Repositories and Metamodels

1 Contents

1   Repositories and Metamodels

QualiWare keeps its data in repositories and the structure of that data in metamodels. Repositories and metamodels are related in the way that one repository uses one collection of metamodels, simply referred to as "the metamodel". The same metamodel can be, and usually is, used for several repositories.

Both repositories and metamodels are managed from the web module Repository Administrator (RA) which is a web application. The link to this application can be found under the Modules section of the QualiWare Execution Framework (QEF) console.

1.1   What is a Repository

A repository is a container for data objects. The repository handles revisioning, language variances, permissions etc. for the objects. It is stored in a database.

QualiWare can handle several repositories at a time, but a user always connects to one main repository at a time. The user can access data from other repositories in that connection, if these are linked to the primary repository as so called "remote repositories".

1.2   What is a Metamodel

A metamodel contains information on how to interpret and handle data in objects. All objects have a type named a "template". Definitions for templates are the central information in the metamodel. Usually the metamodel also has code, image data and other data to support the definitions.

For structural reasons, one metamodel only contains definitions for a set of templates. However metamodels refer to each other, so several metamodels are used as a larger set. The entire set of metamodels is referred to by an initialization metamodel, which then loads other referred metamodels. Which initialization metamodel to use is defined by the specific QualiWare product purchased and is subject to licensing terms.

Currently metamodels for QualiWare are used by both the QualiWare Integration Server (QIS) and QualiWare Lifecycle Manager (QLM). They are distributed with the QLM installer, and are located in the subfolder "Models" in the QLM installation.

1.2.1   Metamodels in QLM

When QLM is installed it has all the files it needs for the metamodels. QLM simply loads the information from the files.

1.2.2   Metamodels in QIS

QIS has two ways of using metamodels, which can be chosen independently for each repository:

- Development mode
- Production mode

1.2.2.1   Development Mode

QIS can load the metamodel directly from the files in the Models folder. This is known as "Development mode". The benefit of this mode is that development and customization of the metamodel can be done rapidly. Any change to a file is automatically detected by QIS, and the new information is automatically loaded. The drawback is the same. Any accidental or incorrect change to a file will become a problem for users immediately.
Note: Since QLM currently uses the information directly from the files, it is always in development mode.

1.2.2.2   Production Mode

Alternatively QIS can use a repository for storing the metamodel. This is known as "Production Mode". The benefit of this mode is that files can be changed and tested (in another repository in development mode) without affecting users. This mode also allows for revisioning of the metamodel in much the same way as objects in repositories. The drawback is more management, and the fact that QLM is currently not protected by production mode.

1.2.3   Recommendation

Since QLM is currently always in development mode, QualiWare recommends using the development mode for QIS as well. Having QIS repositories in production mode can result in QLM and QIS using different versions of the metamodel.
QualiWare plans to expand production mode to also work with QLM. Once this work is finished, modes can be changed with little effort and no loss of previous work.

1.3   Repository and Metamodel Basics in RA

Repositories are created and managed inside RA. A repository needs some basic setup to work:

- A database
- A pointer to a metamodel

If the repository is in development mode, the metamodel pointer is simply a path to the Models folder in the QLM installation, as well as the name of the initialization metamodel to be used for QIS and QLM.
If the repository is in production mode, the metamodel pointer is a selection of a created metamodel repository, as well as the name of the initialization metamodel to be used for QIS and QLM.

Metamodels are also created and managed inside RA. If all repositories are in development mode, no metamodel repositories need to be created. If one or more repositories are in production mode, the metamodel repository (-ies) needs to be created. Then the information from the files distributed with QLM needs to be loaded into the repository. This process is known as "Deployment". If changes are made in the files these can be redeployed once they are tested and accepted for production.

Please refer to the following sections for precise information on how to create repositories and metamodels, and how to migrate data.