Devoxx University puts JavaFX in practice

With the official launch of JavaFX 1.0 just four days before the start of Devoxx 2008, it came as no big surprise to see ‘JavaFX in practice’ presented in front of a capacity crowd at the first day of Devoxx University.

Richard Bair kicked off the presentation with a close look at the JavaFX script language. As a member of the Swing team at Sun, Bair is the lead of the component toolkit API for JavaFX. “For an experienced Java developer, JavaFX may look a little weird at first”, he said. “But once you start working with the syntax, you will see it starts to feel very natural after only a short while.” Bair elaborated on what he called ‘the only two features that really drive JavaFX’: binds and object literals. “If you are used to working with Java script, object literals in the JavaFX script language look very similar.”

Cool graphics

Next up was engineer Jasper Potts from Sun’s Swing Team to demonstrate the capabilities of JavaFX on the graphical side of things. Potts explained how a scene graph is built up step by step, by creating a scene, a stage, nodes, and more. “JavaFX is really helpful”, Jasper Potts said. “You can actually start working with only one node, and build a scene graph from there. The nodes are the basis of every item in the scene.” Potts explained the use of node variables and node functions. In several demos, he demonstrated how easy it is to come up with graphically attractive stuff, within a matter of only a few lines of code. The demo showing the use of images clearly underlined the importance of binds and object literals. “These things are all over the place”, added Richard Bair. “They really are what is important in JavaFX.”

Jasper Potts continued his talk with a chapter on text box. “This is the first of the UI controls we want to develop in the future”, he said. “If you are a Swing developer, a lot of the APIs will look very familiar.” There’s a simple reason behind this. “We are using all the good stuff from all communities in these new UI controls.” But at the same time, Sun is making the controls easier to use, clearing out a lot of the difficulties that were associated with these UI controls in the past. Bair and Potts demonstrated how easy it actually is to wrap a Swing component and embed it in a scene graph. “You can do that the other way around as well. It only takes about six lines of code to drop a piece of JavaFX into a Swing project.”

Focus on NetBeans

JavaFX tools architect Martin Brehovsky presented an overview of the tools that support JavaFX 1.0. “Right now we are focusing on the plug-ins for NetBeans 6.5”, he said. “They allow mixing Java and JavaFX classes in one single JavaFX project, which obviously makes them very powerful instruments for the developer.” The plug-ins for Eclipse are still in ‘a very experimental mode’. However, they will be open sourced soon, providing basic support for the development of JavaFX pieces in Eclipse.

Very important for the productive collaboration between designer and developer is the availability of the JavaFX Production Suite. This tool allows exporting files from Adobe Photoshop and Illustrator to the JavaFX platform. “It often happens that developers are not really capable of creating cool graphics”, explained Brehovsky. “With the Production Suite, the designer can actually save his work for JavaFX and pass it on to the JavaFX developer.” When the designer still wants to change the design – later in the process – the developer doesn’t have to look for the piece of code that is involved. All the designer has to do is make the changes in Photoshop or Illustrator and use the save for JavaFX functionality.

Much more to come

Richard Bair concluded the talk on ‘JavaFX in practice’ stating that the JavaFX 1.0 software development kit provides the APIs that are necessary for building creative applications. “What about enterprise applications”, a member of the audience asked. “That’s of course where we want to go with this”, Bair answered. “After all, we’re Sun.” As it looks now, future updates to JavaFX 1.0 will arrive regularly and quickly, with a scheduled release of version 1.1 in February and version 1.5 in June 2009. As for Linux support for JavaFX, there is no official timeline yet. But Bair did confirm that Sun has been working on a Linux version of JavaFX for at least six months already.
How close are you to IT?

As a steady employer, Smals offers you interesting ICT-projects of social significance. Use your expertise in Java to build a social security internetportal for a wide audience, allowing over 400 million visits a year. Construct applications to quickly register all recruitment nationwide, even when initiated from a mobile. Or work for eHealth, the secured information exchange for healthcare services.

Engage in the future and defy society's challenges in Java.

Do you want to build tomorrow?

Java Developer
Java Architect
Functional Analyst
Project Leader

Discover all our vacancies at www.smals.be.
More information at 02 787 53 12.
Recruitment - Rue du Prince Royale 102 - 1050 Brussels

Think big, act Smals.
www.smals.be
**Hot Talk!**

Every day, a steering comity member of Devoxx advises us on what you should not miss at Devoxx. **Jo Wyns** has some advice for you on the second day of the University.

Ok, the train is on the track, everybody found his/her way around the venue, the newbies are discovering, the year-ly addicts are relieved the tradition remains unbroken... Welcome at day 2 of Java... Devoxx! I keep on misspelling this. It will take me until Friday to brainwash my right brain hemisphere. But what the heck, I’m here, you’re here, and it’s a five day java community party. Let me run you through my selection of today!

I’m especially interested in tools that help to streamline collaboration. So Jazz is certainly on my list. I expect the tool set to have matured over previous presentations, and that they will not solely present the tool, but mainly the concept and collaboration ideas behind it. For the afternoon, I’ll probably compensate with the GWT talk. It attracts my attention since its first public appearance. Reason for this is my old love for componentized web applications, battles I won, wars I lost, in selecting proper frameworks (sic). If you’re more into web development, front ends, or app servers, I’m sure you can find your way around. Also consider the ‘java performance’ talk, it can really bring new insights into your code. Over noon, I’ll take a quick lunch, and visit one of the quickie talks. Every year I find some nice surprises among them. Real developers, telling about their solution to real problems! Of the tools in action sessions in the early evening, my attention goes to Shard-ing with Hibernate. Horizontal database partitioning sounds like I’m going to learn something here.

Every year, during this conference I apply two golden rules: a) find a sufficient dose of completely innovative, challenging presentations, and b) enjoy every minute up to the very last. That’s why I will probably stay for the BOF sessions until closing time. During the day, don’t forget to pay a visit to the white boards upstairs. I’m a loyal fan of this non structured, low entry multimedia forum. I invite you to open a poll, write down comments, make suggestions to the java-stars, recommend enhancements to the conference, ... The content will be posted on our site. Make sure you contribute your opinions and make sure you make this the community conference!

---

**TIME**

<table>
<thead>
<tr>
<th>TIME</th>
<th>ROOM 8</th>
<th>ROOM 5</th>
<th>ROOM 4</th>
<th>ROOM 9</th>
<th>ROOM 6</th>
<th>ROOM 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>08.00 - 09.30</td>
<td>Registration and breakfast: Java, Enterprise, Partner Talk</td>
<td>Web, New Languages, Partner Talk</td>
<td>Tools in Action: Flex Today and Tomorrow (Matt Chatin, James Ward and Chet Haase), Comet: Never More! (Jonas Jacobi and John R. Fallows), Jython in Action (Jim Baker and Tobias Ivarsson)</td>
<td></td>
<td>Jazz in Action: Jazz in Action (Jan Van De Poel, Jerome Grandbastien and Jack Verstappen)</td>
<td></td>
</tr>
<tr>
<td>09.30 - 12.30</td>
<td>Java Performance (Kirk Pepperdine and Holly Cummins), Seam in Action (Dan Allen)</td>
<td>Seam in Action (Dan Allen)</td>
<td>GWT in Practice (Didier Girard, Pierre Mage and Dmitry Buzdin), ActiveMQ and ServiceMix (Bruce Snyder)</td>
<td>Comet: Never More! (Jonas Jacobi and John R. Fallows), Jython in Action (Jim Baker and Tobias Ivarsson)</td>
<td>Jazz in Action: Jazz in Action (Jan Van De Poel, Jerome Grandbastien and Jack Verstappen)</td>
<td></td>
</tr>
<tr>
<td>12.30 - 13.30</td>
<td>RIA, Enterprise</td>
<td>RIA, Enterprise</td>
<td>Web</td>
<td>SOA/BPM, Partner Talk</td>
<td>Java</td>
<td>Advanced OSGi (Peter Kriens)</td>
</tr>
<tr>
<td>13.30 - 16.30</td>
<td>Filthy Rich Clients: beyond Java (Romain Guy and Chet Haase), Pro Spring 2.5 (Jan Machacek)</td>
<td>Pro Spring 2.5 (Jan Machacek)</td>
<td>GWT in Practice (Didier Girard, Pierre Mage and Dmitry Buzdin), ActiveMQ and ServiceMix (Bruce Snyder)</td>
<td>GlassFish in action - Forget the app server for a minute! (Linda Schneider, Jean-Francois Arcand, Paul Sandou, Alexis Moussine-Pouckine and Bogdan Stefanescu)</td>
<td>Advanced OSGi (Peter Kriens)</td>
<td>Mobile Web Applications using the Apache Cocoon Framework (Davy Van Deursen)</td>
</tr>
<tr>
<td>16.30 - 16.45</td>
<td>Retreat, tools in Action</td>
<td>Retreat, tools in Action</td>
<td>Retreat, tools in Action</td>
<td>Retreat, tools in Action</td>
<td>Retreat, tools in Action</td>
<td>Retreat, tools in Action</td>
</tr>
<tr>
<td>16.45 - 17.15</td>
<td>Sharding in Hibernate (Andrew Glover), Integration Testing a Mule Application (Antoine Borg)</td>
<td>JSF, Ajax, and Seam portlet development with the JBoss Portlet Bridge (Thomas Heute), Systems management with project RHQ (Heiko Rupp)</td>
<td>Creating performance test data with the benerator (Volker Bergmann)</td>
<td>Mobile Web Applications using the Apache Cocoon Framework (Davy Van Deursen)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17.25 - 17.55</td>
<td>RESTing easy with Grails (Andrew Glover), Dependency Management with Apache Ivy (Xavier Hainin)</td>
<td>Swing Explorer - coo tool for Swing application developers (Maxim Zakharenkov)</td>
<td>Profiler, the better debugger? (Heiko Rupp)</td>
<td>Java-monitor in action (Kees Jan Koster)</td>
<td>Developing composites, multi-language portlets with NetBeans and Eclipse (Satya Ranjan)</td>
<td></td>
</tr>
</tbody>
</table>
JBoss AS 5 ready for prime time

After three years in R&D, JBoss Application Server 5 is ready for prime time. With completely re-archi-
tected components, JBoss AS 5 represents the most comprehensive redesign of JBoss Application Serv-
er to date. Here’s an interview with Sacha Labourey, CTO of Red Hat Middleware, to discuss the rationale
and goal behind JBoss AS5. We reprint this article with kind permission of DZone.

Re-architecting an app server from the ground up
is certainly no small task, especially for an app
server as complex as JBoss. In this case, the pro-
cess took almost three years. The question that
a lot of people have been asking is: were there
more than just technical issues at play here?

Sacha: Intuitively people usually think that it is an issue
with the J2EE spec and how much time it took us to im-
plement this pretty significant update of the J2EE spec.
But actually, it is not the case. If you look at the EJB3
container for example, I think JBoss has been the first
company on the market to provide an EJB3 implemen-
tation of that specification. So, actually, the EE services
have been ready for quite some time. In fact, where we
spent our time was mostly on the base infrastructure for
the application server. That is where we spent actually
most, if not all of our engineering resources, building a
new Microcontainer and also moving some of these core
services to the next generation of their implementation.
Also, very basically as we expand the size of the JBoss
code base, we also grow the complexity of what we have
to manage. It is true that during that process we also had
to simplify the way we build our application server, the
way we decompose the various modules in ways that al-
low for more flexibility.

What were some of the main technical challenges
in the re-architecture?

Sacha: JBoss has been one of the first, if not the first
company to provide an application server on the micro-
kernel. That was back in 2002. If you remember, at that
time the default design for an application server was truly
a monolithic stack, where you would receive a big block
called an application server. You would run it and that’s
it. You could not really define what you wanted to do, to
use what you did not want to use. At that time, JBoss was
really liked by developers because they could really trim
down their application servers to the bare minimum. This
was not just useful for developers. It was useful for OEM,
for ISV. They have been big consumers of that technol-
ology. We wanted to push that to the next step with the
Microcontainer. This Microcontainer is really at the heart
of many different paradigms, things like class loading,
metadata, AOP (Aspect Oriented Programming), depen-
dency injection, dependency trackings, and so on.

What the Microcontainer brings as well is a different way
to configure the application server. One of the issues
we had in the past with JBoss is that the configuration
was pretty easy to do by editing files. But, if you actu-
ally wanted to do changes on-the-fly and expect these
changes to survive restored, it was slightly more complex
to achieve.

We have revamped some of the most important
services we had in the past. For example, the messaging
implementation is brand new. We used to have a service
called JBoss MQ. We have moved to JBoss Messaging.
JBoss Messaging is truly amazing. It has impressive
performance. You can find some benchmarks on our
website. We provide some native layer integration if you
want. If not, we just revert back to pure Java behaviour.
It is fully clusterized. We can cluster in terms of load
balancing or in terms of failover. We have a very differ-
ent way to cluster this environment. It is all based on
the JBoss Cache technology that we have been using
for the last five years with JGroups for the application
server clustering. Everything is based on this common
layer. JBoss Messaging is really a step forward when it
comes to messaging. I don’t think you will find any other
Java open source messaging implementation that can
beat that level of functionality and performance.

We also included the new transaction monitor engine as
part of the app server. If you remember back in 2005,
we acquired the Arjuna Transaction Monitor both from
Arjuna and Hewlett-Packard. Now, we have integrated
those components as part of the application server. This
transmission engine is rock solid. It has more than 20
years of expertise in transaction management. It was
the first JTA and the first JTS transaction management
implementation on the market. So, we have amazing ex-
pertise there. It is also extremely fast. It provides again
JTA and JTS behaviour. That is going to be a pretty im-
portant change, especially for the banking industry.

We also did some changes in the web services arena.
We implemented a brand new web services stack, compa-
tible with EE 5. And also what we provide is an
abstraction layer so that you can actively plug multiple
backend stacks if you so desire.

We have completely decoupled the metadata manage-
ment from the application server. It’s something we call
the profile service, and this profile service is fully plug-
gable. It could be simply a set of files, it can be a data-
base, it can be a JCR repository. And the idea here is
to end up with a unified way to store metadata for one
node or a cluster or a farm of JBoss instances, and very
quickly provision new instances. That also, just to re-
late it to the other project we started called JBoss DNA,
aimed at building a complete registry and repository for
not just the application server, but our entire SOA plat-
form strategy.

If you want to read the full interview, go to
Scaling Scrum?

Ivar Jacobson

Scrum is excellent for lightweight management of small projects developed by co-located teams. However, it is not designed for large complex systems, or systems that build a SOA (banks, car companies) or PLA (product line architecture for instance Ericsson) or for large organizations with or without outsourcing. Moreover, Scrum is just one practice. You need many more to develop software. But can’t we just scale Scrum? Scaling is challenging. In the 1970’s people tried to scale ‘programming in the large’. Later people tried to scale objects to ‘objects in the large’ (mega-programming). Both never made it. Hundreds of books and even more papers were written on the subject, but it didn’t help. Instead ideas that were scalable at their roots became successful such as components and objects. At Ericsson we never had to struggle with ‘programming in the large’ since we went straight to components and later to objects.

Why is it so difficult to scale? Explaining this is not difficult but takes a lot of space. Instead let me make an analogy. Suppose you have been able to produce a new chemical substance by using a test tube and a Bunsen burner and the new substance helps with an important problem. Now you want to produce tons of the new substance. Nobody would dream of industrializing the laboratory method by simply building a larger laboratory with hundred meter long test tubes and fifty meter high Bunsen burners. Instead a chemical process would be built in which you would not recognize the original method.

Now, why do we need to scale Scrum itself? Why not let Scrum stay the pearl it is. Treat it as one of your practices in your portfolio of practices also including for example practices for requirements to test, architecture, components. Scale by composing Scrum with other practices. To the fathers of Scrum I would say: please, don’t extend Scrum with all kinds of other practices such as use cases, user stories test-driven design, etc. and brand the whole soup as Scrum. Doing so will make Scrum heavier and it will threaten the survival of Scrum because of deficiencies in these other practices. We help companies around the world to adopt Scrum. However, we don’t see Scrum as the centre of all you need. Instead, the centre is the practices you already have in your organization, the ones your people already have adopted. Replace them carefully with better practices – one by one. Scrum could be one of these better practices but not the only one.

Do you agree with this approach? Now, it doesn’t happen by itself. It is clear that a practice based approach is what is needed. We have been promoting and working with practices since 2003. Now, even IBM and Microsoft have started to join this movement. And this is really good for the industry. It is very smart!
Having fun in a great and focused team, supported by the best tools and technologies...

...that’s how we deliver better solutions faster

ACA IT-SOLUTIONS

Ever since 1998, ACA IT-Solutions offers end-to-end solutions and services, using Java technology. With an agile approach, solutions are implemented rapidly, effectively and in a flexible manner, with a focus on quality, robustness and reliability. Today, ACA IT-Solutions is probably the largest and most successful independent Java solution provider in the Benelux. With a commitment to deliver high quality IT solutions, ACA IT-Solutions effectively combines commercial and open source Java technology. ACA IT-Solutions strives to build its solutions based on open standards, using the best technologies in combination with expert people and a proven methodology.

Our activities consist mainly of:

- Identity & Access Management
- Agile Requirements Modeling
- Java software development

As a committed partner, ACA IT-Solutions helps its customers with the implementation and integration of critical enterprise solutions all over the Benelux.

GET TO KNOW US DURING DEVOXX AND GO HOME WITH AN IPOD SHUFFLE.

At ACA IT-Solutions we love to combine business with pleasure. How we put this into practice you can find out yourself at Devoxx. Want to get to know us better? Visit our booth and perhaps you are 1 of the 30 winners who goes home with a personalized iPod shuffle! At the end of each day, there will be a draw of the remaining iPods!

WE ARE CURRENTLY HIRING:

- Senior Java Solutions Engineers
- Portal Solution Engineer
- ESB/SOA Solution Architect
- Business Analyst
- Enterprise Content Management Solution Engineer
- Account Manager

We offer an attractive salary package according to your experience, including a guaranteed training budget, company car, portable, insurance package, retirement savings, etc.

For more details, visit our website at http://www.aca-it.be
Communication makes or breaks offshoring projects

The shortage of labour can be a driver to offshore or nearshore projects. Most companies have great doubts by the cost effectiveness of those adventures. More positive views are expressed on nearshoring than offshoring. One thing is sure, outsourcing is never a solution for internal organisational problems.

Offshoring is defined as sourcing service activities to a foreign, lower wage country. A few years ago India has been one of the popular countries for European IT companies. As the costs are rising there, companies are changing their view to other Asian countries like China. ACA IT-Solutions does not believe in offshoring. Serge Craehegs, marketing manager: “Many organisations use cost reduction as the main driver for taking their development projects offshore. They figure that when 70% of IT projects go way over time and budget, they might as well do so at a lower rate. At ACA IT-Solutions we believe in tackling the real underlying problem and offer a valid alternative for offshoring. Get control over your IT development process, making those deadlines and staying within available budget.” Not all companies share his vision, but full enthusiasm on outsourcing in general and offshoring in particular is hard to find. All companies seem to have their doubts. “Outsourcing is a solution for the capacity problem”, states Francis Devolder, managing partner Xplore. “But you get other problems in return.” Marco Braakman, managing director Info Support adds: “It can be useful but outsourcing is only possible when your specifications are very precise.” Bob van Rompuy, Java Competence Centre Manager at Real Dolmen thinks that theoretically one can solve the shortage with the surplus elsewhere. “We have experiences in India. In practice there were many unforeseen obstacles to overcome.”

Hidden costs

When outsourcing, are all Java projects suitable? Devolder: “I do not believe that projects in the higher ranges make sense. Application and maintenance projects are easy to outsource.” Redant: “It is difficult to say which projects are easy to outsource and which not. Big integration projects with backend systems are difficult in a remote team. Agile development in its purest form is not undoable but less suitable. Clearly defined modules of software applications are a lot easier and appropriate to outsource.” Van Rompuy would not outsource small local projects. “A project for a small social institution for example. To develop that you need to know what you are talking about. In Belgium every developer is familiar with the design and can change small failures. Big international projects with high volumes are easier to outsource.” Braakman: “Rebuilding existing systems, maintenance for these systems and projects where specifications are easy to transfer are outsourceable. New systems require a lot of communication. When you want that to outsource, the specifications must be 100% waterproof.” Communication is seen by all as the bottleneck or focal point for offshoring. There are language problems and not to forget general cultural differences. These require extra investments and generate costs. Van Rompuy: “The story of costs remains very theoretical. Maybe the personnel costs make it cheaper. But what about the travel costs, the costs for project management and the extra costs to keep up a remote system?” Also the costs of international phone calls should not be underestimated. Devolder agrees: “It is difficult to get figures for the total cost of ownership. The big advantage of offshoring lies in its perception that it is cheaper.”

Nearshoring

Near shoring is defined as sourcing service activities to a foreign, lower wage country that is relatively close in distance or time zone or both. Van Rompuy is very clear: “I believe in the potential of nearshoring. Many disadvantages from offshoring are diminished such as similar language, easier communication and less travel time. The cost advantages might be smaller but the capacity and quality are great.” The legal aspects of outsourcing are easier in nearshoring as there are European laws and regulations. Braakman: “You also have to take care that you obtain the Java source code afterwards. Maintenance will otherwise be very difficult. One should agree upon the whole issue of intellectual property rights before the project starts.”

Local representative

Franky Redant from Nonillion has practical experience with outsourcing projects to Bulgaria. “Our goal is to be a European company and offshoring to Asia is not a part of that. We choose Bulgaria because the educational system is of high quality and Belgium is very well known. The costs were not our main focus.” Although European, Redant tells that the Bulgarian employees have other expectations as the Belgian ones. “Training opportunities are seen as overriding importance. Salaries stay at a lower place. Business cards and smart phones are valued much more.” Good relations with the local representative are very important. Devolder tells from his experience with Poland: “We had some citizens from Poland in our projects. One of them returned to his country and is now responsible for the local branch.” Finally international companies have more possibilities for outsourcing. Van Rompuy: “It is easier for big international companies. When they have projects with an international character they just incorporate different branches. Finally, he remarks: “I am very curious what influence of the recession will bring. Do we continue outsourcing or do we retreat to our home market?” From offshoring via nearshoring back to Belgium?
Let the show begin!

The first day at Devoxx was quite crowded, with a lot of familiar faces from last year, but also a large number of newcomers. Be on the lookout for our camera, you may be in tomorrow’s edition of Parleys.