INFORMATION SOCIETY TECHNOLOGIES
(IST)
PROGRAMME

OpenMolGRID

SPECIFICATION OF THE GENERIC USER INTERFACE FOR DATABASE ACCESS

Contract Reference:	**IST-2001-37238**
Document identifier:	**OpenMolGRID-4-D4.1a-0102-1-3-ClientSideDBAccess**
Date:	**07/10/03**
Work package:	**WP4: Grid Integration**
Partner	**FZJ, UU**
Lead Partner	**FZJ**
Document status:	**APPROVED**
Classification:	**INTERNAL**
Deliverable identifier:	**D4.1a**

Abstract: This document describes the client interface to databases within OpenMolGRID

Delivery Slip

	Name	Partner	Date
From	Bernd Schuller	FZJ	16/09/2003
Verified by	Mathilde Romberg	FZJ	07/10/2003
Approved by	G.H.F.Diercksen (TC) R.Ferenczi (QE)	OMC CGX	05/11/2003 07/04/2004

Document Log

Issue	Date	Comment	Author
1-0	1/06/2003	Submitted to EC	M. Romberg, B. Schuller
1-1	16/09/2003	Submitted for internal review	M. Romberg, B. Schuller
1-2	07/10/2003	none	M. Romberg
1-3	07/04/2004		M. Romberg

Document Change Record

Issue	Item	Reason for Change
1-1	Improved document structure, added section on hardware and software requirements. More detailed description of GUI. Adopted new document template	External review Brussels, 17/06/2003
1-2	typing errors corrected	internal review process
1-3	project information	

Files

Files in this section relate to actual storage locations on the BSCW server located at https://hermes.chem.ut.ee/bscw/bscw.cgi. The URL below describes the location on BSCW from the root OpenMolGRID directory

Software Products	User files / URL
Word 2000/XP	OpenMolGRID/Workpackage 4/Deliverables/ OpenMolGRID-4-D4.1a-0102-1-3-ClientSideDBAccess

Project information

Project acronym:	OpenMolGRID
Project full title:	Open Computing GRID for Molecular Science and Engineering
Proposal/Contract no.:	IST-2001-37238
European Commission:	
Project Officer:	Annalisa BOGLIOLO
Address:	European Commission - DG Information Society F2 - Grids for Complex Problem Solving B-1049 Brussels Belgium
Office	BU31 4/79
Phone:	+32 2 295 8131
Fax:	+32 2 299 1749
E-mail	annalisa.bogliolo@cec.eu.int
Project Coordinator:	Mathilde ROMBERG
Address:	Forschungszentrum Jülich GmbH ZAM D-52425 Jülich Germany
Phone:	+49 2461 61 3703
Fax:	+49 2461 61 6656
E-mail	m.romberg@fz-juelich.de

Contents

1. Introduction

1.1. Purpose and Scope

This document deals with the specification of the client side graphical user interface to access databases containing information about chemical substances. The server side components necessary to realize database access are described and specified in [1].

1.2. Document Overview

Workpackage four (WP4) of the OpenMolGRID project deals with the Grid integration of components being developed in WP1 – WP3. From WP1 the data warehouse has to be integrated through an access module and a corresponding user interface based on the underlying Grid infrastructure UNICORE (Uniform Interface to Computer Resources, [2]).

Access to databases as a data source is currently not provided by UNICORE but it is essential for the data warehousing part of the project (WP1). As the data warehouse can be seen as 'just another database', WP4.1 will focus on a solution for database access in a more general and extensible way. In this task the open interfaces of the UNICORE infrastructure will be used to develop the necessary additions to the system. The plugin mechanism in the UNICORE client allows the addition of new application specific functions to the graphical user interface. The requirements specification from WP1 is taken here as the basis for the design of a seamless, platform and application independent interface for database access. The graphical user interface (GUI) will be developed and added to the existing UNICORE user interface as a plugin written in Java. The GUI requests all input from the user necessary to do the database query or upload as derived from the specifications in WP1. It includes consistency checks for the input so that it is ensured that all necessary information is provided to the server. The interface generates the Abstract Job Object (AJO) and sends it to the selected UNICORE server. On the server side of the Grid system the uniform queries and upload requests have to be translated into requests for the target database, executed, and the resulting output has to be made available in a format suitable for further processing.

1.3. Document Structure

Section 2 gives some background on the type of information that is contained in the databases under consideration. The user requirements for the graphical user interface (GUI) are given in Section 3. Finally, in Section 4 the GUI for database access within the UNICORE client is specified.

2. Background

Knowledge about physical and chemical properties and biological activity of existing molecules is needed when designing new structures with given properties. The knowledge can be obtained from different data sources, which usually focus on certain groups of properties. In the course of this project the toxicity of substances is in focus. The data from different sources will be collected, merged and normalized in a data warehouse to provide qualified access to the relevant data. The properties of the data under consideration are described in [7]. Short explanations of the terms toxicity and data warehousing are given in the remainder of this section. The Grid middleware UNICORE and the overall architecture of the OpenMolGRID system are described in [2].

2.1. Toxicity

All chemicals are toxic under some condition of exposure. Therefore, it is necessary to define these conditions as well as the quantity involved in the exposure in order to compare the toxicity characteristics of chemicals. Data from an acute study may serve as a basis for hazard categorization, labelling, or child-resistant packaging and may also serve to designate pesticides that may be applied only by certified applicators. It is also an initial step in establishing a dosage regimen in sub chronic and other studies and may provide information on absorption and the mode of toxic action of a substance. An evaluation of acute toxicity data should include the relationship, if any, between the exposure of animals to the test substance and the incidence and severity of all abnormalities, including behavioural and clinical abnormalities, the reversibility of observed abnormalities, gross lesions, body weight changes, effects on mortality, and any other toxic effects.

Acute oral toxicity reflects the adverse effects occurring within a short period of time after oral administration of either a single dose of a substance or multiple doses given within a 24–hour period. The test substance is administered orally by gavage in graduated doses to several groups of experimental animals, one dose being used per group. The doses chosen may be based on the results of a range finding test. Subsequently, observations of effects and deaths are made: levels of exposure (LC50) or dose (LD50) estimated to kill 50 percent of a specific population of animals under controlled conditions and dose-response (mortality) relationships are usually considered.

2.2. Data Warehousing

The main aim of data warehousing is to provide central and uniform access to a large number of data sources. This involves storing large volumes of information in a central repository. In this repository information from the various sources is stored in a consistent way. Where possible, it is translated into a common format. From a user's point of view, querying is made much simpler. Complex queries are less difficult to generate as all information is logically in one place and is normalised in the same way. The OpenMolGRID data warehouse (MOLDW) is specified in [3].

3. Requirements for the User Interface to Data Warehouse and Database Access

The graphical user interface for querying the data warehouse and other databases has to meet two main requirements: It has to be general enough to handle all possible use cases *and* the resulting complexity has to be manageable by the user.

3.1. Hardware and Software Requirements for Running the User Interface

- Software: The database access user interface is realised a s a plugin (written in Java) to the UNICORE client, which is also a Java application. The UNICORE Client to be installed by the user as a prerequisite to using the plugin should be of version 4.1.5 or higher. The user's desktop machine needs to have the Java runtime environment (virtual machine) installed, in version 1.4.0 or 1.4.1. Currently, the latest version at the time of writing (JRE 1.4.2) has changed its keystore handling and therefore does not support the Client mentioned above, but this situation is expected to change. Java provides cross-platform compatibility, so the client machine can run any operating system for which the appropriate Java virtual machine is available.
- Hardware: Since Java applications are memory intensive, it is recommended that the user's desktop machine should have at least 256 megabytes of physical memory installed. Processor speed is not considered critical, a Pentium III (at 1Ghz) is sufficient.

3.2. User Requirements for the User Interface Behaviour

The OpenMolGRID users want to access both data from the data warehouse (the one developed in this project, MOLDW) and data from databases related to the topic in the same seamless way. The data sources are for example ECOTOX (an online resource that provides toxicity data for aquatic and terrestrial life, [4],[5]), NTP (an online resource providing chemical health and safety information,[6]), excel sheets compiled locally at a site from different sources, or MOLDW. The interface should offer the selection of all accessible data sources and present a seamless way to access these diverse data sources.

The graphical user interface must be general and flexible enough to handle all the data warehouse and database query scenarios. Therefore, the graphical user interface should support the biggest possible option set. Depending on the content and structure of the data source being accessed, GUI elements should be disabled, and not be used to compose a query. In this way, users use the same GUI for all data sources considered here.

3.2.1. Search Options

The relevant data for the OpenMolGRID project is related to chemical properties and biological activities. The most important properties for this project are toxicity related. Therefore the user interface has to primarily support toxicity attributes. The requirements described here are extracted from the use cases given in [8]. Typically, a user queries the data sources for chemicals by CAS number, chemical name or chemical formula. These criteria must have top priority. Criteria to further restrict the result of a query are species, endpoints, modes of action, references, and descriptors (see [7]). Another way to search for information on a chemical is by its structure. The OpenMolGRID user wants to be able to draw an arbitrary molecule structure and search for it in available data sources to retrieve its properties.

3.2.2. Output Requirements

The output from the data access must be presented in viewable form for the users to be able to examine the result and to trim it for further processing steps. On the other hand the data must be available in a format for further automatic processing within the OpenMolGRID system. Users also wish to export the results from a query to their local machine, in formats such as tab-delimited file or comma-separated values.

3.2.3. General Interface Requirements

Expert users want to be provided with detailed information on the structure of a data source and be able to perform a native data source query. Expert users will also want to be able to directly compose an SQL query, since any graphical user interface imposes restrictions on the queries that are possible.

The user must be able to provide credentials (username and password) to the system, in case the underlying data source requests this. Security issues are covered in more detail in the deliverable describing the server side components [1].

3.3. Requirements on the System Architecture

The GUI must reflect the content of the data source being accessed. For example, the possibility to query by chemical structure will not be present for all data sources. The GUI therefore should disable the corresponding elements, in order to assist the user, and not confuse her with non-functional GUI elements. Therefore, a description of each data source to be accessed must be available to the user interface component, which allows the GUI to adapt itself to the data source under consideration. For this purpose, we use a feature provided by UNICORE, namely application metadata. On the UNICORE server side, for each database a software resource (an application) for the access to that database has to be defined and provided. The definition of the database access application includes a metadata file describing the structure of the database. For a more detailed description of the database access application, we refer to [1], here we provide the metadata file format as Appendix A.

4. Specification of the User Interface

4.1. Basic Principles

The basic Grid infrastructure for OpenMolGRID is UNICORE [2]. On the client side, a Java application (the UNICORE client) provides access to the system.

The UNICORE client provides the following functionality:

- local file input/output,
- resource information of the target system from the server
- data import to and data export from the machines executing the actual job,
- support for job preparation
- the generation of abstract job objects (AJOs),
- submission of AJOs to the target site,
- status and output retrieval, and
- support for output visualisation.

The user interface will be realised as a plugin to the UNICORE client (for a programmer's guide to the UNICORE plugin interface, see [9]). It extends the "Job preparation" part of the client, and also the "Job monitoring" part, providing additional result panels.

The UNICORE client provides a facility to load and store jobs that are being prepared. This includes storing the current state of the database access GUI.

4.2. Graphical User Interface

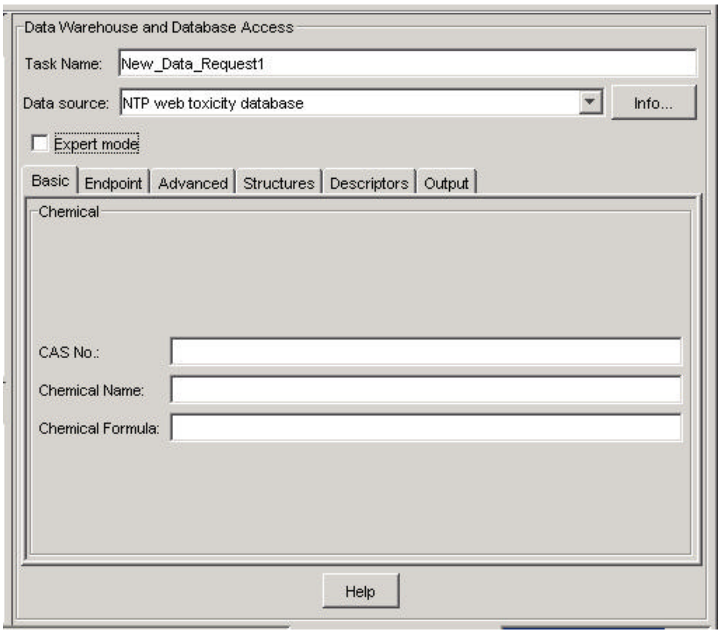
From the user requirements (see section 3.2) a layout for the GUI has been developed: Primarily the query parameters are grouped to reflect different levels of user experience and of qualified search. The following grouping of query parameters has been established:

Basic	CAS number, chemical names, chemical formula
Endpoint	Type of target species, endpoint (e.g. IC50), exposure time
Advanced	Mode of action, source database, reference
Descriptors	Generating program, type, some output options
Structure	Input/Generation of a structure for structure and substructure search
Output	Output options including the type of information to be included in the output and its format
SQL Query Editor	For expert users only; an SQL query for the specified data source can be entered directly
Info	For expert users only: Information on the database structure necessary to formulate a direct query, such as table names and field names.

The user does not have to specify all options, and the query can also be very simple. The less input the user provides the broader the query will be. A query is accepted if at least one of the possible options is specified. The user interface will not allow the user to request all data of the data source with one query, an appropriate error message will be sent to the user. The output of the query will be returned to the user's workstation in a result file.

Currently, it is not yet fully specified whether some of the fields will have predefined value lists, which fields are free text, and so on. This will depend on the structure and content of the OpenMolGRID data warehouse as the biggest and most important data source. This issue will be resolved in collaboration with Workpackage 1. So far, the following search options are based on fixed value lists: target species, endpoint and mode of action.

Depending on the selected data source, the different panels of the GUI are modified. This means that if a data source for example does not provide a query by CAS number this field is disabled (greyed out) on the "Basic" panel when the user selects that data source.



Task Name: New_Data_Request1

Data source: NTP web toxicity database

□ Expert mode

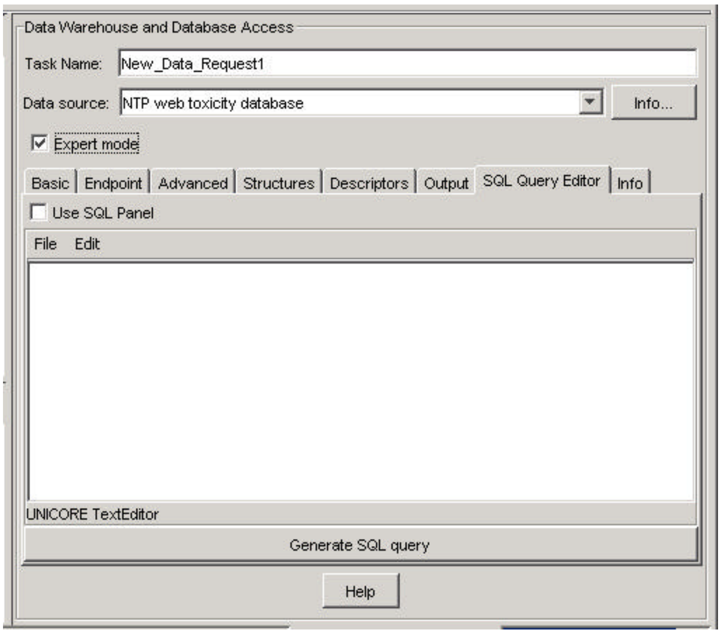
Figure 1 Sketch of basic query dialog

Figure 1shows the main database access GUI with the "Basic" selection options. In the upper part the task name and the database to be queried can be specified. The "Info" button provides the user with basic information on the data source while the "Expert mode" checkbox adds the additional panels to the GUI as shown in Figure 2.

4.3. "Expert Mode": Additional Functionality

By default the "Expert mode" is disabled but can be activated by clicking on the corresponding checkbox on the main panel. The additional panels "SQL Query Editor" and "Info" will automatically be added on expert mode activation.

The "SQL Query Editor" panel is intended for users wishing to enter a query directly in SQL. When the "Use SQL Panel" checkbox is activated, the content of the editor panel is used as query, i.e. sent to the server upon job submission. There is no SQL parser built-in, so the content is not checked by the system before submission. The expert user has to know exactly what she is doing, the advantage for her is that the interface allows for building more sophisticated queries than the standard GUI provides.



Data Warehouse and Database Access

Task Name: New_Data_Request1

Data source: NTP web toxicity database

☑ Expert mode

☐ Use SQL Panel

Figure 2 Sketch of "expert mode" SQL query editor

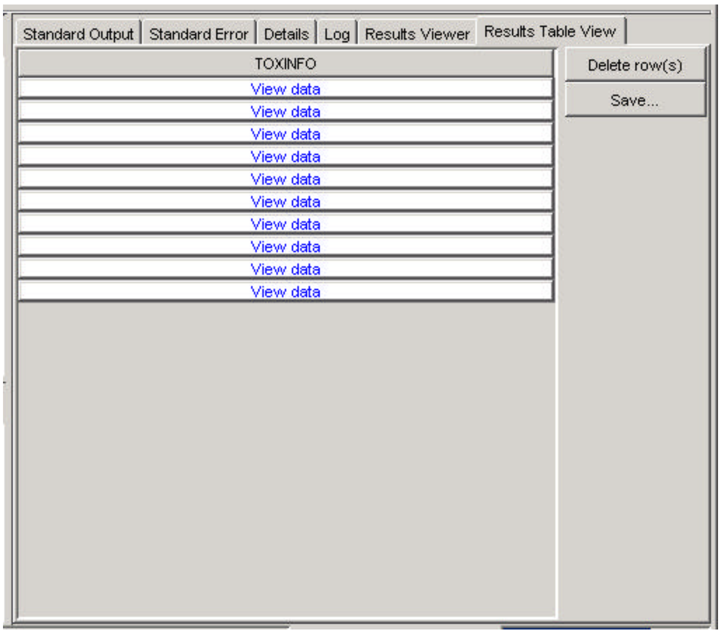
It can also be used to locally load/store a specific query in SQL form. The editor panel contains a button to generate SQL from the current GUI state, for example to read it before sending it, to further refine it, or to save it for later use.

Another tab, "Info", is visible in expert mode only and shows detailed information about the selected data source, such as table names, field names and types.

4.4. Output Visualisation

Visualisation of the output can be enabled on the "Output options" tab of the user interface. Visualisation is optional, since in some cases large result sets are generated, and it might be unsuitable to transfer them to the user workstation for visualisation.

The job monitoring part of the GUI provides two views of the output data: XML format and table form (see Figure 3). The table is not editable, but rows can be marked, deleted, and saved for further processing. Details of fields (for example molecular structure) can be viewed using external, user configurable viewers like editors or visualization tools. The XML format can be used as input to other programs or tasks. It is the output generated by the Database Access Tool (DBAT), the application for providing access to a database.



TOXINFO
View data
View data
View data
View data
View data
View data
View data
View data
View data
View data

Figure 3: Sketch of output table from query to NTP

5. References

[1] Deliverable D4.1c,
Specification of the database access interface

[2] Deliverable D4.5a,
Description of the OpenMolGRID Grid architecture, security architecture, and infrastructure and the deployment of the project's testbed

[3] Deliverable D1.1a,
Specification of software components for UNICORE-compliant data input/output format, complementary data warehousing methods, and protocols and a user interface defining process control parameters, process logic, and resource assignment

[4] Deliverable D1.1b,
ECOTOX – Terretox data specification

[5] Deliverable D1.1c,
ECOTOX – Aquire data specification

[6] Deliverable D1.1d,
NTP data specification

[7] Deliverable D1.3,
Properties and Priorities of the Data for Pharmaceutical and Phytopharmaceutical Compounds

[8] Deliverable D2.1,
Specification of software modules for descriptor calculation and model development and their Grid interface components; Annex I, Use Cases, chapter Data Warehouse

[9] UNICORE Plugin Programmer's Guide, Ralf Ratering, Pallas GmbH,
http://www.unicore.org/downloads.htm selection 'Plugins'

6. Terminology / Glossary

AJO	Abstract Job Object
CAS	Chemical
DBAT	Database Access Tool
FZJ	Forschungszentrum Jülich
GUI	Graphical User Interface
JRE	Java Runtime Environment
MOLDW	OpenMolGRID Data Warehouse
NJS	Network Job Supervisor
NTP	National Toxicity Program
SQL	Structured Query Language
UNICORE	Uniform Interface to Computer Resources
UU	University of Ulster
WP	Work Package
XML	Extensible Markup Language

Appendix A: Metadata File Format

Application metadata are part of the UNICORE server site components providing database access, and are sent to clients as part of the site resources.

The client side can use these metadata for example to adapt or customize itself to the given resource. In the case considered here, database access, the client needs the information contained in the metadata in order to build a query that is valid for the given data source.

Contents of the Metadata File

The metadata file contains the following information encoded in XML:

- Database name,
- Access restrictions,
- Information about the database intended for the user,
- Table names, and
- Field names and field types.

An XML document type definition (DTD) is defined (dbat_meta.dtd) and is given below. It defines the various XML tags used in the metadata file. A brief description of the tags used is as follows.

`<dbat_meta>` root element, specifies the XML document type

`<database>` contains all information that refers to this database; it has two attributes:

- "name": specifies the name of the database.
- "access": indicates whether the user needs a user name and password (or some other authentication) to access the database.

`<description>` sub-tag of `<database>;` description of the database intended for the user. At the present time, it is intended that this is a paragraph of html code, that can be included in a Java GUI element.

`<table>` sub-tag of `<database>`; has the name of the table as an attribute and contains sub-tags describing the table and the fields in it. Multiple tables can be present.

`<field>` sub-tag of `<table>` with the following attributes:

- "name": the name of the field
- "type": this is the high-level data type of the field, for example "2D structure", or "CAS number" or "Chemical formula". A draft for a complete list of these data types is available in Deliverable D1.3 ([7]) of the OpenMolGRID project. This type information is very important, since it is needed to process the results, or to visualize them.
- "description": this can be used in the GUI as well, to present a description of output data to the user.
- "query": This attribute is used to specify whether fields can be queried. For example, a database might include chemical structure information, but may not allow to use it in a search.

In this way, the client receives a list of tables, their descriptions and the fields contained in those tables.

Metadata Specification

The metadata file format specification (XML DTD) is defined in the following way:

```
<!--
  dbat_meta.dtd
  Specification of DBAT metadata format
  Version: 1.0, April 14, 2003
  OpenMolGRID, http://www.openmolgrid.org
-->
<!ELEMENT field EMPTY>
<!ATTLIST field name CDATA #REQUIRED
                description CDATA
                type CDATA #REQUIRED
         query CDATA #REQUIRED
  >
<!ELEMENT description (#PCDATA)>
<!ELEMENT table (description,field+)>
<!ATTLIST table name CDATA #REQUIRED>
<!ELEMENT database (description,table+)>
<!ATTLIST database name CDATA #REQUIRED
                   access CDATA #REQUIRED
  >
<!ELEMENT dbat_meta (database)>
```

Example

Let us imagine a simple Web database, where toxicity information is queried by specifying a CAS number or a chemical name. The result is plain text. The XML file giving the required information looks like this:

```
<?xml version=”1.0”?>
<dbat_meta>
  <database name=”TXT_WEB” access=”public”
    <description>Web toxicity database</description>
    <table name=”TOXICITY”>
      <description>Toxicity information</description>
      <field name=”OMG_CAS” query=”yes”/>
      <field name=”OMG_CHEMNAME” query=”yes”/>
  <field name=”INFOTEXT” type=”text/plain” query=”no”/>
    </table>
        ....
    </database>
</dbat_meta>
```

This information allows the client plugin to build a valid SQL query like

```
    SELECT * FROM TOXICITY WHERE OMG_CHEMNAME=”Benzene”
```

Furthermore, the plugin knows that only CAS number or chemical name can be used in queries and that there will be a plain text column in the result.