

[A \(Mostly\) Agile Java Day at Jazoon](#)

Articles

Posted by:

Posted on : 2009/6/26 1:09:17

[Methods & Tools](#) is the sponsor of a large number of [software development conferences](#), but I cannot find the time and budget to visit them. This year I managed to find some time after the publication of my summer issue to visit Wednesday the [Jazoon](#), an important Java event located in Zurich. Besides the global morning keynote, the conference has five parallel tracks, so the first thing in the morning is to make your choice between 30 presentations. This is easier for me, because I am not a Java specialist and so I focused on talks with content that could reach outside the Java community.

After James Gosling the previous day, the Wednesday keynote was given by [Danny Coward](#), Chief Architect of Sun's Client Software, which means Java SE, Java ME and JavaFX. The topic of his talk was the top five points of both JDK 7 and JavaFX 1.2. For the upcoming JDK 7, the main improvement will be on modularity; multiple languages (Groovy, JRuby) performance support on the virtual machine; some language addition like multiple exception handling; more I/O API; a new garbage collector. For the new version 1.2 of JavaFX released in June, the top five points are: release on new platforms like Linux and some mobile phones; more UI components; layouts; improvement of performance; better ways to use data with RSS/atom feed support and a simple asynchronous framework. He announced that another release of JavaFX is scheduled for the end of the year. The keynote was completed with some nice demonstrations of the JavaFX capabilities.

Refactoring of Large Software Systems My first choice in the technical session was [Refactoring of Large Software Systems](#) by Sibylle Peter and Sven Ehrke of [Canoo Engineering](#). The talk presented the refactoring of the architecture for an investment banking system. After four years, the system architecture had become decayed, the original developers were no longer around and there were only few automated functional tests available. Important evolutions were postponed, as modifications often created bad side effects. However the application was very important for the bank and they wanted to keep using it for some more years. The first activity was to analyze the 1800 Java classes and their dependencies. The strategy was then to restructure the class content to separate clearly the services and the presentation functions using a service oriented architecture. The project, that has already consumed 10 men/years and is still on going, is run on an agile mode. Pair programming was an essential tool to ensure that refactoring practices were consistent across the team. The main customer engineer served as a [product owner](#). This was fundamental for the Canoo team, as they needed somebody able to take decisions and knowing also the impact of the existing application on other systems. Another important success factor was doing the refactoring without modifying the functionalities. This allowed running tests comparing the results of the refactored application with the original system. Trust is an essential component of the project, as the users have to let an external team restructure their critical system without breaking it. However as refactoring progressed, a team of the bank was also able to start evolving the system again. When release dates were different between the two team, a [branch and merge](#) configuration management solution was adopted. Some other lessons learned during the first phase were to have a large investment in automated testing and continuous integration, have an increased communication between the external refactoring team

and the bank developers to synchronize architecture vision and transfer knowledge. It is also important not to underestimate the time needed to learn the system functionalities (where do you functionally test your technical refactoring) and test the final results. After the presentation, I talked with Sibylle and Sven to transform their presentation in an article for Methods & Tools. They agree on the principle, so stay tuned to get a comprehensive written presentation of their experience.

The Power of Value - Domain Driven Design and Value Objects The next presentation was "The Power of Value - Domain Driven Design and Value Objects" by [Dan Bergh Johnson](#) of [Omegapoint](#). Although I was initially a little bit afraid that this presentation contains too much code, Dan truly achieved its goal: showing how to refactor code to make it more understandable, even for domain experts without programming knowledge... and doing it in a lively and funny way, which is important when you get up early to assist to a conference and the caffeine effect starts to diminish. His definition of a value object is "an object with data and behavior that has conceptual value". His conclusions are that value objects help to decrease complexity (and compound value objects are even better), that entities relieve complexity and value objects add extensibility, testability and evolution. In the second part of the talk, he took the example of a part of code dedicated to charging a credit card and completely refactored it according to the principles presented previously.

JavaServer Faces The last presentation I saw was "A complete Tour of JSF 2.0" by [Ed Burns](#) of Sun and [Martin Marinschek](#) of [IRIAN Solutions](#). They presented of the new release of JavaServer Faces that had its specifications approved at the end of May. Ed Burns recognized the criticism of the first version of JSF and announced that they have adopted a lot of innovations created in the community to integrate them in version 2.0. The new release is not a revolution, but a strong evolution. Among the improvement, he cited composite components; AJAX support, either declarative or programmatic; partial state saving that will improve performance; more system events; better resources management: full "library" concept, support for versioning and internationalization; the replacement of JSP by Facelets; the possibility to have bookmarkable pages. For the presenters, JSF is a real open source project (that is not a "just Sun" project) with a lot of different contributors. The new version is more much easier to use and well worth a second look. To complete this presentation, I talked with Micha Kiener of [mimmacom](#) which distributes IceFaces in Europe. Micha is also a committer to JavaServer Faces and his point of view is interesting, even if he suggested that it could be biased due to his position ;o) He also thinks that the community around JSF is strong and that the technology would continue to evolve even in the case where Oracle, that just bought Sun, decided to withdraw its financial support. He doesn't see JavaFX as a direct competitor to JSF. His view on the market is that some products (JavaFX, Flex, Silverlight) target more the mashup, media, gaming or mobile market. On the other hand, JSF, GWT and their Ajax frameworks competitors are more used to develop applications with a "traditional" user interface, like on-line banking for instance. That is how JSF is used by some major Swiss corporations. He made also interesting remarks on the open source market and the commercial companies that try to monetize these technologies. He sees some products, like user interface or programming frameworks, more as "contributors" to the software development infrastructure. You can make some money in training and support as companies adopt them, but this stops when the company has its own competencies. Other technologies, like databases or web servers, are applications that run the everyday software infrastructure. In this case it is easier to sign long term support contracts. I also had the chance met some of the Jazoon organizers, Christian Frei and Andreas Knobel, of [Keynode](#). They were very kind with me and happy with the fact that the conference audience was higher than last year. There registered this year 1000 participants, around 50% of them coming from outside Switzerland. Videos of some of the conference talks should be available in July on [Parleys](#) the excellent conferences video web site. You can also stay tuned on our [Java videos and tutorial directory](#) where they will be listed.