In this package, you will find:

A Biography of the author of the book
A preview chapter from the book, Chapter NO.1 "What's New in Java EE 7"
A synopsis of the book’s content
Information on where to buy this book

About the Author

NDJOBO Armel Fabrice is a Design Computer Engineer and graduate from the National Advanced school of Engineering Cameroon, and Oracle Certified Expert, Java EE 6 Java Persistence API Developer. In the past, he has worked with Java EE to develop several systems among which are a helpline system and a commercial management application. He currently works on several systems based on EJB, Spring and ICEFaces solutions. In addition, he has made a technical review of the books Pro Spring Integration published by Apress and Beginning EJB 3, Java EE published by Apress. He is also a copywriter in developpeze.com (his home page is: http://armel-ndjobo.developpez.com/).

First of all, I want to thank the Lord Jesus who makes everything possible for me. I would like to thank Dr. Georges Edouard Kouamou for making me love Software Engineering. I would like to thank Fomba Ken Collins for his critiques that have helped me improve the quality of the book. I would also like to thank Serge Tsala, Armel Mvogo, and my fiancée Ines Tossam who encouraged me during the writing of this book. And finally, I would like to express my gratitude to Licel Kenmue Youbi for all his efforts.

For More Information:
www.packtpub.com/java-ee-7-first-look/book
Java EE 7 First Look

When we considered writing this book, the main objective was to present the new features of Java EE 7 platform. But while writing, we came to realize that it would be interesting to make a clear and concise presentation of the relevant specifications and how to implement them. This led us to imagine a project that will help present almost all the specifications affected by Java EE 7 and how to put them together in order to build a large-scale application.

In short, this book does not aim to show how to implement the different aspects of each Java EE 7 specification or list the best practices. Rather, it positions itself similar to the yellow pages for a city that has been built recently. In other words, this book will help you discover the innovations introduced by Java EE 7 and give you ideas to build solid applications.

What This Book Covers

Chapter 1, What's New in Java EE 7, gives an overview of the improvements made in the Java EE 7 platform.

Chapter 2, New Specifications, explains concepts concerning the new specifications that have been added in Java EE 7 and shows how they can be used.

Chapter 3, The Presentation Layer, demonstrates the implementation of the improvements brought by the Java EE 7 platform for the presentation layer specifications.

Chapter 4, The Java Persistence API, shows how your Java application can store and retrieve data from the database in a safe way and explains the innovations that have been made in the relevant specifications.

Chapter 5, The Business Layer, begins by giving a presentation of the improvement in the business layer and then demonstrates how various Java EE 7 specifications can be put together for the realization of an application.

Chapter 6, Communicating with External Systems, demonstrates how a Java EE 7 application can interact with heterogeneous systems.

For More Information:
www.packtpub.com/java-ee-7-first-look/book
Chapter 7, Annotations and CDI, explains how annotations and CDI can be used to improve the quality of applications.

Chapter 8, Validators and Interceptors, shows how the validation and interception of data can be implemented in a Java EE environment to ensure the quality of the data handled by an application.

Chapter 9, Security, demonstrates the implementation of security and setting up a personal module for security in Servlet and EJB containers.

For More Information:  
www.packtpub.com/java-ee-7-first-look/book
What's New in Java EE 7

Because of their use, distributed applications require some non functional services such as remote access, security, transaction management, concurrency, and robustness, among others. Unless you have APIs that offer these types of services, you need to implement them all from scratch and therefore, increase the number of bugs, reduce software quality, and increase production costs and time. The Java EE platform was set up to save the developer from these concerns. It is made up of a set of APIs that facilitate the development and deployment of distributed, robust, scalable, and interoperable applications.

Since its first release in 1999, the Java EE platform has evolved over time by offering a newer, richer, and simpler version than the previous one. In order for you to have an overview of the improvements in Java EE 7, this chapter addresses the following topics:

- A brief history of Java EE
- The main goals of Java EE 7
- Novelties of Java EE 7

A brief history of Java EE

Formerly called J2EE, the first version of Java EE platform was officially released in December 1999 with 10 specifications. Among these specifications, there were Servlets and JavaServer Pages (JSP) for data presentation, Enterprise JavaBeans (EJB) for the management of persistent data, remote access to business services through RMI-IIOP (Remote Method Invocation over Internet Inter-ORB Protocol), and the JMS (Java Message Service) specification, which was used to send messages.

Despite efforts and many contributions, early versions of Java EE were too complex and difficult to implement. This led to much criticism and caused the rise of competing frameworks such as Spring Framework.

For More Information:
www.packtpub.com/java-ee-7-first-look/book
What’s New in Java EE 7

Having drawn lessons from its previous failures, the platform has considerably evolved over time until the production of Java EE 5, which permitted the platform to regain its lost esteem. From this version, Java EE continues to provide easier, richer, and more powerful versions of the platform.

The preceding diagram gives an overview of the important changes made to Java EE platform since the release of the first version in December 1999. This diagram highlights the release dates, updates, and major improvements in each version. It also allows us to have an idea about the central theme behind each version.

The main goals of Java EE 7

Since May 2006, the Java EE platform has known remarkable evolution in terms of implementation. First, with Java EE 5, it greatly simplified the development of applications by allowing the transformation of a simple Java class (POJO class) into a business object through annotations or XML descriptions. Still in the line of simplification, Java EE 6 helps enrich annotations and introduces new concepts such as pruning, RESTful Web Services, CDI, EJB Lite, and configuration by exception and web profiles. This allows the platform to provide many easy-to-deploy and consume services. After the success of Java EE 6, the JCP (Java Community Process) envisaged turning the platform into a service by providing an infrastructure for cloud support. But, due to lack of significant progress in the concerned specifications, it revised its objectives. It is from the perspective of preparing the migration of the Java EE platform to the cloud that Java EE 7 focuses on productivity and HTML5 support. Having missed the big goal (that is, the migration to the cloud), it will reach its new goals through completion of Java EE 6 features and addition of some new specifications.

For More Information:
www.packtpub.com/java-ee-7-first-look/book
Productivity

Productivity in Java EE 7 has been improved upon in many ways. By simplifying some APIs such as JMS and JAX-RS, Java EE 7 platform significantly reduces boilerplate code. As you will notice in the chapters that follow, sending a JMS message can fit on one line and no longer requires the creation of several objects as was the case with JMS 1.1, where it was first necessary to create a Connection, Session, MessageProducer, and TextMessage.

Java EE 7 has integrated new APIs to better address the needs of enterprise applications relative to the processing of large amounts of data. We have, for example, the concurrency utilities, which allow the creation of managed threads within a container and give developers the ability to break down large processes into smaller units that can be computed concurrently. Similarly, there is a Java API for batch processing to manage bulk and long-running jobs.

Finally, Java EE 7 is enriched in annotations and has set a focus on configuration by exception. Whether it is for data source or batch processing, compatible Java EE 7 containers provide a number of default objects. It is even possible to produce complex applications with minor configuration.

In short, the new platform frees the developer from performing a number of tasks and the creation of several types of objects that are required for setting up an application.

HTML5 support

Some people might wonder why the support of HTML5 is so important. The answer is simple: HTML5 is the latest release of the HTML standard. More so, it offers new features that ease the building of more powerful and suitable web applications. For example, via the `<audio>` and `<video>` elements of HTML5, you can play, pause, and resume audio and video media content without the use of a third-party plugin such as Flash. Through the canvas element and WebGL library (a subset of OpenGL), you can easily integrate 2D and 3D graphics in your website. With regards to communication between the client and server, the perfect integration of WebSocket protocol in HTML5 allows us to build a web application with full-duplex P2P communication and get over some limitations of HTTP for real-time communication. Using this protocol, you will have no difficulty in realizing chat applications or other web applications that require real-time communication between the client and server, such as trading and e-commerce platforms. In terms of data exchange, the native support of JSON format in HTML5 simplifies processing of information and reduces the size of documents. Many other areas have been improved, but for now we will only mention these ones.

For More Information:

www.packtpub.com/java-ee-7-first-look/book
Given all these innovations, the support for HTML5 features was added in JSF (JavaServer Faces), a new API was added to Java EE 7 platform to build WebSocket-driven applications and another API to process JSON data format.

**Novelties of Java EE 7**

The Java EE 7 was developed as a Java Specification Request (JSR 342). It has a total of 31 specifications including 4 new specifications, 10 major releases, and 9 MRs (Maintenance Releases). All these specifications are taken into account by the GlassFish Server 4.0 (accessible via the address https://glassfish.java.net/download.html), which is the reference implementation of Java EE 7.

The new specifications introduced in Java EE are as follows:

- **Java API for JSON Processing 1.0** ([http://jcp.org/en/jsr/detail?id=353](http://jcp.org/en/jsr/detail?id=353)), which provides support for JSON processing. It offers Java EE components the ability to parse, generate, transform, and query JSON format.

APIs inherited from the Java EE 6 platform that have undergone major changes are the following:

- **Java Platform, Enterprise Edition 7 (Java EE 7) Specification** ([http://jcp.org/en/jsr/detail?id=342](http://jcp.org/en/jsr/detail?id=342)), when compared to Java EE 6, further simplifies development, adds support for HTML5, and prepares the platform to migrate to the cloud.
- **Expression Language 3.0** ([http://jcp.org/en/jsr/detail?id=341](http://jcp.org/en/jsr/detail?id=341)) was separated from JSP specification request, and it came with many changes including an API for standalone environments, lambda expressions, and collections objects support.

For More Information:

www.packtpub.com/java-ee-7-first-look/book
• JavaServer Faces 2.2 (http://jcp.org/en/jsr/detail?id=344) integrates the support for the HTML5 standard and brings features such as resource library contracts, Faces Flow, and stateless views

• Java Persistence 2.1 (http://jcp.org/en/jsr/detail?id=338) brings us the opportunity to execute Stored Procedures, create named queries at runtime, construct bulk update/delete via the Criteria API, override or change the fetch setting at runtime, and make explicit joins as in SQL

• Enterprise JavaBeans 3.2 (http://jcp.org/en/jsr/detail?id=345) introduces the ability to manually disable the passivation of stateful session beans and has also relaxed rules to define the default local or remote business interface

• Java Message Service 2.0 (http://jcp.org/en/jsr/detail?id=343) simplifies the API

• JAX-RS 2.0: The Java API for RESTful Web Services (http://jcp.org/en/jsr/detail?id=339) simplifies the implementation of RESTful Web Services and introduces new features including Client API, asynchronous processing, filters, and interceptors

• Contexts and Dependency Injection for Java EE 1.1 (http://jcp.org/en/jsr/detail?id=346) introduces many changes, some of which are access to the current CDI container, access to the non contextual instances of a bean, and the ability to explicitly destroy bean instances

• Bean Validation 1.1 (http://jcp.org/en/jsr/detail?id=349) introduces support for method and constructor validation, group conversion, and message interpolation using expression language

Only the following APIs are affected by maintenance releases:

• Web Services for Java EE 1.4 (http://jcp.org/en/jsr/detail?id=109)

• Java Authorization Service Provider Contract for Containers 1.5 (JACC 1.5) (http://jcp.org/en/jsr/detail?id=115)

• Java Authentication Service Provider Interface for Containers 1.1 (JASPIC 1.1) (http://jcp.org/en/jsr/detail?id=196) standardizes the use of some aspects of the specification

• JavaServer Pages 2.3 (http://jcp.org/en/jsr/detail?id=245)

• Common Annotations for the Java Platform 1.2 (http://jcp.org/en/jsr/detail?id=250) adds a new annotation for managing priorities

• Interceptors 1.2 (http://jcp.org/en/jsr/detail?id=318) adds standard annotation for managing the execution order of interceptors

What's New in Java EE 7

- Java EE Connector Architecture 1.7 (http://jcp.org/en/jsr/detail?id=322) adds two annotations for defining and configuring the resource adapter's resources
- Java Transaction API 1.2 (http://jcp.org/en/jsr/detail?id=907) provides the possibility to demarcate transactions declaratively and define beans whose lifecycle is identical to the current transaction
- JavaMail 1.5 (http://jcp.org/en/jsr/detail?id=919) slightly simplifies the development of sending an e-mail by adding annotations and methods

Summary

After briefly introducing the evolution of Java EE and analyzing the objectives of the latest platform, we listed all the specifications that were improved upon or added in Java EE 7. In the next chapter, we will focus on new specifications to highlight their usefulness and show how they can be implemented.
Where to buy this book


Free shipping to the US, UK, Europe and selected Asian countries. For more information, please read our shipping policy.

Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.