Devoxx 2008 pulls out all stops

With Wednesday’s keynote speech about the long awaited release of JavaFX 1.0, the Devoxx Conference clearly started off on a high note. Today and tomorrow, some 3,200 Java developers from around the world gather in Antwerp, picking up all the latest news on Java.

The voice of Java developers, reads the title of this year’s conference guide. ‘Devoxx, formerly known as JavaPolis’, is added in small print at the bottom of the guide’s cover. At the kick-off of the Devoxx Conference yesterday, it became clear – as during the University days earlier this week as well – that the new name is nothing but good news for organizer Stephan Janssen and his team. “Welcome to the seventh edition… I mean, the first edition of Devoxx”, said Janssen, addressing the audience on Wednesday morning. And he had more than one reason to be cheerful. With some 3,200 visitors from 35 countries, Devoxx was already completely sold out before its actual start, confirming its status as the biggest vendor independent Java conference in the world.

The event welcomes 160 speakers, providing Parleys.com with more than enough content for the coming year. Janssen specifically welcomed 400 students from 14 Belgian colleges and universities who accepted the invitation to visit Devoxx for free. During his opening speech, the Devoxx founder also took the opportunity to announce the revival of the Belgian Java User Group. “There is a big Java community in Belgium”, he said. “We’ve been laying a little low recently. As of January 2009, we will start organizing bi-weekly evening sessions, at various locations in the country, to get the community up and running again.”

JavaFX apps premiered at Devoxx
Sun’s Chief Architect Client Software Danny Coward kicked off the conference with a keynote on JavaFX. Presenting his top ten of thing the Java developer needs to know about JavaFX, he underlined once more the importance of last week’s official JavaFX 1.0 release. During the presentation Richard Bair, Martin Brehovsky, Jasper Potts and Josh Marinacci – most of who we had seen in action already during the Devoxx University days – stepped in to put Coward’s statements into practice. Sun’s JavaFX specialists presented several applications for the very first time in front of a live audience.

Brehovsky and Potts explained how they had quickly designed the animated Devoxx scene that was projected onto the giant cinema screen in the conference rooms, demonstrating how JavaFX integrates into graphic design tools. Richard Bair demoed an online media cabinet, allowing to share all types of media content between several devices. Josh Marinacci surprised the audience even more when he demonstrated how a JavaFX application only needs to be coded once to have it running anywhere. The audience spontaneously applauded his demo when he pulled a media player out of the webbrowser, and placed it on the desktop – without interrupting playing the actual media content.

RoxorLoops set the tone
In between the presentations beatboxer RoxorLoops entertained the audience. With nothing more than his mouth, his voice and a microphone, he created a wide variety of sounds: from a full drumkit to a complete band performing a Britney Spears song. RoxorLoops was the direct link to the speaker-headed figures that we know from the media campaign that announced Devoxx 2008. As these figures are a literal representation of the voice that Devoxx claims to be for the Java developer, at the Devoxx Conference RoxorLoops was the first one to make his voice heard. And we are pretty confident he won’t be the last...
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
Every day, a steering committee member of Devoxx advises us on what you should not miss at Devoxx. With Gert Leenders’ advice for the second day of the conference, you’ll get quite busy.

First thing to attend today is the two-piece keynote about ‘effective Java’ and the ‘Java SE 7 Update’, both very interesting topics brought by two eloquent speakers – a definite no miss.

My first choice of the day was a tough one: I hesitated between the sessions ‘Towards a dynamic VM’ and ‘HTML 5 Web sockets’ but my web background finally made me go for the latter (HTML 5). I’m pretty confident that Jonas Jacobi will keep us listening.

Second pick of the day proved even harder to take... In the end I decided for ‘Don’t do it – Common Performance Antipatterns’ by Alois Reitbauer. I’m not familiar with this speaker but his abstract, in particular the performance topic, really awoke my interest. Don’t we all deal with performance issues from time to time?

Though I’m not a true SOA buff, but as this architecture keeps gaining importance, my third session will be the JBoss SOA Platform – An Overview and Methodology by Mark Little, a solid option for anyone with even the slightest interest in SOA.

Time for a short break now, followed by an easy selection for me: since both BPM and REST tickle my fancy, I look forward to attending ‘BPM 2.0 – A REST based architecture for next generation workflow management’ by Christina Lau. The abstract sounds very promising so this session should be quite entertaining.

Lack of time and an abundance of information on offer have always been obstacles for me to stay on track with all of today’s IT evolutions. Even at Devoxx I’m sure you have faced the dilemma of major interesting sessions being scheduled at the same time, making it impossible to attend both. For that Parleys has brought me the solution to virtually attend all Devoxx’ sessions (and a lot more) wherever and whenever I want! So my last choice of the day is an obvious one: ‘The next versions of Parleys.com’ by Stephan Janssen.

Enjoy your day at Devoxx! ☝️
You program in Java, but still use a relational database.

Unlike relational databases, InterSystems Caché® is a perfect match as the back end for object-oriented programming. It’s the world’s fastest object database, and runs SQL queries up to 5 times faster than relational databases. Plus, with an innovation by InterSystems called Jalapeño™, Caché persists Java objects without relational mapping – reducing development time for Java programmers by as much as 40%.

Caché delivers massive scalability on minimal hardware, requires little administration, and incorporates a rapid Web application development environment. It’s available for UNIX, Linux, Windows, Mac OS X, and OpenVMS – and is deployed on more than 100,000 systems ranging from two to over 50,000 users.

We are InterSystems, a global software company with a 30-year track record of innovations that enrich applications.

Meet us at DEVOXX – booth 12 and attend our presentation on December 10 at 2 pm in room 6. Visit InterSystemsBenelux.com/DEVOXX to learn more and win a skype phone.
A RESTful Web service is a network based service that utilizing REST architectural style. The bounding of the architecture induces architectural properties that are highly desirable for services of the Web.

An engineer who understands the Web based standards and the REST style can make informed and predictable engineering decisions. In this respect REST is just one of many tools available in that engineering toolbox, but it is a very powerful and often misunderstood tool. However, even if an engineer has a good grasp of the Web and REST, it can be frustratingly hard to develop RESTful Web services using some Java APIs. Such APIs, while not necessarily restricting the application of the REST style, are either too low level and require a lot of seemingly unnecessary effort on the part of the engineer, or encourage non RESTful approaches, such as the common use of cookie-based sessions.

JAX-RS, the Java API for RESTful Web services, is a high level annotation driven API that makes it easier to develop RESTful Web services in Java. While an API by itself is not RESTful, the JAX-RS API brings key REST data elements to the fore as Java artefacts and therefore can help guide the engineer who may not be well versed in the Web and REST as well as being intuitive for more advanced engineers. This article presents an overview of the most important REST constraint, the properties it induces, and data elements with respect to the Web that are relevant to JAX-RS. Then an overview of the JAX-RS API is presented with reference to those data elements.

Uniform interface constraint

The uniform interface constraint bounds the architecture of RESTful Web services so that a client, such as a browser, can utilize the same interface to communicate with any service. This is a very powerful concept in software engineering that makes Web based search engines and service mash-ups possible. It induces properties such as:

- simplicity, the architecture is easier to understand and maintain;
- modifiability or loose coupling, clients and services can evolve over time perhaps in new and unexpected ways.

Further constraints are required, that introduce data elements highlighted in bold, as follows:

- for every resource, identified by a URL;
- a client exchanges with the resource, via HTTP requests and responses using the fixed set of HTTP methods;
- one or more representations, identified by media types;
- the contents of which link to further resources.

The above process repeated over and again should be familiar to anyone who has used a browser to fill out HTML forms and follow links. That same process is applicable to non browser based clients.

A RESTful Web service is a network based service that utilizes Web-based standards, such as URLs, media types, HTTP, HTML, Atom and XML, and whose architecture is bounded by the constraints defined by the REST architectural style. The bounding of the architecture induces architectural properties that are highly desirable for services of the Web.

An engineer who understands the Web based standards and the REST style can make informed and predictable engineering decisions. In this respect REST is just one of many tools available in that engineering toolbox, but it is a very powerful and often misunderstood tool. However, even if an engineer has a good grasp of the Web and REST, it can be frustratingly hard to develop RESTful Web services using some Java APIs. Such APIs, while not necessarily restricting the application of the REST style, are either too low level and require a lot of seemingly unnecessary effort on the part of the engineer, or encourage non RESTful approaches, such as the common use of cookie-based sessions.

JAX-RS, the Java API for RESTful Web services, is a high level annotation driven API that makes it easier to develop RESTful Web services in Java. While an API by itself is not RESTful, the JAX-RS API brings key REST data elements to the fore as Java artefacts and therefore can help guide the engineer who may not be well versed in the Web and REST as well as being intuitive for more advanced engineers. This article presents an overview of the most important REST constraint, the properties it induces, and data elements with respect to the Web that are relevant to JAX-RS. Then an overview of the JAX-RS API is presented with reference to those data elements.

Uniform interface constraint

The uniform interface constraint bounds the architecture of RESTful Web services so that a client, such as a browser, can utilize the same interface to communicate with any service. This is a very powerful concept in software engineering that makes Web based search engines and service mash-ups possible. It induces properties such as:

- simplicity, the architecture is easier to understand and maintain;
- modifiability or loose coupling, clients and services can evolve over time perhaps in new and unexpected ways.

Further constraints are required, that introduce data elements highlighted in bold, as follows:

- for every resource, identified by a URL;
- a client exchanges with the resource, via HTTP requests and responses using the fixed set of HTTP methods;
- one or more representations, identified by media types;
- the contents of which link to further resources.

The above process repeated over and again should be familiar to anyone who has used a browser to fill out HTML forms and follow links. That same process is applicable to non browser based clients.

A RESTful Web service is a network based service that utilizes Web-based standards, such as URLs, media types, HTTP, HTML, Atom and XML, and whose architecture is bounded by the constraints defined by the REST architectural style. The bounding of the architecture induces architectural properties that are highly desirable for services of the Web.

An engineer who understands the Web based standards and the REST style can make informed and predictable engineering decisions. In this respect REST is just one of many tools available in that engineering toolbox, but it is a very powerful and often misunderstood tool. However, even if an engineer has a good grasp of the Web and REST, it can be frustratingly hard to develop RESTful Web services using some Java APIs. Such APIs, while not necessarily restricting the application of the REST style, are either too low level and require a lot of seemingly unnecessary effort on the part of the engineer, or encourage non RESTful approaches, such as the common use of cookie-based sessions.

JAX-RS, the Java API for RESTful Web services, is a high level annotation driven API that makes it easier to develop RESTful Web services in Java. While an API by itself is not RESTful, the JAX-RS API brings key REST data elements to the fore as Java artefacts and therefore can help guide the engineer who may not be well versed in the Web and REST as well as being intuitive for more advanced engineers. This article presents an overview of the most important REST constraint, the properties it induces, and data elements with respect to the Web that are relevant to JAX-RS. Then an overview of the JAX-RS API is presented with reference to those data elements.

Uniform interface constraint

The uniform interface constraint bounds the architecture of RESTful Web services so that a client, such as a browser, can utilize the same interface to communicate with any service. This is a very powerful concept in software engineering that makes Web based search engines and service mash-ups possible. It induces properties such as:

- simplicity, the architecture is easier to understand and maintain;
- modifiability or loose coupling, clients and services can evolve over time perhaps in new and unexpected ways.

Further constraints are required, that introduce data elements highlighted in bold, as follows:

- for every resource, identified by a URL;
- a client exchanges with the resource, via HTTP requests and responses using the fixed set of HTTP methods;
- one or more representations, identified by media types;
- the contents of which link to further resources.

The above process repeated over and again should be familiar to anyone who has used a browser to fill out HTML forms and follow links. That same process is applicable to non browser based clients.
Parleys.com v3 previewed at Devoxx

The new release of Parleys.com includes spaces and channels, once again broadening the website’s use in the developers’ community.

“It’s our aim to keep adding functionality to Parleys.com”, explains Devoxx founder Stephan Janssen who wrote the first version of the site. “In v3 we decided to rewrite the software, and add spaces and channels.” The new functionality allows organizations or groups – starting with national Java user groups – to create their own spaces on Parleys.com. In these spaces, they can add channels for the streaming of talks. “We built a publication client, so everyone will be able to publish the talks themselves.” The publication client brings audio, video and slides together on one timeline. A new talk can be previewed first, and then published. “It’s all quite easy to use”, says Stephan Janssen.

The JUG space is freely available for all Java user groups from around the world. Commercial organizations are given the opportunity to rent their own space on Parleys.com. “How they manage the spaces and channels will be entirely up to them. They can choose to implement a pay per view model, just as they can offer the content for free.”

Inspiring visitors and presenters

Parleys.com was first announced on JavaPolis 2006. Devoxx founder Stephan Janssen developed that first version in his spare time. In its first year, the website received a million hits from 70,000 unique visitors. It had an offer of 120 hours of online Java talks. Two years later, Parleys.com has grown quite a bit. The site has over 200,000 unique visitors. The site’s offering has doubled, with more than 250 hours of online Java talks from around the world, resulting in about 7 TB of video streaming per month. The talks on the site have each been viewed 3,000 times on average. The most popular talks have been viewed up to 15,000 times and more.

Stephan Janssen: “When we first started out with Parleys.com, some people feared that the website would have a negative impact on the number of visitors we have at Devoxx. What happened was just the opposite. People who saw the talks on the site, decided they wanted to be at the next conference. The fact that the talks are viewed that much after the conference is of course also an extra motivation for the presenters. Over the years, Parleys.com has become an important marketing channel for Devoxx.”

Ruby doesn’t need IDE support

Reading through Ruby developer communities and forums, one can get the impression that most Ruby and Rails developers write code in text editors and use command-line tools for all the rest. And they seem quite happy with their tools of choice, so why bother talking about IDEs?

Of course, using a text editor and the command-line works just fine for Java developers as well. But almost every Java developer uses an IDE nowadays. There are many different reasons: productivity, integration, ease of use, etc. Every developer chooses his own reasons and his own IDE. JetBrains builds tools for developers who want to stay close to the code, use the best editor available and automate the routine tasks. So, for those who want an IDE, JetBrains has announced the first public preview of RubyMine, a dedicated IDE for Ruby developers.

Why use an IDE? One of the things you can’t get without a smart IDE is fast feedback about errors in your code. With a dynamic language such as Ruby it is even more of an issue. RubyMine helps to speed up coding by suggesting relevant code completions, but also analyzes code to see where it may cause errors on runtime. In fact, RubyMine integrates all the tools that a Ruby or Rails developer needs to work faster — an editor with smart code completion and refactoring, debugging tools, version control integration, and more. Right now RubyMine is in the pre-beta stage, Public Preview is a good time for people interested in working together to help make an IDE that’s perfect for them. It’s not ready for most developers, so if you’re looking for a nice, clean, polished product then you’re going to have to wait until Q1 2009. But if you’re the kind of person that really likes to dig into a tool at the earliest stage, you should check out the planned feature list below, and meet the community in the early access preview: www.jetbrains.com/ruby.

Current and Upcoming Features

- Intelligent editor, with code completion, code snippets and refactorings;
- Web development with Rails, including best-of-breed JavaScript, HTML and CSS editing and preview;
- Smart navigation with Rails-specific project view and quick model-view-controller jumping;
- Ruby debugger, with full support for Rails applications debugging and easy-to-use interface;
- On-the-fly code analysis providing error-free coding;
- VCS integrations, covering Git, Perforce, Subversion and CVS;
- RSpec and Test::Unit support, with GUI-based test runner.

One of the important, though not so visible, changes inside IntelliJ IDEA 8 is the IntelliJ platform extraction. It allows building new IDEs for other languages and RubyMine is just the beginning.
An overview of the Java Platform, Enterprise Edition

Roberto Chinnici

The Java EE 6 platform is the latest in a line of Java enterprise platform releases dating back to 1999. In the first three releases, the main focus was on making the platform more powerful and comprehensive, so that enterprise developers would have everything they need to get their job done.

Many of the new requirements centered on expanding the number of enterprise system with which a Java EE application server can interface. This direction explains the addition of the Connector API in J2EE 1.3 and of SOAP web services in J2EE 1.4. Starting with Java EE 5, the main focus for a release has shifted towards simplification, under the moniker of ‘ease of development’. In Java EE 5, ease of development has manifested itself with the addition of several new, annotation based APIs sharing a common set of patterns: Enterprise JavaBeans (EJB) 3.0, Java Persistence Architecture (JPA) 1.0 and Java API for XML-based Web Services (JAX-WS) 2.0. Additionally, one of the main features of Java EE containers, that of offering access to shared resources such as database connections and message queues, has too been recast using annotations. Besides placing more of the information in the source code itself, the use of annotations greatly reduces the need to edit the XML-based deployment descriptors, one of the traditional sources of annotations.

With respect to the growing demands made on web containers, Servlet 3.0 adds an asynchronous programming model to the existing synchronous one, so that long-lived services such as chat rooms can be implemented without wasting resources, in particular threads, on idle connections.

Many improvements

In a completely different direction, the last few years have seen a growth in the recognition of the architectural model of the web, which goes by the name of REST (Representational State Transfer). In parallel, the number of web application offering a programmatic, HTTP-based interface to non-browser clients has grown explosively, resulting in entire new data formats and protocols such as RSS, Atom and AtomPub. Java EE 6 introduces an entirely new API focused on RESTful services, the Java API for RESTful Web Services (JAX-RS) 1.0. Thanks to this API, developers can now easily model their data as a set of persistent resources and expose them to their clients over the web through a uniform interface.

JavaServer Faces (JSF) 2.0 too contains many improvements in the area of ease of development. The popular facelet page description language is brought into the standard, as is support for Ajax components, already prototyped by several toolkits. JSF 2.0 also pays more attention to thread stealing and, alas, often overlooked) ease of development issues, such as offering better error reporting at development time. The much awaited Bean Validation JSR (JSR-303) offers a single validation model applicable across the web tier (JSF, JAX-WS and the persistence tier (JPA)).

In our drive towards offering the best possible web platform, we haven’t forgotten the business tier. EJB 3.1 continues the drive towards simplification started in the previous release. Several common patterns are captured and simplified by the introduction of specific annotations for them, e.g. singleton beans and calendar-based timers. There is also a new, no-interface view of a local EJB component, which cuts the number of Java classes needed to define a component down to a single one.

New features

We are also looking at overcoming a different set of factors that in the past has limited adoption of EJB technology. In particular, the success of the EJB 3.0 programming model has renewed interest in using EJB components in web applications. EJB 3.1 removes a significant barrier by allowing EJB component classes to be packaged directly inside a web application, obviating the need for multiple levels of packaging (i.e. having to place a war file and an EJB jar inside a larger ear file). We are also looking at Web Beans 1.0 to further bridge the gap between JSF and EJB technologies, making it possible to use EJB components inside the rich, contextual web application environment.

Simplification

With Java EE 6, we intend to continue on this path of simplification without forfeiting any of the power inherent in the platform. Whereas the previous version focused on the business tier, in Java EE 6 the center of attention is the web tier. The first visible sign of the importance we place on the web tier is the creation of the Java EE 6 Web Profile.

Profiles are a new feature in the Java EE 6 platform. They are bundles of technologies focused on the needs of a specific community, such as the web developer community. With profiles, developers get a smarter, simpler, more focused platform suited to their needs. Profiles extend beyond development into deployment, by enabling Java EE application server vendors to ship small footprint versions of their products tailored to specific production uses. Although the contents of the Web Profile haven’t been finalized at this time, you can expect it to offer a complete set of APIs that covers all the needs of modern web application development.

DIY

There are many more exciting new features in the Java EE 6 component JSRs mentioned above, as well as in those that we couldn’t discuss in detail, such as JPA 2.0 and Connectors 1.6. Space does not allow me to do justice to the wealth of new features in the Java EE 6 platform. I’d like to invite you to download the specification and early access release of these technologies and try them out for yourself!

Roberto Chinnici: “With Java EE 6, we intend to continue on a path of simplification without forfeiting any of the power inherent in the platform.”
Correos: a new IT corporate model

With more than 66,000 professionals, 5,900 million dispatches a year and a consolidated turnover of 2,115 million euro in 2007, Correos, Spain’s public postal service provider, is one of the largest national organizations. In order to leverage its activity and provide a better quality service to both citizens and companies, the entity established a comprehensive IT purchasing strategy taking place between 2004 and 2009.

Also known as Plan Integra, it brings a total 240 million euro investment, assigned to updating hardware and software resources, operative applications and management systems. All operating applications are based in J2EE and Oracle databases, in order to find out everything going on with the business in real time. As Santiago Pereira, Head of Software Architecture Area at Correos explains: “The commitment to Java during the last five years has been crucial, since all mission critical operative applications are developed over J2EE platform.”

Previously, Correos corporate applications were based on tailor made software developments over Microsoft’s DNA architecture and mostly built on Visual Basic programming language.

Centralized infrastructure

“The main reason for implementing Java was that it allowed us to deploy a centralized infrastructure, so we were able to find out and manage everything going on within the business in real time. Betting for Java was the best and most secure way to achieve it”, says Correos’s Software Manager. Two main operative applications at Correos are SGIE (Integral Dispatching Management System) and Iris 6. The first one supports all operations managed by Correos’s delivering units, made up of more than 2,000 centers. The deployment has just been completed, allowing to merge different logistic systems all over Spain. On the other hand, Iris 6 is the application in charge of the postal admission process, managing Correos’s customer service at the postal offices network. Right now, it’s going through a centralization and automation transformation in order to replace present day bar codes with RFID technology. Besides, the applications messaging delivery system is based on the JMS standard, relying on IBM Websphere Message Broker as an advanced Enterprise Service Bus (ESB). “This solution is the foundation for messaging delivery. All applications are linked to the broker, which supports routing, message transformation and protocol adjusting policies. This allows us to get applications very well stuck but less fit.”

Homogeneous development

SGIE and Iris 6 are developed over Java and Oracle databases, as well as other applications such as Mercurio Track & Trace system which provides full visibility of the dispatches status at any time and SIE, statistical solution in charge of generating activity reports. To achieve it, “our corporate developing tool is Rational Application Developer, which is actually being adapted so we can leverage development productivity and get a more homogeneous development process”. This is just because Correos places its developments under contract with several consulting companies such as Indra, Atos or Evers, “and it’s really important to streamline the submitting applications”, as Pereira emphasizes. Besides development homogeneity, it is crucial to rely on a solid availability of the corporate infrastructure, supported by IBM pSeries servers, AIX operating system and IBM WebSphere Application Server. “To help us with this task, we’ve hired IBM services for development supporting issues, architecture design and systems certification.” The agreement was renewed in June for 1.8 million euro, replacing the previous one (extended through three years for 8.4 million euro) and providing software, hardware and services since 2005 to 2008.

Betting for SOA

Java has also performed as the integration foundation among applications, allowing the organization to use new messaging standards and web services. In this sense, “we’re increasingly convinced that a Service Oriented Architecture offers great advantages in terms of reliability and reusability, and therefore we decided to deploy an ESB to get the different applications centrally connected”. Correos is still at the very beginning of SOA, but the goal is to keep on going in this way so it can really trust the real profits of the services. Plan Integra has also considered updating all the hardware equipment and internal managing applications. In fact, in 2006 Correos increased its computing power by a factor of 25 purchasing three IBM pSeries 595 servers, two IBM TotalStorage DS storage arrays and a couple of IBM LTO 3584 tapes. In addition to more than 2,000 servers and databases and 20,000 desktops actually running.

The MPLS telecommunications network has been outsourced by Telefónica within a contract for four years and an amount of 74.3 million euro. Besides, Correos has spread out mobile providing PDAs to 30,000 postmen and it is totally renewing its data centre within a linking model. Finally, the organization is relying on SAP ERP to manage its corporate finance. SAP CRM to help its sales force (replacing Siebel) and SAP Human Resources software in order to bring human capital management and payroll processes under control. Therefore, Java, Oracle and SAP make up the future IT scope Correos will keep on betting for. As Pereira concludes, “this strategy has no expiration date, whilst the business keeps running we will be supporting our operative applications on J2EE.”

From 240 million euro provided by Plan Integra, Correos has already invested close to 33 million euro in J2EE developing infrastructure, just considering main business systems. In global terms, there are several key business benefits from deploying Java – less developing times, more performance and scalability – but much greater quality, operations control and service level for final users overall. The IT purchasing strategy brings a total 240 million euro investment between 2004 and 2009.

Correos main figures

- More than 66,000 professionals
- Over 2,200 multiservice offices
- 13,000 delivering vehicles
- 25 million daily dispatches and 5,900 a year
- 2007 revenues: 2,115 million euro
- 2007 profits: 104 million euro
- 2007 investments: 233 million euro
- Estimated internet business turnover: 5 million euro during 2008

New data centre in hosting

With a budget of 20 million euro, Correos is deploying its new principal data centre. From that total, 5 million euro belongs to the hosting agreement with Interxion. 10.6 million corresponds to the purchase of new hardware and storage IBM arrays, and the rest to communications and security issues. The project will double computing processing power, storage and backup systems thanks to new equipment adding to 200 servers recently consolidated and virtualized by HP over VMware technology.

The new data centre will be running at the end of the first quarter of 2009. Up to date, Correos held its main data centre at its general headquarters and a second backup one located at Telefónica’s offices.
Think big, buy small

Ivar Jacobson

Let’s be realistic. In a column like this I can only give you a hint of what you as a buyer need to do. It is amazing how many outsourcing agreements would benefit from this basic advice.

1. Split the work into smaller, separately defined pieces that potentially can be given to different vendors.

2. Separate work that is exploratory in nature from work that is mere production. Production can be calculated at a fixed price, whereas exploratory work can’t reasonably be calculated in advance. For instance don’t ask for a fixed price until you know the key requirements and until you know how you want it to be built.

3. It is beyond human ability to specify all requirements for software upfront. Don’t accept a contractual model where all requirements have to be specified and agreed upon early. Even worse is to agree on requirements upfront and then pay an unspecified amount for each change.

4. Outsourcing software development without an architecture is like asking someone to build a house without an architecture. You would never do that. However, for software, drawings are not enough. You need to make sure that the architecture can be realized. All important risks (performance, etc.) must have been eliminated. That requires you need some executable code.

5. As for a house, you need to inspect at certain points to make sure that the work is on track. Inspection in our case must include executable code to be meaningful.

Now, this is not easy to do. Still, this is just the basic. For large outsourcing projects such as building an enterprise wide SOA system, the required competencies are much higher.

6. You need to be able to split the work among several subcontractors. Thus, you need to set a building standard. Otherwise different vendors will deliver non-compatible software. It is not enough to specify what platform to use, but you need to specify how to make drawings, what design practices to use, when inspection needs to happen and more.

Think big, buy small. Western companies outsourced big M$ projects to India. Japanese companies are not giving up control that easily. Their outsourcing contracts to China are in the order of $100k. Many western companies are now taking back their software to maintain and develop it at home. The new strategy is to insource the business critical applications, but outsource less critical software. Reflecting on this, it seems we are going back to where we were before the outsourcing frenzy started. Again, if you don’t have this competence or you cannot get it, don’t even bother to outsource your software development. You need to be smart!

Ensure that the architecture can be realized. All important risks (performance, etc.) must have been eliminated. That requires you need some executable code.

5. As for a house, you need to inspect at certain points to make sure that the work is on track. Inspection in our case must include executable code to be meaningful.

Now, this is not easy to do. Still, this is just the basic. For large outsourcing projects such as building an enterprise wide SOA system, the required competencies are much higher.

6. You need to be able to split the work among several subcontractors. Thus, you need to set a building standard. Otherwise different vendors will deliver non-compatible software. It is not enough to specify what platform to use, but you need to specify how to make drawings, what design practices to use, when inspection needs to happen and more.

Think big, buy small. Western companies outsourced big M$ projects to India. Japanese companies are not giving up control that easily. Their outsourcing contracts to China are in the order of $100k. Many western companies...
Developer network: ‘try it, you’ll like it’

We created a developer network to share information and to download and distribute the Appcelerator solution. Zuercher moves to http://try.appcelerator.org. “You’ll find a ‘try it, you’ll like it’ page where you can take Appcelerator for a spin. You pick a condition and an action and see the result in real-time.” The app command line tool provides the means to create your own project and to update the distribution locally.

After installing the service and the plugin from the developer network, you can pull down the service. The next step is to create your own project. Once the project is created one can add the javastream plugin to the project that will merge xml and other. Once deployed, the standard user interface will come up and then you’ll be going from nothing to something very fast. The next step will be to author your webpages. Further on one can consider decomposing the application using content pages.

Open ended widgets

Zuercher demonstrated how the client side service broker enables local messaging and how mock services can be defined in either html or javascript. The client code is 100% reusable. There is also a core set of widgets available. “I think they are fundamentally needed.” The widget framework is lightweight and will only load when needed.

Widgets leverage the web expression languages, so you can communicate with them. The framework is open ended so people can write their own widgets.

The Core Widgets are broken down in the ‘visual’ and ‘non-visual’. The Visual Widgets are content, data table, tab panel, pagination, panel, charts, ... The Non-Visual Widgets are: data cache, http (enables to reuse existing Web 1 services), if (for conditional), message (publishing messages in response to events), script (to define complex business logic), state machine which is useful in event driven architecture.

Forms and localisation

You take a traditional form-based notation and move that into a messaging supporting transformation. Appcelerator enables this through matched fieldset attributes. Forms also support validation (when using data adhering to a particular set of rules) decorators and activators. The latter will enable or disable individual controls, like the ‘submit’ button.

“Already identified the major use of resource bundles inside the application to build for internationalisation”, Zuercher recalled. Define a javascript file a resource bundle. That’ll be a traditional JSON object with individual name enrichment) or choose for app:get.

Beans, debugging and adding

Spring beans implement individual service methods. The spring config file is going to be in your project’s config directory (config/spring.xml) and the java source will be in app/services. Services methods are annotated with @org.appcelerator.annotation.Service. The architecture allows for several approaches to do transformations. One can use course grained objects or one can opt for the fine grained approach. “If you use Hibernate and you’ve got some fine grained model objects you can leverage that as well.” For custom serialisation one can use the RawMessageDataList and RawMessageDataObject.

In debugging one can use service tester to simulate web client interactions and user gestures. “In addition to that I can recommend using our abstracted logger interface in javascript.” For stepping through code Zuercher recommends $MQ and $MQL. debug=1 is interesting to troubleshoot items and see detailed information. Furthermore there is the java-perf plugin for logstats=1 and one also can expand posts in the individual FireBug console.

How To

How to add Appcelerator to an existing spring project?

There is the web.xml and libs. There are also three different servlets: AjaxServiceTransportServlet, DownloadTransportServlet and UploadTransportServlet. Integration alternatives? You can do it in a traditional manner but you can also colocate (for Web1.0/ MVC enrichment) or choose for app:get.

Have a look at http://appcelerator.org. There’s also the blog.zuerchtech.com
The Java world develops rapidly and Java professionals have to stay up to date. The preferred skills are very dependent on the needs of a customer. To be a good Java professional do not stop at knowing the syntax. Employers provide training directed to the needs of the business. Events like Devoxx and JavaOne are very popular training places.

The java specialist does not exist. Serge Liberloo, senior Java development architect at Computer Support: “Java is broad and keeps on changing. Tim Pijpops, development team lead at Cegeka: “It is not realistic to think that you can know everything. A good conceptual basis is most important.” A few years ago a Java developer could be a real technical oriented person. Those times are over. Andy Stynen, software development manager at Dataflow: “A Java professional should not only be a guru in programming, but with the use of other methodologies such as scrum a java professional should also be able to understand business cases, make critical estimates and think together with the client or end-user.”

The amount of technologies grows every year and has consequences for the professional. Pijpops: “A Java specialist must not only know the technologies but also know how to choose between them. The question to be answered is: ‘Which is the best choice to provide business value for the customer in the most productive way?’” Sun recognized Java Champion Bert Ertman who is technology manager of Info Support’s competence centre Java sees obstacles: “It is dangerous that in the Java world you can get new technologies so easy. You have to be strong, to make the most rational choice for the customer and not the most beautiful one.” Stynen agrees: “New techniques should always be looked at critically. You should use technologies, for what they were meant for. Not just jump on every new one. In this view, new technologies, techniques and skills are important and interesting for customers, only when they need them.”

Preferred tools
When asking the Java specialist about the preferred tools of the moment, the answers differ. Pijpops: “In the past, there is a fight between fans from Java EE and fans from Spring, but now Java EE is also evolving towards a lightweight architecture. Most important is to be pragmatic, the customers needs and wishes should have priority. “What are their favourite tools?”, asks Stynen: “This is very difficult, most of the developers have their own preference, based on their experience. But there are definitely a couple of basic items which should be present. These are completeness, extensions, ease of work, stable and lightweight.”

Patrick Hellemans, project manager at Inter Access finds the question difficult as well: “That is not so easy to say. It is sector and company dependent. There is much attention on Rich Internet Applications but also Identity management and Security are very important as more and more data come available on the Internet. Front end solutions have fewer possibilities with older technologies.” Liberloo agrees: “Web applications are doing well. Also Flex is getting more and more attention.”

Training
With all the new technologies, Java professionals need training that is for sure. Every company has its own strategy. Craeghs: “Books are still the number one source of knowledge, closely followed by online resources such as forums and websites.” Pijpops adds: “We have reading groups on various themes such as refactoring, design patterns, user stories and domain driven design. Every two weeks we meet one hour to discuss the content. The knowledge sharing motivates people to stay in front.”

Apart from reading there are the classical training opportunities. At Cegeka, SpringSource gives training on a regular basis. Bart Meert, competence manager Dataflow: “We organize several extensive courses about new technologies and frameworks, several evening road shows where new technologies are presented.” Employees also visit courses in one of the many training institutes. JavaOne, Devoxx and user groups are alternatives for classical training. Liberloo: “We send our employees to JavaOne. There you make a real deep dive into Java.” All companies value internal knowledge sharing and have regular meetings.

Training on the job.
The cooperation between junior and senior specialist is popular. The shortage on the market to find qualified senior employees is behind this. Pijpops: “Training on the job is essential. We create teams from less and more experienced employees. Pair programming is also common at Cegeka. Two Java developers are working together behind one screen so you can learn and work very fast.”

The amount of training days is mostly not set. Hellemans: “Most important is that there is budget for training and it is dedicated individually. And of course there is tension between billable hours and capacity. But we see training costs as an investment and not purely as costs.”

Although training is very important to survive as Java professional, it is not the reason people change jobs. Meert: “I don’t think that just training should be the only reason to choose a specific employer. It’s the combination of well solicited trainings, good guidance and follow up and the interesting projects that follow these trainings that’s the most important in someone’s career progress.” And Serge Craeghs, marketing manager at ACA ICT-Solutions concludes: “Not as long as people have the opportunity to innovate and extend their capabilities on the job. Once people get the feeling they are ‘not moving forward’ on technological grounds during their daily projects for an extended period of time, they will look out for other opportunities. Receiving training without the possibility to apply what you have learned will not make a great difference.”
Blogging activity at Devoxx

Yesterday, the conference really kicked off with keynotes and lots of parallel sessions. The audience is also changing from the people who came to the University. Here’s a number of testimonies on why people come to Devoxx.

Blogs at Devoxx

There’s a large number of bloggers at Devoxx. Here’s a sample of what they wrote. For more blogs, go to www.devoxx.com.

“Give it a try”, says Sven Reimers.

“Being early as everyday Florian and myself got interviewed for the Parleys magazine,” writes Sven Reimers. “It was a surprise seeing that those guys still remembered us from last year. So it still seems to be a small conference - even with 3200 Java enthusiasts. The second University Day started with the session Java Performance by Kirk Pepperdine and Holly Cummins. A quite interesting presentation showing a lot of new and old aspects for performance analysis (remember everything you learned is wrong). One new tool that may be of interest is The IBM® Monitoring and Diagnostic Tools for Java™ - Health Center™ in Early Access. It provides the possibility to get monitoring directly from the VM without using bytecode injection. To use this you will need the latest IBM JDK as well. So give it a try.”

More from Sven on http://nbguru.blogspot.com/

“Comet never more”, says Koenraad Vastmans.

“This session was all about HTML 5 and particularly the communications aspects: server side events and web sockets. This session really exceeded my expectations! Both speakers knew very well what they were talking about. They are the founders of Kaazing, who have written an implementation of the SSE and WebSocket implementation that can be used in today’s browsers (so we don’t have to wait until 2022 in order to use these fancy techniques).

More on Koenraad’s blog: http://arrayindexoutofboundsexception.sky-netblogs.be/category/1291650/1/Javapolis

Viviane Festjens from the Flemish broadcasting company VRT is a regular visitor to Devoxx. Her main interest this year is security and Rest interfaces. VRT has been using Java for a long time, among other for managing playlists.

Kristen Leyn and Yolande Hillemare are the driving force behind Devoxx, working on this event all year long and attending to the organization aspects of Devoxx non-stop during the event.

Alain Tuezney from the Belgian weather service KMI plans his visits to Devoxx very carefully beforehand, writes a report and discusses new technology with his colleagues. “Usually I choose the sessions according to what we plan to do the next years.” Parallelism is high on his agenda this year. “He’s been here for the last 4 editions and was never disappointed.

Ole Jankowska is Polish but works for the Dutch company HintTech. Coming to Devoxx is part of the training program for her new job. She’s very interested in Devoxx and also wants to network with other ‘ladies in Java’.

Justin Mariaslevan from Belgian VDS Computing and Martin Davies from Framenred Ireland are old friends. Justin encouraged Martin to come to Devoxx. “This is the best Java conference in the world, he had to be here. It’s a great place to encounter new products that are perhaps better than what we are using now.”

Parleys is published by Minoc Target Media on behalf of Devoxx


For more information on Parleys and Minoc Target Media, please contact Luc Franco, luc.franco@minoc.com | Minoc Business Press | Parklaan 22, box 10 | 2300 Turnhout, Belgium | Tel. +32 14 46 23 26