

UAKGQuery: Administration Guide

January 30, 2002

DocumentID: GradSoft-AD-19/09/2000-1.2E

Contents

1	Introduction	1
2	Compilation	2
2.1	Software needed:	2
2.1.1	for Windows NT	2
2.1.2	for Unix	2
2.1.3	Software needed for transaction support	2
2.2	Compilation and Installation	2
2.2.1	For Windows NT	2
2.2.2	For Unix	3
3	Using	4
3.1	Starting of server	4
3.1.1	Logging	5
3.1.2	Command String options	5
3.2	Starting of UAKGQueryService Client	6
4	Changes	6

1 Introduction

UAKGQuery is a CORBA service which provides CORBA API for access to relational database. This release supports Oracle as relational database.

Using UAKGQuery is possible to organize effective and uniform data access from different sources in heterogeneous computing environment.

2 Compilation

2.1 Software needed:

2.1.1 for Windows NT

- ORB - ORBacus-4.0 or higher
- C++ Compiler: Microsoft Visual C 6.0 or higher.
- Database server: Oracle 8.1.5 or higher (needed only for compilation of server part)
- Database server: InterBase 6 or higher (needed only for compilation of server part)
- GradC++ ToolBox 1.4.0 or higher.

2.1.2 for Unix

- ORB - one of
 - ORBacus-4.0 or higher,
 - TAO-1.1, TAO-1.2 or higher
- C++ compiler : gcc-2.95.2 or SunPro CC 4.2
- Database server: Oracle 8.1.5 or higher (needed only for compilation of server part)
- Database server: InterBase 6 or higher (needed only for compilation of server part)
- GradC++ ToolBox 1.4.0 or higher.

2.1.3 Software needed for transaction support

If you want to use transactional features of `UAKGQuery`, ORBacus Transaction service and OOC ¹ ORBacus ORB is needed.

Note: transactional features are absent subject to `UAKGQuery` has been compiled for Windows NT.

2.2 Compilation and Installation

2.2.1 For Windows NT

1. Unpack archive `UAKGQuery-1.1.0.zip`
2. Change current directory to `CosQuery.1`

¹now iona

3. Edit file `env_inc.nt.mak` and set next variables in it:
 - `PROJECT_ROOT` - `<full_path_to_CosQuery.1> \CosQuery.1`
 - `ORB_DIR` - directory, where ORBACUS is installed.
 - `OCI_HOME` - directory, where Oracle is installed (uncomment line `HAVE_ORACLE=1`).
 - `IBASE_HOME` - directory, where Interbase is installed (uncomment line `HAVE_INTERBASE=1`).
 - `MSVC_HOME` - directory, where MS Visual C++ is installed.
 - `INSTALL_DIR` - absolute path to directory, where you want to install UAKGQuery. (warning: C++ ToolBox has to be installed there)
4. Change directory to `CosQuery.1`
5. Start compilation:
 - for server: start `make.bat`
 - for client: start `make.bat client`
6. Start installation process:
 - for server: run `make.bat install`
 - for client: run `make.bat install-client`

2.2.2 For Unix

1. Unpack archive `UAKGQuery-1.1.0.tar.gz`
2. Change directory to `CosQuery.1`
3. Start `./configure`. Set configure options if needed.
 - `--with-ots` - if you want to use CORBA Transaction Service.
 - `--with-ob-xa` - if you want to use XA extensions of CORBA Transaction Service .
 - `--toolbox=<path-to-toolbox>` - if you installation directory of GradSoft C++ toolbox differs from installation directory of UAKGQuery (default is `/usr/local`)
 - Also you can set location of ORB and other packages: use command `./configure --help` to see list of all possible options.
4. Start compilation:
 - for server: run `gmake`
 - for client: run `gmake client-stub-library`

5. Start installation:

- for server: run `make install`
- for client: run `make install-client-stub-library`

After installation next files will be installed into `INSTALL_DIR` directory (which is defined as variable `INSTALL_DIR` in NT Makefile or as configuration parameter `prefix` for UNIX)

1. (a) idl-modules:
 - `CosQuery.idl`
 - `CosQueryCollection.idl`
 - `CosQueryIDLConfig.idl`
 - `CosQueryIDLConfigV2.idl`
 - `RC.idl`
 - `UAKGQuery.idl`
 - to subdir `<install-dir>/idl`
- (b) header files of CORBA stubs and skeletons, received during compilation of IDL files by IDL compiler ,
 - to **include** subdir
- (c) header files
 - `DecimalAccess.h`
 - `FieldDescriptionAccess.h`
 - `FieldValueAccess.h`
 - `RecordAccess.h`
 - `RecordDescriptionAccess.h`
 - to **include/CosQueryFacade** subdir
- (d) library `UAKGQueryClient.lib` (for Windows NT) or `libUAKGQueryClient.a` (for UNIX)
 - to **lib** subdir
- (e) binary `UAKGQueryService`
 - to **bin** subdir.

3 Using

3.1 Starting of server

You must set network port for requests listening in Server configuration parameters.

The way of doing this is ORB-dependent, it can be done from server command line or from ORB configuration file: consult with your ORB manual for details.

Below is list of such options for common object request brokers:

- For ORBacus use option `-OApport <port>` or property `ooc.iiop.port` in configuration file.

- For OmniORB use option `-ORBpoa_iiop_port <port>`
- For TAO-1.1 use option `-ORBport <port>`
- For TAO-1.2 use option `-ORBEndPoint iiop://:<port>`

Another precondition for succesfull service start is existance of installed Oracle8 client in working environment.

3.1.1 Logging

UAKGQuery use standart UNIX syslog interface for logging. So, consult with you operation system documentation for details, next example will work with most of supported UNIX systems:

- Add in file `/verb|/etc/syslog.conf|` next strings:


```
!UAKGQueryService
*. * /var/log/UAKGQuery.log
# in addition call administrator on fatals.
*.fatal |mail -s "UAKGQuery fatal" admin@his.cell.phone.kiev.ua
```
- Create empty file `/var/log/UAKGQuery.log`

In addition you can dup log output to file of to stderr by setting options of UAKGQueryService.

3.1.2 Command String options

- Command String ORB parameters, as written in you ORB manual.
- `--ior-file-UAKGQueryService <fname>` - write ior of UAKGQueryService to file with name **fname**.
- `--ior-stdout` - output QueryService IOR to standart program output
- `--with-naming` - register UAKGQueryService in CORBA Naming Service (note, that CosNaming initialization parameters must be aviable to ORB)
- `--config <fname>` - read configuration from file with name **fname**.
- `--ORACLE_XA <xa_string>` - use Oracle XA transaction Oracle. This option is aviable only for UNIX in the case of server was compiled with XA support.
- `--log-to-file <fname>` - dup log output to file **<fname>**
- `--log-to-strerr` - dup log output to UAKGQuery stderr.
- `--log-sql` - log processed SQL sentences.
- `--no-syslog` - does not use syslog for logging.

3.2 Starting of UAKGQueryService Client

When you start ordinary UQAKGQuery client, which receive reference on UAKGQuery via resolving of initial references, you can use next command line parameters for client ORB:

```
-ORBInitRef UAKGQueryService=\
            corbaloc::srv-host-name:port/UAKGQueryService
```

where:

- `srv-host-name` – IP address of machine, where UAKGQueryService runs.
- `port` – port, on which UAKGQueryService listens for requests.

4 Changes

1. 24.01.2002 -
 - (a) installing procedure description for Windows NT made to be adequate
 - (b) pointed, transaction service for Windows NT is absent
2. 21.01.2002 - version dependency check for new release
3. 05.07.2001 - reflected 1.0.2 software requirements, added section about logging.
4. 06.05.2001 - review.
5. 26.04.2001 - review, corrections of NT install procedure.
6. 12.07.2000 - first english edition.