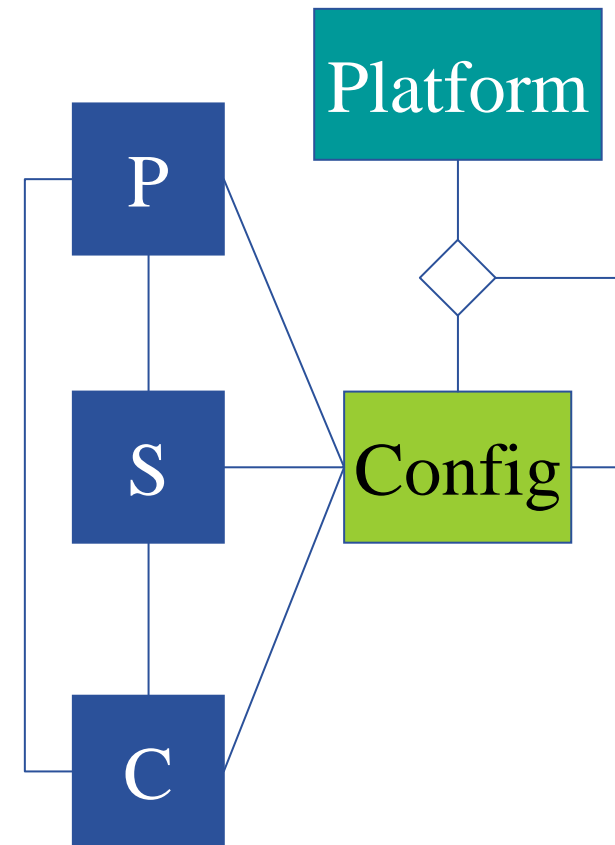
- **In order to understand how to build and test, ETICS relies on metadata**
- **ETICS data-model (defining metadata) is loosely based on a subset of CIM and OMG/MDA (CIM and OMG/MDA models are complex)**
- **Simple to view and edit by the user, this metadata allows the ETICS Service to perform complex reasoning, hence performing high-level functions on behalf of the user – e.g.**
  - Dependency analysis
  - Property processing



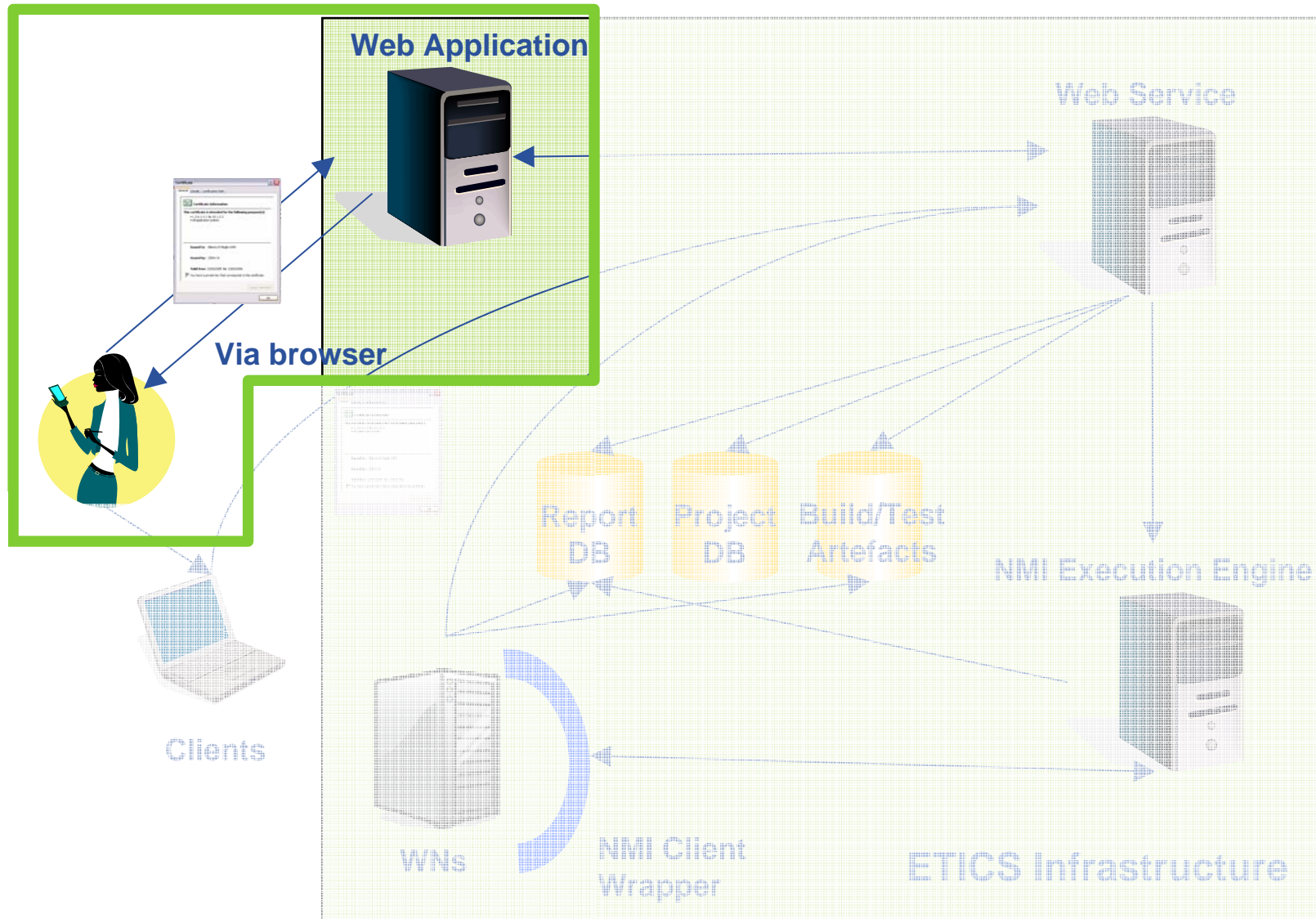
- **Project:** container for components and subsystems
- **Subsystem:** container for components
- **Component:** a module configuration can also be made of sub-module's configurations
- **Configuration:** different versions of the modules (e.g different CVS tags, dependencies, platforms)



- **Platform**: represents an specific combination of *architecture, operating system* and *C/C++ compiler*
  - E.g. *slc3\_ia32\_gcc323*
- **Allows users to specify platform specific metadata**
  - **VCS commands (e.g. CVS)**
  - **Build commands**
  - **Test commands**
  - **Properties**
  - **Environment variables**
  - **Dependencies**



- **Metadata is persisted in a MySQL 5 database**
- **For future scalability, the ETICS database could upgrade to other database implementation – e.g. Oracle**
  - No MySQL specific feature used
  - Load balancing possible to reduce load on single data server
  - Oracle not used by default, since not Open Source
- **Tools available to dump and move ETICS database**
  - Support relocation
  - System upgrade

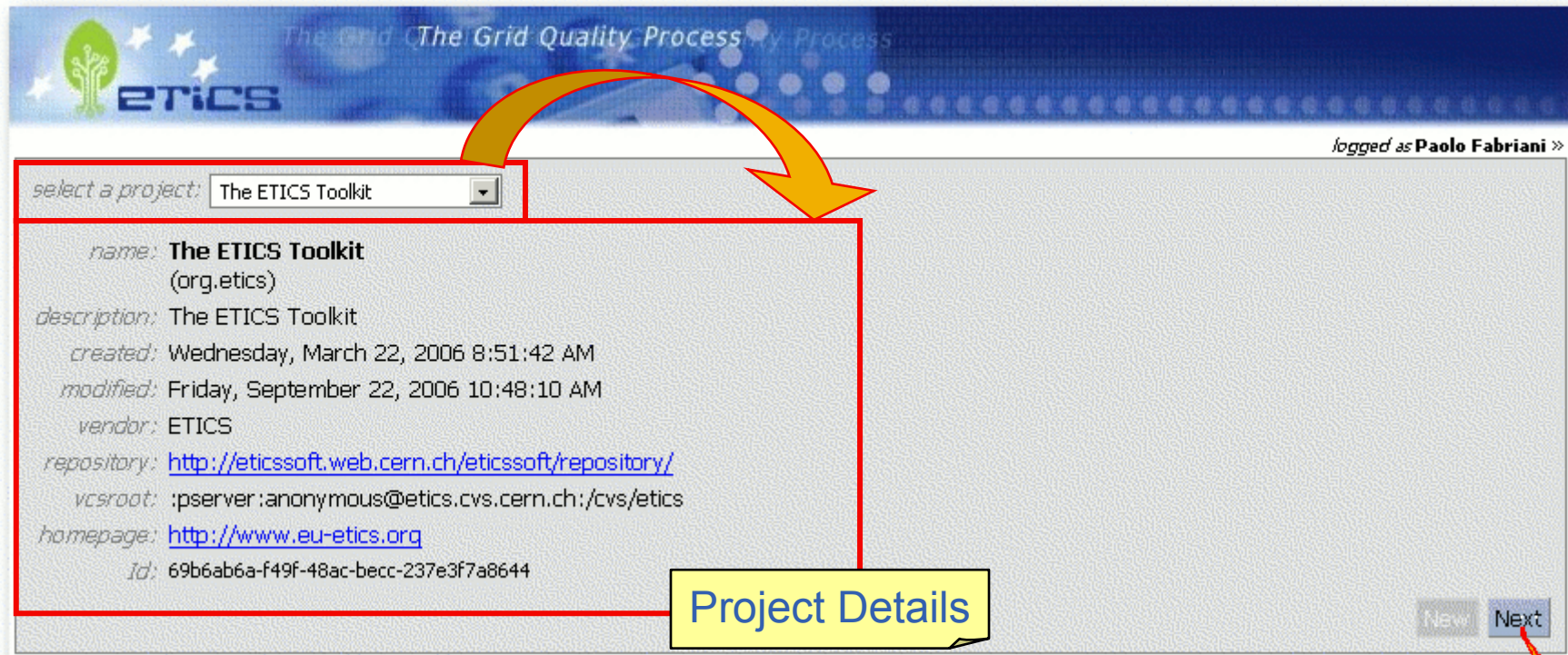


- **ETICS composed of following Web Applications:**
  - Build and Test
  - Repository
  - Administration
- **The applications are being integrated under a single user experience**

- **Build and Test Web Application (WA) allows users to**
  - Browse metadata
  - Edit metadata
  - Submit build and test jobs
  - Compatible with Firefox browser
- **State-full and secure application**
- **WA written in Java, jsp, javascript and JDuck (Engineering client/server web framework)**
- **Interfaces with the ETICS Build and Test Web Service**

- **Web Application is the simplest way to understand ETICS**
- **We will now go through a few simple steps to explain how ETICS works using the Build and Test Web Application**
- **Note: Much more details will be provided during the demo**

- All ETICS registered projects are available from a central entry point: <https://etics.cern.ch:8443/etics>
- Project details are shown upon selection in a drop-down list

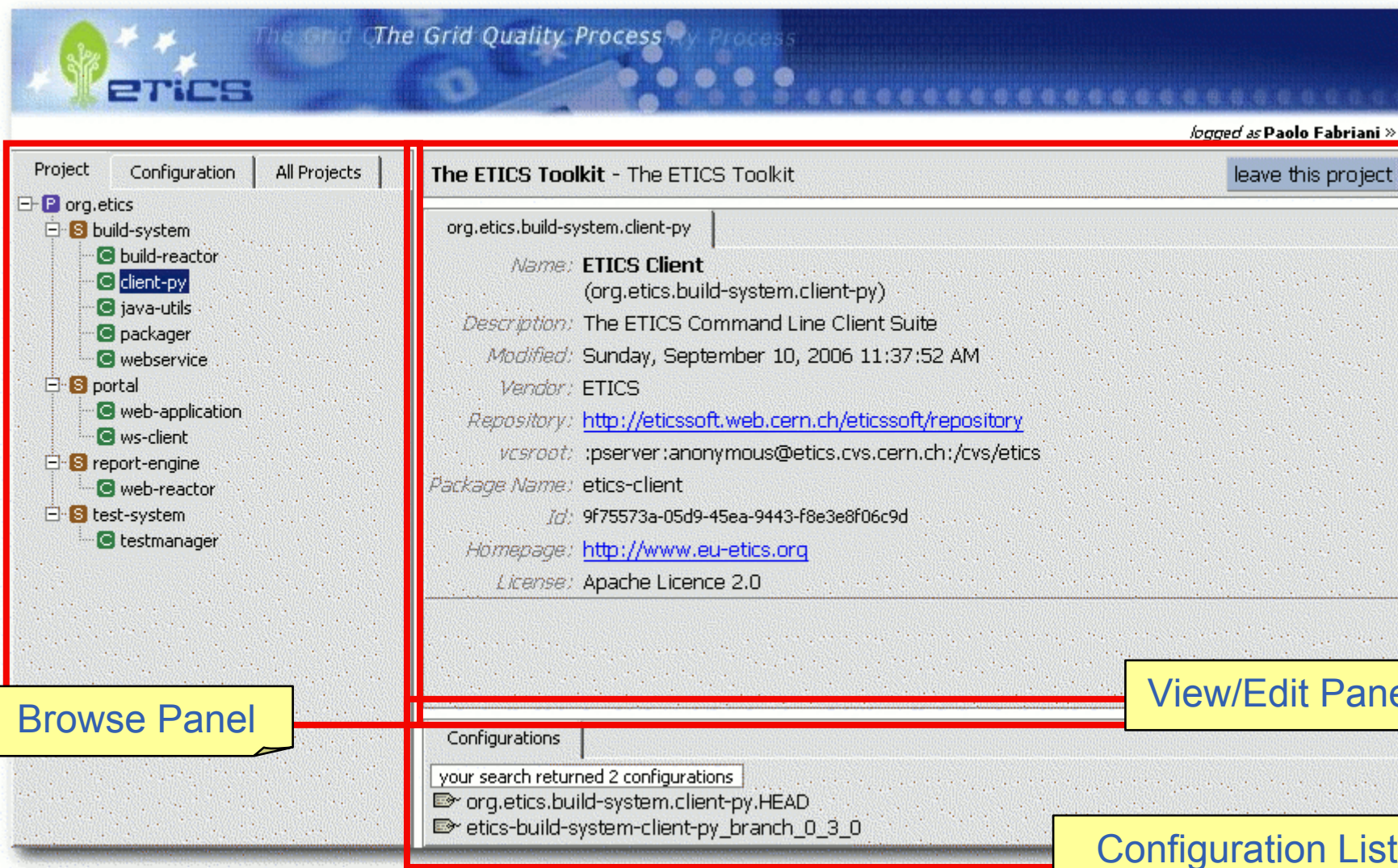


The screenshot shows the ETICS web interface. At the top, there is a header with the ETICS logo and the text "The Grid (The Grid Quality Process) Process". Below the header, there is a navigation bar with the text "logged as Paolo Fabiani »". The main content area features a dropdown menu labeled "select a project:" with "The ETICS Toolkit" selected. Below the dropdown, a red box highlights the project details, which are as follows:

*name:* **The ETICS Toolkit**  
(org.etics)  
*description:* The ETICS Toolkit  
*created:* Wednesday, March 22, 2006 8:51:42 AM  
*modified:* Friday, September 22, 2006 10:48:10 AM  
*vendor:* ETICS  
*repository:* <http://eticssoft.web.cern.ch/eticssoft/repository/>  
*vcsroot:* :pserver:anonymous@etics.cvs.cern.ch:/cvs/etics  
*homepage:* <http://www.eu-etics.org>  
*Id:* 69b6ab6a-f49f-48ac-becc-237e3f7a8644

A yellow box labeled "Project Details" is positioned below the red box, with a red arrow pointing from the dropdown menu to it. In the bottom right corner of the interface, there are "New" and "Next" buttons, with a red arrow pointing to the "Next" button.





The screenshot shows the ETICS web interface. At the top, there is a header with the ETICS logo and the text "The Grid (The Grid Quality Process) by Process". The user is logged in as "Paolo Fabriani".

The main content area is divided into two panels:

- Browse Panel:** A tree view on the left showing the project structure under "org.etics". The "client-py" sub-project is selected.
- View/Edit Panel:** A detailed view of the selected project, "The ETICS Toolkit - The ETICS Toolkit". It displays the following metadata:
  - Name:** ETICS Client (org.etics.build-system.client-py)
  - Description:** The ETICS Command Line Client Suite
  - Modified:** Sunday, September 10, 2006 11:37:52 AM
  - Vendor:** ETICS
  - Repository:** <http://eticssoft.web.cern.ch/eticssoft/repository>
  - vcsroot:** :pserver:anonymous@etics.cvs.cern.ch:/cvs/etics
  - Package Name:** etics-client
  - Id:** 9f75573a-05d9-45ea-9443-f8e3e8f06c9d
  - Homepage:** <http://www.eu-etics.org>
  - License:** Apache Licence 2.0

Below the main view, there is a "Configurations" section showing a search result: "your search returned 2 configurations". The list includes:

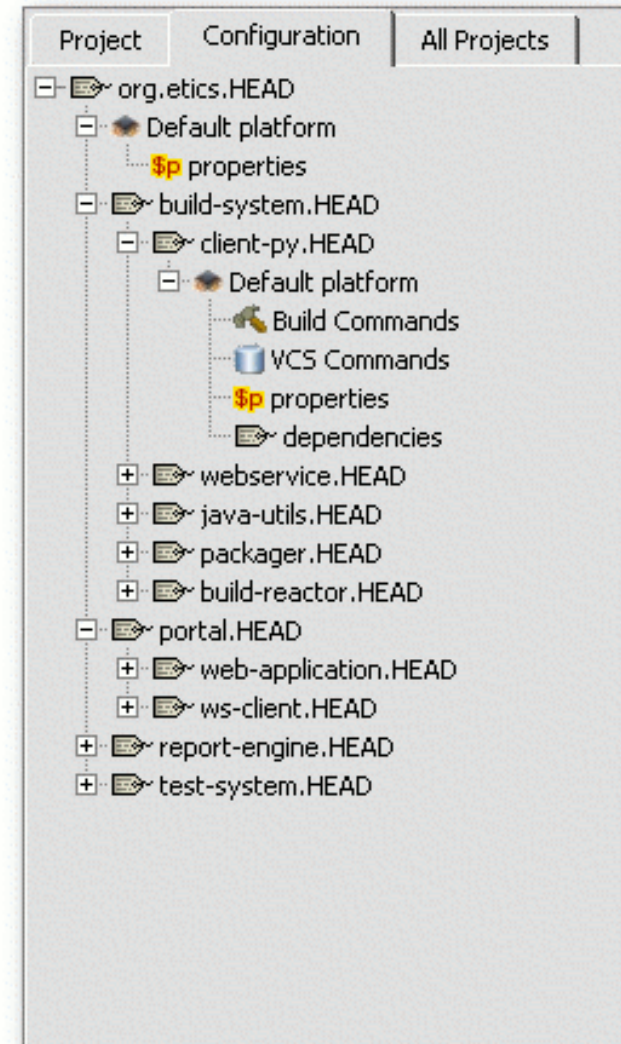
- org.etics.build-system.client-py.HEAD
- etics-build-system-client-py\_branch\_0\_3\_0

Browse Panel

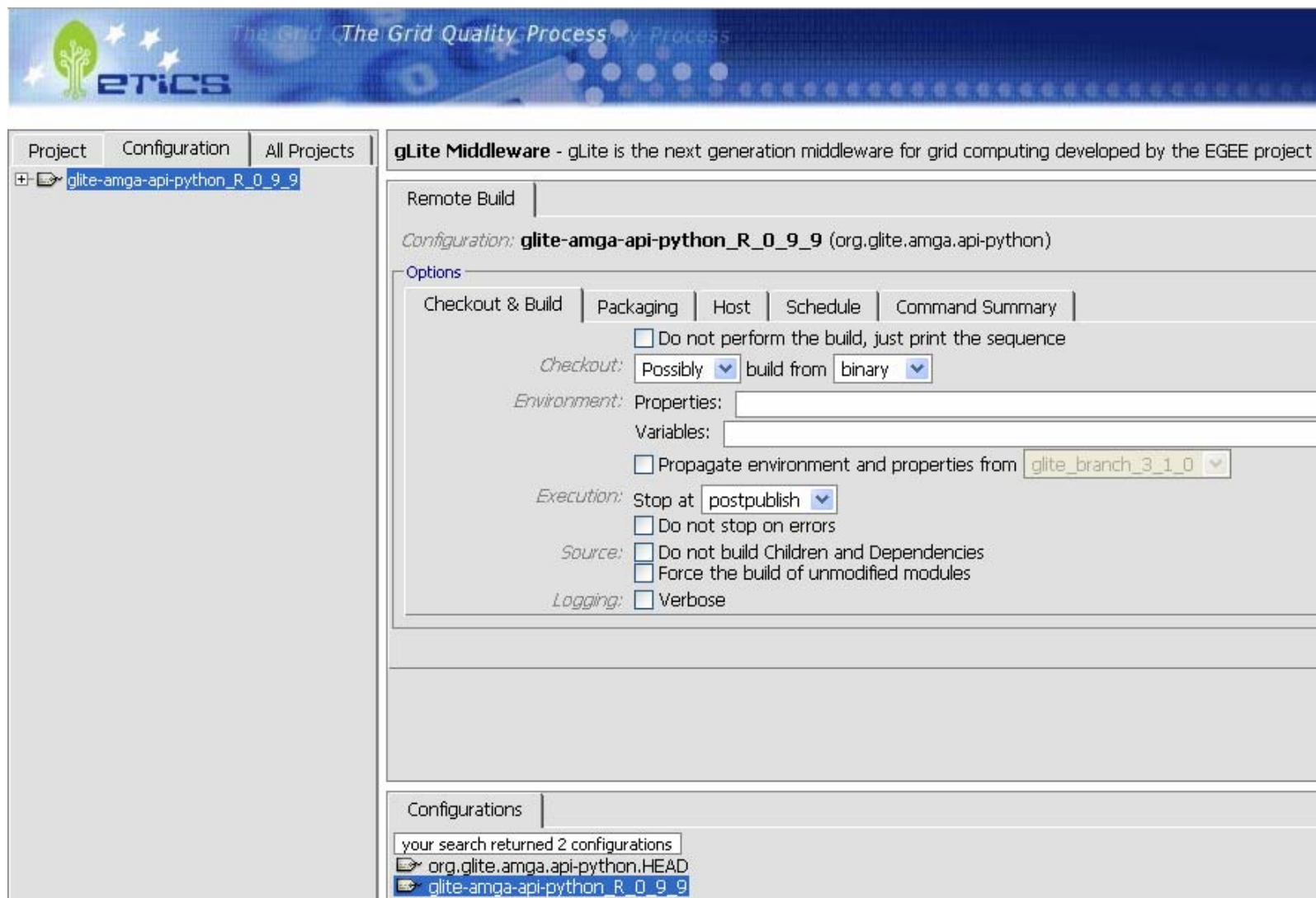
View/Edit Panel

Configuration List

- **ETICS defines *configuration* similar to version in CVS**
- **Configurations can be defined differently for different *platforms***
- **Configurations include what to do to checkout, build and test software modules:**
  - Version control system (VCS) - e.g. CVS, SVN - commands
  - Build commands
  - Test commands
  - Properties and environment
  - Dependencies
- **Properties and environment variables are *inherited and propagated***
- **Each command includes clear set targets.**  
**e.g. build: clear, init, checkstyle, doc, compile, package, publish**



- Here are a few examples
- **Version Control System Commands:**
  - Checkout: `cv`s -d `${vcsroot}` co `${moduleName}`
- **Build Commands:**
  - Init: `mkdir -p etc; cp ${security.location}/etc`
  - Configure: `./configure`
  - Compile: `ant -f build.xml`
- **Test Commands:**
  - Init: `./scripts/install-script ${target-hostname}`
  - Test: `./scripts/test-script ${reportDir}`



The screenshot shows the ETICS web interface for configuring a remote build. The top banner includes the ETICS logo and the text "The Grid (The Grid Quality Process) by Process".

The main content area is titled "gLite Middleware - gLite is the next generation middleware for grid computing developed by the EGEE project". Below this, the "Remote Build" section is active, showing the configuration for "glite-amga-api-python\_R\_0\_9\_9 (org.glite.amga.api-python)".

The "Options" section contains several tabs: "Checkout & Build", "Packaging", "Host", "Schedule", and "Command Summary". The "Checkout & Build" tab is selected and contains the following options:

- Do not perform the build, just print the sequence
- Checkout: Possibly (dropdown) build from binary (dropdown)
- Environment: Properties: (text input)
- Variables: (text input)
- Propagate environment and properties from glite\_branch\_3\_1\_0 (dropdown)
- Execution: Stop at postpublish (dropdown)
- Do not stop on errors
- Source:  Do not build Children and Dependencies
- Force the build of unmodified modules
- Logging:  Verbose

At the bottom, the "Configurations" section shows a search result: "your search returned 2 configurations". The results are:

- org.glite.amga.api-python.HEAD
- glite-amga-api-python\_R\_0\_9\_9 (selected)

**Project name:** org.etics  
**Module name:** org.etics.build-system.plugin-framework  
**Module config:** etics-build-system-plugin-framework\_B\_1\_1\_0  
**Build status:** **Success**  
**Description:** Plugin framework for the client  
**Dependencies**

Page generated at 06/02/

[Back to m](#)

### Checkout / download log

```
02/06/07 00:57:05.209 WARNING main [_doCheckout] - Trying with sources from VCS...
02/06/07 00:57:05.216 INFO main [write] - Checking out configuration 'etics-build-system-plugin-framework_B_1_1_0'
02/06/07 00:57:05.236 INFO main [write] -
02/06/07 00:57:24.082 INFO main [write] - Module 'org.etics.build-system.plugin-framework' is up-to-date with respect to CVS for tag
'etics-build-system-plugin-framework_B_1_1_0'... skipped
02/06/07 00:57:24.089 INFO main [write] -
02/06/07 20:08:36.268 WARNING main [_doCheckout] - Trying with sources from VCS...
02/06/07 20:08:36.290 INFO main [write] - Checking out configuration 'etics-build-system-plugin-framework_B_1_1_0'
02/06/07 20:08:36.354 INFO main [write] -
02/06/07 20:08:37.215 INFO main [write] - Module 'org.etics.build-system.plugin-framework' is up-to-date with respect to CVS for tag
'etics-build-system-plugin-framework_B_1_1_0'... skipped
02/06/07 20:08:37.221 INFO main [write] -
```

### Build log

```
02/06/07 20:11:40.205 INFO main [write] - Configuration 'etics-build-system-plugin-framework_B_1_1_0' hasn't changed for target 'postpublish'... s
02/06/07 20:11:40.273 INFO main [write] -
```

### Test report

No log available


The Grid (The Grid Quality Process) / My Process

ETICS

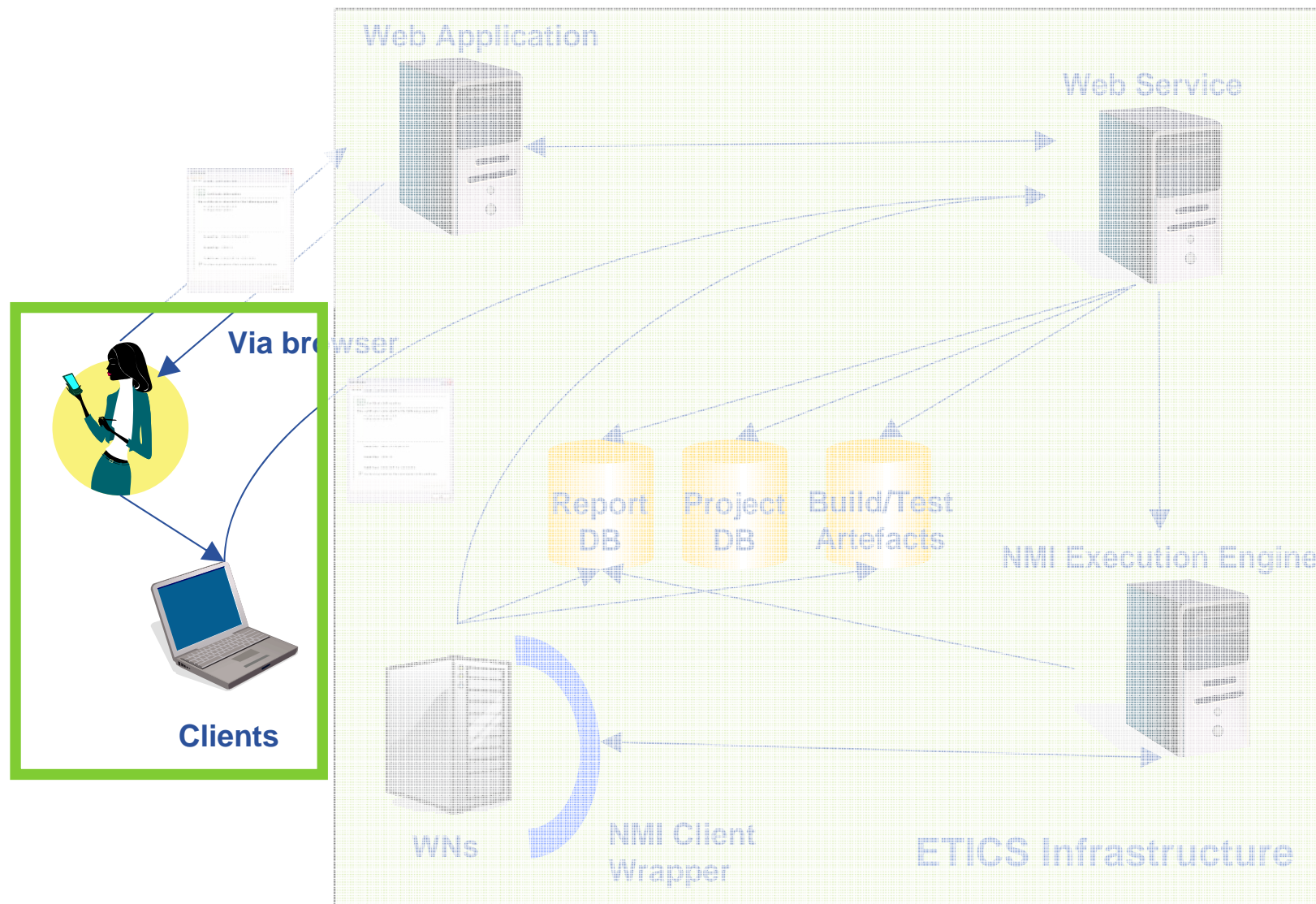
ETICS Repository

Repository

- Registered
  - OMII-Europe
  - dicom
  - egrid
    - etics-client-setup-080.py
    - etics-client-setup.py
    - etics-nmi-scripts.tar.gz
  - externals
  - gridway
  - mpi-start
  - myproject
  - org.diligentproject
  - org.etics
  - org.glite
    - bdi
      - 3.8.5
        - noarch
          - bdi-3.8.5-1.noarch.rpm**
          - bdi-3.8.5-1.tar.gz
    - glue-schema
    - lcg-ManageVOTag
    - lcg-info
    - lcg-infosites
    - lcg-mon-stdout
    - lcg-mon-tools
    - lcg-python-rqmaqueryiter
    - lcg-tags
    - logo.jpg

 **bdi-3.8.5-1.noarch.rpm**

Storage Type:	Registered
Project:	org.glite
Package:	bdi
Version:	3.8.5
Platform:	noarch
Type:	File .rpm
Last Modified:	Sat Feb 03 10:42:43 CET 2007
Size:	12 Kb
Link:	<a href="#">Download here</a>



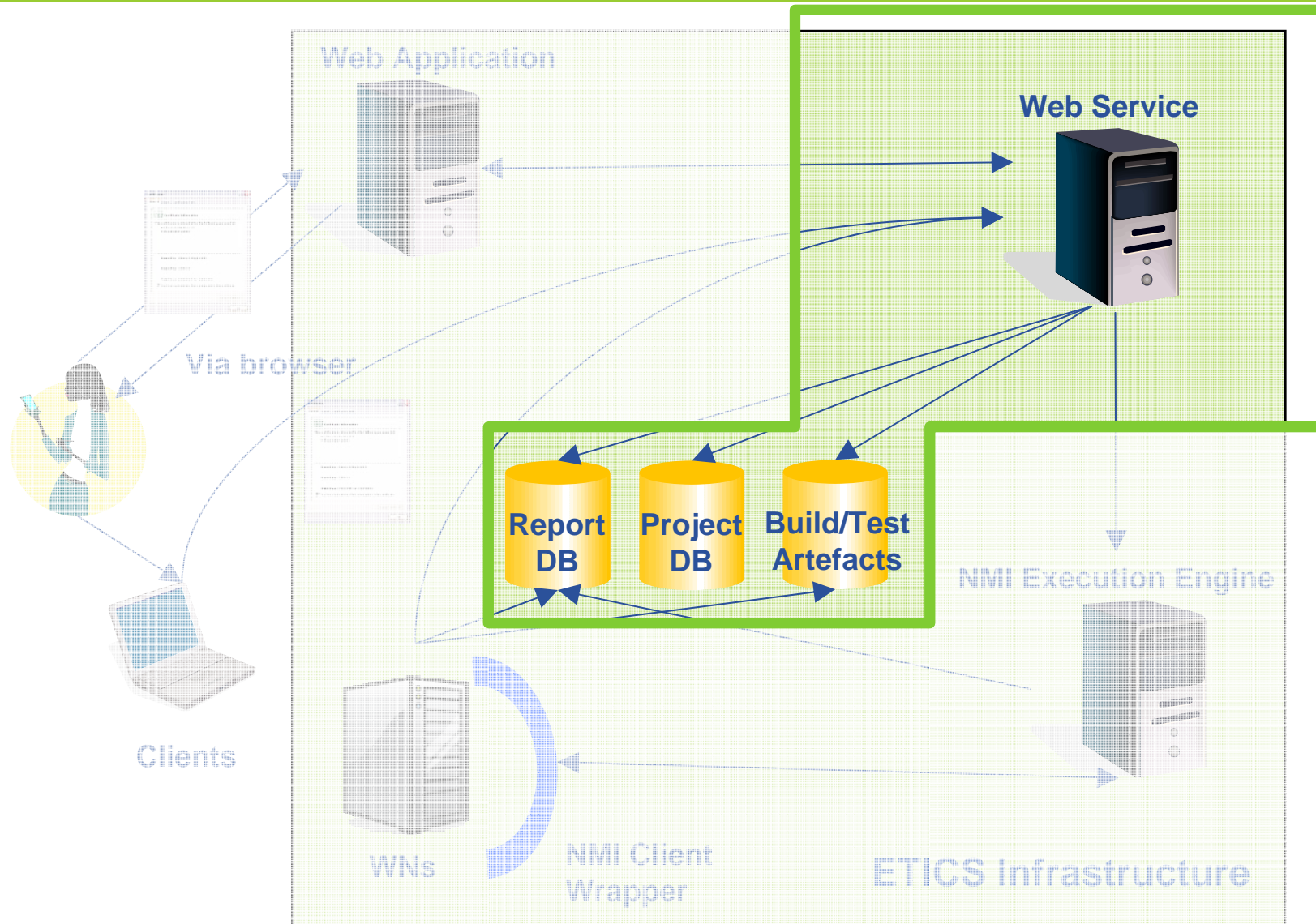
- **Build and Test Command-Line client (CLI) allows users to**
  - Build and test locally
  - Browse metadata
  - Edit metadata
  - Submit build and test jobs
- **State-full and secure application**
- **CLI written in Python, using ZSI (SOAP toolkit for Python) for web service interfacing and serialisation**
- **Interfaces with the ETICS Build and Test Web Service**



- **Download project metadata:**  
`etics-get-project <project-name>`  
**Action: download project metadata**
- **Checkout and/or download software:**  
`etics-checkout -c <config-name> <module-name>`  
**Action: download binary packages or fetch code from source**
- **Build the software:**  
`etics-build <module-name>`  
**Action : build and unit test code hieratically, and generate packages and reports**
- **Test the software:**  
`etics-test <module-name>`  
**Action : test code hieratically, and generate packages and reports**

- **Command-line Client can also be used for editing:**  
*etics-module [<options>] [<module-name>]*  
*etics-configuration [<options>] [<module-name>]*
- **Can be used for scripting or interactive sessions**
- **Provides the same features as the Web Application**

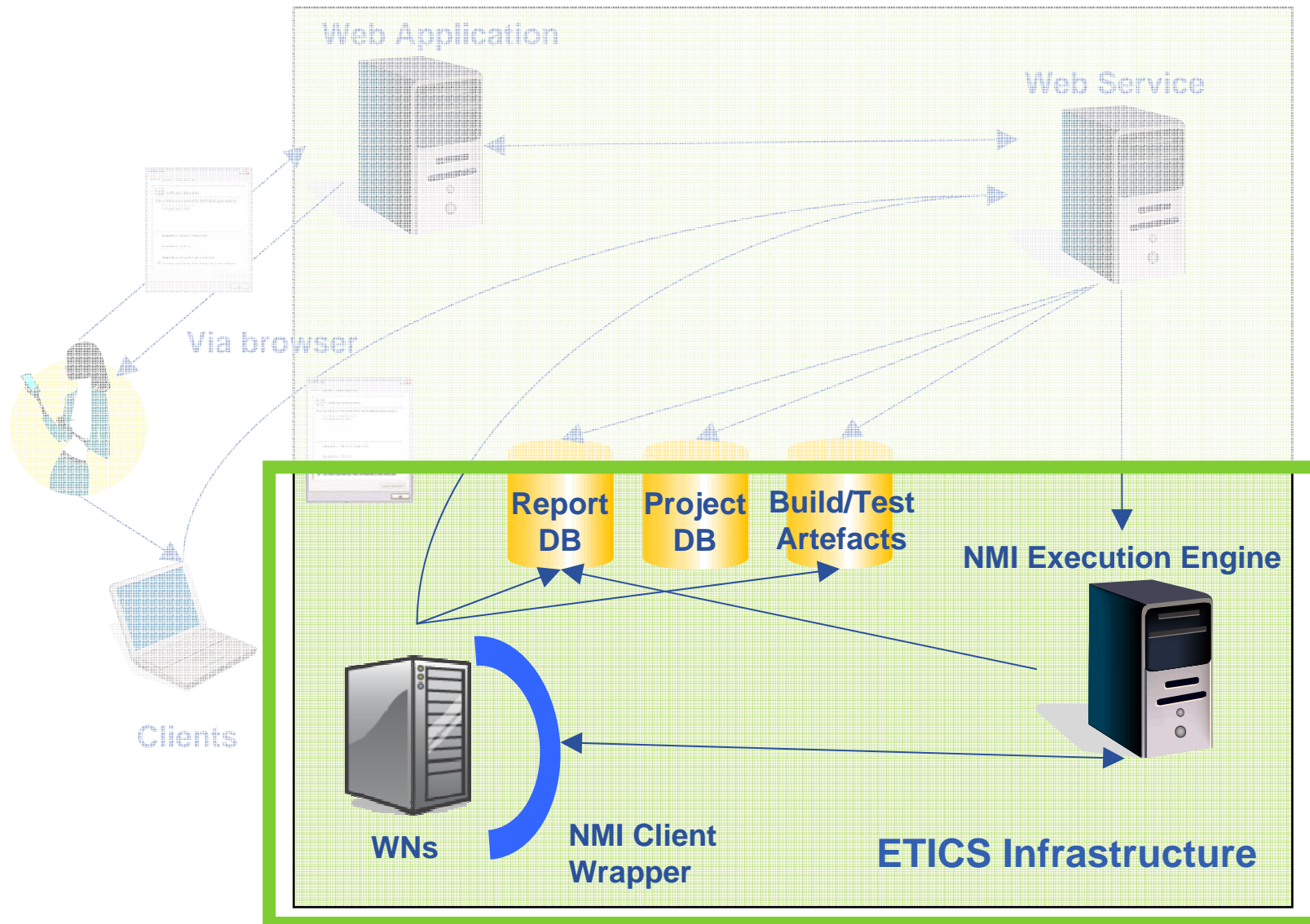
- **Framework allows users with investments in existing build, test, certification and/or software quality assurance tools to leverage this investment**
- **Users can select, from a rich set, which metrics are important to them and have ETICS automatically calculate them during build and test, and make them visible**
- **Promote inclusion of contributions from ETICS user community (more in this in dissemination presentation)**
- **Examples of already available plugins:**
  - **IPv6 compliance tests**
  - xUnit tests automatic execution
  - Multi-language line of code count (SLOC)



- **Build and Test Web Service (WS) provides single secure interface to the ETICS backend system**
  - Metadata (persisted in database)
  - Execution engine
  - Security (role based)
- **Heart of the ETICS System, it performs complex queries on behalf of the clients (WA and CLI)**

- All accesses to the WS must be performed using a secure port
- Authentication is performed using digital certificates (x.509)
- Authorisation is role-based
- Clients connecting without certificate are mapped to Guest, with minimal privileges (most likely just read access)
- Service can be setup to leverage certification authority management efforts such as EUGridPMA





- **ETICS builds on Grid technology (NMI/Condor) for its Execution Engine**
- **Users can submit automatically build and test jobs without having the burden of managing these resources**
- **In order to provide reproducible builds and tests, ETICS only artefacts built and tested using controlled resources, managed by the Execution Engine, can be registered**



- **Following features were added to NMI/Condor to better support the ETICS Execution Engine**
  - **Private resources:** projects can provide specific resources for exclusive usage
  - **Freeze:** lock the node for further analysis in case of failure
  - **Root-enabled environment:** build or test jobs can request to run as root
- **Users can register private machines if special setup and/or specific licenses are required**



## NMI Build & Test System

### Run Statistics

Total Runs:	5120	
In-Progress:	3	0.06%
Failed:	365	7.13%
Removed:	2660	51.95%
Succeeded:	2092	40.86%

### User Information

Total Users:	8
Most Active:	tomcat4

• Home > Runs Overview

Search Parameters			
Keyword:	<input type="text"/>	Project:	Show All
User:	Show All	Component:	Show All
Run Type:	Show All	Result:	Show All
Platform:	Show All	Submission Date:	MONTH / DAY / YEAR
Advanced Search		Build Run ID:	<input type="text"/>
Submission Host:		Only Pinned Runs:	<input type="checkbox"/>
Search		Reset View	

Run Results										
ID	Result	User	Type	Project	Comp	Start	Duration	Description	Platforms	Archive
5120	Complete	tomcat4	BUILD	org.diligentproject	script	Nov-29-2006 17:59	00:09:02	null	x86_slc_3	
5119	Complete	tomcat4	BUILD	org.etics	org.etics.portal.web-application	Nov-29-2006 17:54	00:08:19	Build System Web Application (ver. 0.6.0)	x86_slc_4	
5118	Failed(2)	tomcat4	BUILD	org.glite	org.glite.e2emonit.pinger	Nov-29-2006 17:40	00:06:47	glite-pinger is a network monitoring tool that makes RTT and ICMP packet loss measurement	x86_slc_3	
5117	Failed(2)	tomcat4	BUILD	org.glite	org.glite.e2emonit.pinger	Nov-29-2006 17:31	00:05:17	glite-pinger is a network monitoring tool that makes RTT and ICMP packet loss measurement	x86_slc_3	
5116	Complete	tomcat4	BUILD	org.glite	org.glite.e2emonit.pinger	Nov-29-2006 17:14	00:11:51	glite-pinger is a network monitoring tool that makes RTT and ICMP packet loss measurement	x86_slc_4	
5115	Running	tomcat4	BUILD	org.glite	org.glite	Nov-29-2006 17:07	In Progress	gLite Middleware v. 3.1.0	x86_slc_3	
5114	Running	tomcat4	BUILD	org.glite	org.glite	Nov-29-2006 17:07	In Progress	gLite Middleware v. 3.1.0	x86_slc_4	
5113	Running	tomcat4	BUILD	org.glite	org.glite	Nov-29-2006 17:06	In Progress	gLite Middleware v. 3.1.0	x86_64_slc_4	
5112	Complete	tomcat4	BUILD	org.glite	org.glite.e2emonit.pinger	Nov-29-2006 17:03	00:16:59	glite-pinger is a network monitoring tool that makes RTT and ICMP packet loss measurement	x86_slc_4	
5111	Failed(2)	tomcat4	BUILD	org.glite	org.glite.e2emonit.pinger	Nov-29-2006 16:54	00:11:16	glite-pinger is a network monitoring tool that makes RTT and ICMP packet loss measurement	x86_slc_4	
5110	Complete	tomcat4	BUILD	org.etics	org.etics.administration.web-application-test	Nov-29-2006 16:25	00:05:57	null	x86_slc_3	
5109	Failed(2)	tomcat4	BUILD	org.glite	org.glite.e2emonit.pinger	Nov-29-2006 16:15	00:11:42	glite-pinger is a network monitoring tool that makes RTT and ICMP packet loss measurement	x86_slc_4	
5108	Failed(2)	tomcat4	BUILD	org.glite	org.glite.wms	Nov-29-2006 15:38	01:32:54	org.glite.wms v. 3.1.6	x86_slc_3	
5107	Failed(2)	tomcat4	BUILD	org.glite	org.glite.wms	Nov-29-2006 15:38	02:17:50	org.glite.wms v. 3.1.6	x86_slc_4	
5106	Failed(2)	tomcat4	BUILD	org.glite	org.glite.wms	Nov-29-2006 15:37	01:34:16	org.glite.wms v. 3.1.6	x86_64_slc_4	
5105	Failed(2)	tomcat4	BUILD	org.glite	org.glite.lb	Nov-29-2006 15:37	00:28:50	org.glite.lb v. 1.3.3 RC31_2	x86_slc_3	
5104	Failed(2)	tomcat4	BUILD	org.glite	org.glite.lb	Nov-29-2006 15:37	01:16:03	org.glite.lb v. 1.3.3 RC31_2	x86_slc_4	
5103	Failed(2)	tomcat4	BUILD	org.glite	org.glite.lb	Nov-29-2006 15:36	00:20:48	org.glite.lb v. 1.3.3 RC31_2	x86_64_slc_4	
5102	Failed(2)	tomcat4	BUILD	org.glite	org.glite.data	Nov-29-2006 15:22	01:42:18	org.glite.data v. 3.1.7	x86_slc_3	
5101	Failed(4)	tomcat4	BUILD	org.glite	org.glite.data	Nov-29-2006 15:22	01:29:28	org.glite.data v. 3.1.7	x86_slc_4	

Page [ 1 2 3 4 5 6 7 8 9 10 ... ] of 256

Rows per page 20 Go

- **For users to trust the system, local and remote builds have to be executed in as close a condition as possible**
- **Once a build or test job has landed on a Condor worker-node (WN), the ETICS client takes over**
- **The exact same sequence is executed on the WN, as if on a local machine**
  - Setup
  - Get project
  - Checkout
  - Build/Test
- **The difference is that the WN is controlled and maintained by the ETICS experts and guaranteed to be a clean standard installation**
- **This approach guarantees that over time, builds and tests will be reproducible, a important tenet in software quality standards**

- **Users can exploit ETICS's functionality to explore complex scenarios – e.g. Interoperability between software stacks**
- **Several usecases have already being identified and analysed**
- **These scenarios will be further developed during the second year of the project**
- **We welcome requirements from the current and new user communities in terms of usage scenario:**
  - ▶ **IPv6 testbed requirements**

# Conclusions

- **ETICS defined, integrated and delivered a novel **Configuration, Build and Test Service****
- **The service leverage the rich knowledge foundation of the consortium partners in the domain of software quality assurance, development, integration and test of complex Grid and distributed software**
- **The ETICS service was used in an incremental fashion to build and test the ETICS service, providing rapid feedback to the team**
- **A first release is around the corner and work is going on to build on its foundation during the second project year**
- **We invite you to get involved in shaping its feature to help you in your work**



<http://www.eu-etics.org>