Query Database V2.1 Users Guide

General comments about Query Database Usage

The query database (qdb) utility is an IDL procedure that enables programmers to retrieve and update data from a data source. The data source is usually a relational database and the programmer submits SQL queries and obtains data through a qdb invocation.

For example, in an IDL program the line

```
query_database, 'select * from MyTable', data, nrows
```

submits the query 'select * from MyTable' and returns the row count in 'nrows' and results as an IDL array of structures in the 'data' argument.

Programmers must have a working knowledge of both SQL and IDL in order to use Query Database.

New features specific to V2.1

Multi database connection capability (and the .qdbResources file)

A multi database connection capability has been added to query database. Previous versions of IDL utilized the ***.dbLogin*** resource file which contained the credentials for a single database login. Programmers could either use the default values in ***.dbLogin*** or could specify alternative credentials (user name, password, server, and database) through qdb parameters. The ***.dbLogin*** capability has been augmented in version 2.1 to support the specification of multiple sets of credentials used for making connections. The default name for this new file is ***.qdbResources***.

The details about how to correctly write a multi database connection resource file are available at Query Database single and multi-database resource file formats.

In order to use this feature, programmers should now specify a database resource ID as an argument to IDL query_database calls. This new argument is a keyword, 'dbResourceId'. The following is an example of typical usage:

```
query_database, 'select binary_30 from binary_types order by id', $
     data, nrows, dbResourceId='devTest'
```

The resource ID identifies a unique set of login credentials contained in the ***.qdbResources*** file.

Using an operating system property to specify the resource file

Version 2.1 will check for a property DB_LOGIN_FILE before looking for a resource file such as .dbLogin or .qdbResources in the current working directory or the user's home directory.

Linux example using C Shell

```
setenv DB_LOGIN_FILE myResource.txt
```

Linux example using Bash

```
export DB_LOGIN_FILE=myResource.txt
```