

Application of XML based markup languages for implementing Eclipse Rich Client applications (RCML)

Aleksey Aristov
Weigle Wilczek GmbH
aristov@weiglewilczek.com

The Eclipse Rich Client Platform (RCP) is getting more and more popular. Many organizations are selecting it as a standard platform for their internal application client development. Despite RCP's well-crafted architecture that requires a thorough understanding of design techniques to use it in a full and professional way, a lot of organizations lack a pool of adequate skilled developers that are capable of using RCP to its full potential. This situation prevents some organizations from using this technology, lets them switch back to their old approaches after a short test or poses serious obstacles for creating production quality applications. Also, it might generate a certain frustration and disappointment within the developer group and prevents organizations from a quicker adoption of RCP.

In fact, RCP provides a very strong basis for modern high quality applications, but in some cases the used programming model seems to be not very transparent and intuitive for average-skilled application developers.

For example, it is very hard to explain why Action classes supposed to be used in different parts of the application must implement different interfaces (IAction, IActionDelegate, etc.).

From application developer's perspective some of these technical differences are not extremely important and not very intuitive.

Partly, this problem can be addressed by providing training courses and consulting/coaching services. For this purpose we established the "RCP Training Alliance" (www.rcp-training-com), an initiative that provides professional consulting, trainings and coaching in the area of enterprise RCP-development. Although this efforts steepen the

learning curve significantly and help developers to become quickly productive, some developers stay afraid from the full complexity of RCP application development.

Moreover, in many cases there are developers with a non-Java background. Some of them have a very strong understanding of specific business problems and are critical for the successful completion of IT-projects. With organization's strategic shift to RCP they cannot be replaced easily and it is almost impossible to provide them with sufficient training as they are Java beginners.

We believe that there is a strong need for a technology that will allow them to concentrate more on solving their domain-specific problems instead of allocation their resources on technical internals of the RCP platform.

One of the typical solutions for reducing complexity of the software is to implement a framework that is able to encapsulate certain details of the underlying system and expose more high-level interfaces. In case of RCP such approach does not work too well, because RCP itself is a relatively high-level framework that cannot be easily wrapped without introducing extra complexity.

We would like to propose the introduction of several XML based markup languages that can be used to describe certain aspects of RCP applications. Such aspects might be

- Actions
- Menus/Toolbars
- Perspectives
- Views
- Editors
- Background Jobs

In addition, it should be possible to embed a scripting language (JavaScript) that can be used to provide some programming abilities for the developer.

We strongly believe that such approach will be easily understood and accepted by many software developers. To certain degree it can be compared to Open Laszlo, Macromedia Flex's MXM or to a HTML/JavaScript combination that is used to develop rich internet applications or dynamic web pages, respectively.

Keeping the possibility of combining both the "traditional" and the "markup based" approach within the same project is recognized as a highly critical feature for real life projects. For example, some simple aspects of the application should be implemented as a scripted markup, additional and more advanced parts can be done in a traditional way. Objects and components used in one or the other approach can be accessed and used transparently within the other approach, respectively.

To make developer's life easier and more comfortable it is very important to provide good and convenient development environments with appropriate tooling for such a markup language. It should include tools Eclipse developers got used to like context assistance, syntax highlighting, script debugger, etc.

As an initial research effort we implemented a proof of concept prototype with specific XML tags, very close to RCP core concepts. It has been successfully used to implement a demo application (see www.rcml.net). Moreover, we are working very closely with some of our customers to evaluate and develop the concept in real life projects. Here we work on defining components for a specific domain / problem area and try to find appropriate levels of abstraction.