

Eclipse Summit, Esslingen, 11./12.10.2006

Rich Client Applications Symposium

Position Paper of Mind8 GmbH

Situation Today

Mind8 (www.mind8.com) is using Eclipse since early 2002 as the development application as well as a Rich Client Platform. With our software product **Mind8.Studio** we realize reuse oriented, interdisciplinary processes for engineering production systems. Mind8.Studio is also sold as with the branding Eplan Engineering Center by the company Eplan Software & Service (www.eplan.de). Customers are companies like Audi AG, Grob, Gebr. Heller Maschinenfabrik, Nagel Maschinen- und Werkzeugfabrik etc.

Requirements

Subsequent we describe requirements of us and our customers which are not fully considered in the RCP today:

User Roles:

In the life cycle of a machine or plant different kinds of employees are involved: sales persons, designers, mechanical engineers, electrical engineers, software developers, assemblers, service technicians, operators etc.

According to the model-view-concept every role has to have a different view on an integrated model (the database, see below).

We need to have a configuration of Eclipse specific for any role defined. Some roles must be allowed to freely configure views, editors and perspectives as designed in eclipse today. But also we have to forbid other users to change anything in the configuration of the UI, installing updates with the update manager etc.

Customization:

With our product we enable our customers to apply object oriented concepts when designing and building machines without having to learn a programming language. Designers are defining component classes with parameters. Project engineers use these classes to build custom specific machines. Adjusting the parameters today is possible within a table (comparable to the property view of Eclipse), that is equal for any component. However, designers of components want to design specific UIs for setting the parameters of a component that project engineers can use, e.g. for adjusting control system parameters in a graphical presentation of the control loop.

Defining parameter masks, wizards etc. should be possible without having to program in Java/SWT/JFace. There should be a Framework to configure simple UIs like wizards, parameter masks etc. at runtime.

Support for Client/Server-Applications

Since we use Eclipse everything is based on the file-systems. People from OTI said that this was a design goal from IBM based on experiences with Visual Age. Also preferences are stored in the file-system (.metadata).

Our application is based on SQL-Databases. Due to this difference we can not use many components of the RCP (e.g. the Navigator-View) and have to rebuild it in a similar way for a database centric application.

Preferences are stored individually in every workspace. There are preferences that are globally set, preferences that are closely related to the model in the database and preferences that are specific for a role or a specific user in a role. This should be distinguished clearly in the framework. Preferences that are closely related to the model must be stored centralized in the database.

The Update-Manager today is not applicable for our Client/Server-Application. Certain plugins are directly related to the database schema. We can not allow, that one client updates an installation where as other clients work on an older version at the same time.

The update process for our Client/Server-Application must be centralized. The database must be closed, an update to new versions of Plug-ins. The database schema as well as the data must be updated. Then all clients have to work on the same new version.

.net-Integration:

To realize interdisciplinary processes we integrate existing CAD-Systems, programming systems for controls, office tools, etc. as a federated system (see www.foederal.org). Typical products that we have to integrate are Autodesk Inventor, Word, Eplan5, Simatic Manager etc.

These kind of engineering systems are 99% ms-windows-based. New APIs are increasingly based on .net; hence bridges to the .net world are needed.

Positions

Based on the importance for the Mind8 GmbH and weighting the chances that a common, general solution for a platform can be found we have ordered our points with priorities:

.net-Integration (Priority 1)

- Until now we haven't seen any work in that direction in the Eclipse Community. We think, that a bridge to the .net world will be the most important future feature to help Eclipse being broadly used as an RCP
- What is the best way: Working together with a commercial provider (e.g. j-integra.intrinsyc.com or www.jnbridge.com) or setting up a project to develop an open source solution?

Model based UI-Configuration (Priority 1)

- Using or developing a Framework to configure simple UIs (Should different technologies like Ajax be used?)

Generalize the RCP for Client/Server-Applications (Priority 2)

- IStorage should consequently be used for accessing persistent data. Today there are a lot of inherited burdens, e.g. that associations of editors work on file-names only
- The user of the RCP should be able to decide, if e.g. preference are stored in the .metadata-directory or in an sql-database
- Preference-Model for Client/Server-Applications

Update-Application (Priority 3)

- The Update-Manager is enhanced to be separate Update-Application (Framework) for Client/Server-Applications.

User Management (Priority 3)

- Generic user management in Eclipse
- User Groups and Users can be defined (Rights, IDs, password etc.)
- User Rights can be defined (Read/Write-Access to Data, Views, Configurations etc.)

Contact Information:

Dr.-Ing. Marco Litto
Mind8 GmbH
Gänsheidestr. 59
70184 Stuttgart
Tel.: 0711-210707-0
marco.litto@mind8.com