

Puji syukur kepada Tuhan Yang Maha Esa atas perkenan dan berkatNya sehingga ini dapat diselesaikan dengan tuntas. Sesungguhnya buku *Pemrograman Java Menggunakan IDE ECLIPSE CALLISTO: Dalam Penerapannya pada Pengembangan Aplikasi JavaEE dengan Konsep Enterprise Java Bean dan Web Service* ini dipersiapkan sebagai buku pemrograman Java tingkat menengah. Namun demikian pada dasarnya siapapun yang telah menguasai konsep pemrograman berorientasi objek menggunakan bahasa pemrograman Java serta memahami konsep dan implementasi J2SE (*Java 2 Platform Standard Edition*) dapat memanfaatkan buku ini dengan baik. Lagipula, untuk mengantisipasi para pembaca yang tidak memiliki dasar pemrograman Java, beberapa konsep pemrograman Java yang penting juga dibahas pada bab awal buku ini sehingga siapapun yang mengenal kaidah-kaidah pemrograman, terutama kaidah dasar pemrograman berorientasi objek, menggunakan bahasa pemrograman apapun dapat memanfaatkan buku ini.

Seperti telah diungkapkan di atas, buku *Pemrograman Java Menggunakan IDE ECLIPSE CALLISTO: Dalam Penerapannya pada Pengembangan Aplikasi JavaEE dengan Konsep Enterprise Java Bean dan Web Service* ini adalah buku pemrograman Java tingkat menengah. Secara umum, pembaca diharapkan sudah mengenal konsep pemrograman berorientasi objek, memahami algoritma pemrograman serta struktur data, memahami konsep pengembangan sistem basis data, serta sudah bisa membuat program kecil dengan bahasa apa saja. Selain itu, untuk lebih memahami buku ini, pembaca diharapkan sudah mengenal pemrograman untuk aplikasi banyak-lapis (*multi-tier programming*), minimal secara konsep. Dalam semua hal di atas, karena konsep yang mendasari teknologi Java EE sangat kompleks, diharapkan buku ini bisa digunakan sebagai referensi konsep Java EE, bukan sebagai petunjuk praktis pemrograman yang menjelaskan secara rinci pemrograman Java EE. Dengan memahami konsepnya secara mendalam, diharapkan para pembaca bisa mendapatkan ide yang cemerlang untuk mengimplementasikannya dalam bentuk aplikasi Java berskala besar dan aplikasi Web menggunakan bahasa pemrograman Java.

Selain hal-hal di atas, dalam buku ini kita juga akan belajar tentang penggunaan IDE (*Integrated Development Environment*) yang sangat canggih untuk menuliskan kode Java EE. IDE yang dimaksud adalah Eclipse Callisto. Eclipse Callisto merupakan suatu IDE *open source* yang

sangat canggih dan dapat digunakan untuk membuat program Java apapun; baik J2SE, J2ME, serta Java EE. Eclipse Callisto memiliki berbagai fitur yang bermanfaat untuk mengembangkan program Java dengan cara sedemikian rupa sehingga penulisan program dapat berjalan dengan efisien dan efektif. Hasil akhirnya adalah peningkatan produktivitas pemrogram dan pengembang aplikasi. Dalam buku ini kita akan mempelajari beberapa fitur Eclipse Callisto yang paling penting, terutama yang berkaitan dengan penulisan kode dan pelacakan kesalahan. Sebagai catatan tambahan: Eclipse Callisto yang digunakan dalam buku ini adalah Eclipse Europa versi 3 beserta *plug-in* proyek open source lain yang dinamakan sebagai proyek Callisto.

PEMROSESAN TERDISTRIBUSI

Inti dari pemrograman Java EE adalah pemrosesan terdistribusi. Apa itu pemrosesan terdistribusi? Pemrosesan terdistribusi (*distributed computing*) adalah revolusi dalam pemrosesan *client/server*. Pada model *client/server* ini setiap *client* mungkin menjadi *server* bagi komputer lain; tidak ada sistem *server* terpusat di mana semua *client* menggantungkan diri padanya. Pada era pertama pemrosesan *client/server* terdapat pemisahan yang tegas antara *client* dan *server* sedangkan pada era yang baru saat ini, yaitu teknologi objek terdistribusi (*distributed objects computing*), tidak ada pemisahan yang tegas antara *client* dan *server*. Teknologi objek terdistribusi diharapkan akan mendominasi dan menggantikan model aplikasi *client/server* sebelumnya karena teknologi ini lebih fleksibel, lebih efektif dan lebih efisien.

Komputasi objek terdistribusi menjanjikan sistem *client/server* yang lebih fleksibel sebab komponen perangkat lunak yang dapat digunakan ulang (*reusable components*) tersebar di segala tempat di jaringan (bahkan di jaringan internet global), berjalan pada berbagai platform sistem operasi dan perangkat keras yang berbeda, serta melakukan pengelolaan proses secara mandiri. Semua komputer dapat bertindak sebagai *client* maupun *server*. Objek-objek dapat membantu pemecahan aplikasi tunggal ke dalam banyak komponen yang lebih mudah dikelola dan dimanfaatkan. Objek-objek yang terdistribusi adalah komponen yang dapat digunakan ulang yang tersebar sepanjang jaringan dan dapat diakses oleh siapapun. Komputasi objek terdistribusi memperkenalkan peringkat tinggi dari abstraksi dalam dunia aplikasi terdistribusi.

Komputasi objek terdistribusi adalah kunci pada sistem informasi masa depan. Komputasi objek terdistribusi dapat dihasilkan dari aplikasi terintegrasi yang mungkin saja terpisah secara geografis dan berjalan pada perangkat keras serta sistem operasi dengan platform berbeda. Lebih jauh, sistem informasi harus dapat menghubungkan bagian-bagian aplikasi yang dikembangkan dengan bermacam-macam bahasa pemrograman, menggunakan basis data berorientasi objek serta basis data relasional yang berbeda, dioptimasi untuk penggunaan via internet dan dioptimasi untuk penggunaan dalam departemen-departemen yang terhubung dalam jaringan intranet.

Dalam aplikasi yang berukuran besar masa kini pada umumnya komponen aplikasi diletakkan di sekujur komputer yang masing-masing terhubung lewat sarana jaringan. Dalam hal ini pengembang aplikasi memiliki banyak sekali teknik yang memungkinkan komponen perangkat lunak di suatu lokasi berkomunikasi dengan komponen perangkat lunak yang berada di lokasi lain. Salah satu teknik umum yang dikenal adalah RPC (*Remote Procedure Control*) di mana pada teknik ini komponen aplikasi yang berjalan di suatu komputer dapat menggunakan metode milik komponen aplikasi tertentu yang berada di komputer lain. Teknik lain untuk mengimplementasikan konsep di atas adalah konsep layanan Web (*Web services*) yang merupakan konsep yang relatif baru yang memungkinkan objek dapat saling mempublikasikan dan memanfaatkan layanan yang dimiliki objek lain.

Implementasi dari komputasi objek terdistribusi tidaklah semudah kita mengembangkan aplikasi mandiri (*stand-alone*) biasa. Komputasi objek terdistribusi menuntut kita untuk memahami berbagai konsep baru; konsep yang mendukung dimungkinkannya ‘komunikasi’ antarobjek yang letaknya tersebar di sekujur jaringan komputer dapat dilakukan dengan baik. Buku ini memperkenalkan berbagai konsep pendukung komputasi tersebar itu beserta bagaimana implementasinya menggunakan bahasa pemrograman Java, lebih khusus J2EE dengan dukungan IDE Eclipse Callisto. Java EE adalah teknologi penerus J2EE dengan tambahan anotasi yang meningkatkan efektivitas serta efisiensi penulisan program.

Teknologi Java EE yang merupakan penerus teknologi J2EE sesungguhnya merupakan teknologi bahasa pemrograman Java yang biasanya digunakan untuk mengembangkan aplikasi Java tingkat lanjut,

terutama aplikasi tersebar/terdistribusi yang dapat secara sederhana dikembangkan menggunakan J2SE. Dalam hal ini tentu saja kita tidak hanya akan menggunakan perintah-perintah bahasa Java lanjut, tetapi kita juga akan berkenalan dengan konsep pemrograman Java yang lebih lanjut seperti JSP (*JavaServer Pages*), *Applet*, *Servlet*, EJB (*Enterprise Java Beans*), XML (*eXtended Markup Language*), pemrograman basis data menggunakan JDBC (*Java Database Connectivity*), pemrograman basis data menggunakan JPA (*Java Persistence API*), dan teknologi Hibernate, konsep layanan Web (*Web service*), konsep pemrograman *multi-tier* dan sebagainya. Dalam buku ini kita akan berkenalan dengan konsep yang mendasari pengembangan aplikasi Java berskala besar serta konsep yang mendasari pengembangan aplikasi Web menggunakan bahasa pemrograman Java.

DESKRIPSI BAB-BAB DALAM BUKU

Buku ini mengajak pembaca untuk memahami konsep-konsep yang mendasari pengembangan aplikasi Java EE. Aplikasi Java EE membutuhkan pemahaman konsep yang matang agar bisa diimplementasikan dengan baik dalam bentuk kode Java terkait. Oleh sebab itu buku ini lebih menekankan pada konsep, meskipun tentu saja kita tidak meninggalkan contoh implementasinya. Secara garis besar, isi dari bab-bab yang ada di buku ini adalah sebagai berikut:

- **BAB I: ENTERPRISE JAVA BEANS.** Setelah membahas komponen-komponen di lapisan presentasi, dalam bab ini kita beranjak lebih jauh ke komponen yang mengimplementasikan logika bisnis (*business logic*) aplikasi J2EE. Komponen-komponen yang dimaksud adalah EJB yang terdiri dari 3 jenis: *Session Beans*, *Entity Beans*, serta *Message-driven Beans*. Selain itu kita juga akan membahas akses EJB, lebih khusus Entity Beans, ke sistem basis data relasional menggunakan teknologi JPA (*Java Persistence API*) dan Hibernate yang merupakan lapisan tambahan di atas JDBC yang bertujuan mengotomasi pembentukan kode akses ke basis data. Mulai dalam bab ini secara khusus kita juga akan membahas konsep yang mendasari Java EE beserta bagaimana langkah yang harus kita ambil untuk mengimplementasikannya.

- **BAB II: XML DAN LAYANAN WEB.** Dalam kenyataannya, komponen-komponen aplikasi J2EE dan Java EE yang besar pada umumnya berada di berbagai platform yang berbeda. Untuk itu diperlukan suatu cara sedemikian rupa sehingga komponen aplikasi yang berada di berbagai platform yang berbeda itu dapat saling berkomunikasi. Cara yang sering digunakan oleh komponen aplikasi untuk berkomunikasi adalah dengan saling bertukar pesan berbentuk XML atau lebih khusus lagi SOAP (*Simple Object Access Protocol*). Komponen-komponen yang dapat saling memanfaatkan itu sering dirujuk sebagai layanan Web. Kita akan membahas implementasi layanan Web ini dalam kerangka IDE Eclipse Callisto. Selain hal di atas, dalam bab ini kita juga akan belajar bagaimana caranya suatu berkas XML dapat diuraikan dan ditafsirkan oleh aplikasi yang memerlukan dan menggunakannya.

TENTANG CD PENYERTA

Dalam CD yang disertakan dalam buku ini penulis menyertakan IDE sekaligus kompiler Java yang bernama Eclipse Callisto yang penulis punyah cuma-cuma, karena bersifat *open source*, dari situs www.eclipse.org dan beberapa situs terkait lain sehingga pembaca tidak perlu takut atas pelanggaran hak cipta. IDE Eclipse Callisto ini akan digunakan sebagai basis penulisan dan pembelajaran melalui kode Java, khususnya J2EE dan Java EE yang merupakan teknologi penerusnya.

Provider	Plug-in Name	Version	Plug-in Id
	Event Model	3.0.0.v20060...	com.ibm.etools.emf.event
Aptana, Inc.	Aptana Core	0.2.7.13425	com.apтана.іde.core
Aptana, Inc.	Aptana Core Win32	0.2.7.13425	com.apтана.іde.core.win32
Aptana, Inc.	Aptana CSS Editor	0.2.7.13425	com.apтана.іde.editor.css
Aptana, Inc.	Aptana Debugging Tools	0.2.7.13425	com.apтана.іde.debug.core
Aptana, Inc.	Aptana Debugging Tools	0.2.7.13425	com.apтана.іde.debug.ui
Aptana, Inc.	Aptana Documentation	0.2.7.13425	com.apтана.іde.documentation
Aptana, Inc.	Aptana FTP Synchronization...	0.2.7.13425	com.apтана.іde.syncing.ftp
Aptana, Inc.	Aptana HTML Editor	0.2.7.13425	com.apтана.іde.editor.html
Aptana, Inc.	Aptana IDE	0.2.7.13425	com.apтана.іde.rcp
Aptana, Inc.	Aptana IDE Main	0.2.7.13425	com.apтана.іde.rcp.main
Aptana, Inc.	Aptana Internal Libraries	0.2.7.11838	com.apтана.іde.libraries.internal
Aptana, Inc.	Aptana JavaScript Editor	0.2.7.13425	com.apтана.іde.editor.js
Aptana, Inc.	Aptana Lexer	0.2.7.13425	com.apтана.іde.lexer
Aptana, Inc.	Aptana Libraries	0.2.7.11825	com.apтана.іde.libraries
Aptana, Inc.	Aptana Parsing	0.2.7.13425	com.apтана.іde.parsing
Aptana, Inc.	Aptana PHP Editor	0.2.7.13425	com.apтана.іde.editor.php
Aptana, Inc.	Aptana Regular Expressions	0.2.7.13425	com.apтана.іde.regex
Aptana, Inc.	Aptana Scripting	0.2.7.13425	com.apтана.іde.scripting
Aptana, Inc.	Aptana Server Core	0.2.7.13425	com.apтана.іde.server.core
Aptana, Inc.	Aptana Server User Interface	0.2.7.13425	com.apтана.іde.server.ui
Aptana, Inc.	Aptana Synchronization Core	0.2.7.13425	com.apтана.іde.syncing
Aptana, Inc.	Aptana Unified Editor	0.2.7.13425	com.apтана.іde.editors

KP-1

Adapun cara penginstalasian IDE Eclipse Callisto adalah seperti tertulis di bawah ini. Sebelumnya perhatikan dengan seksama bahwa spesifikasi minimum untuk dapat menjalankan Eclipse Callisto beserta plug-in lainnya adalah **Pentium IV** atau yang lebih baik/cepat, **RAM 256 MB** [direkomendasikan minimal 512 MB], sistem operasi **Windows XP**, dan ruang minimal **hardisk 1 GB**):

- Kliklah satu-satunya berkas di CD penyerta maka berkas dalam CD penyerta akan langsung mengekstraksi semua berkas yang ada ke folder **C:\Master JavaEE and ECLIPSE CALLISTO**.
- Masuklah ke folder **Eclipse Callisto JBoss FREE DOWNLOAD**.
- Klik ganda berkas **jdk-6-windows-i586** yang merupakan berkas instalasi JDK serta JRE di komputer. Ikutilah petunjuk penginstalasiannya. Proses instalasi akan melakukan instalasi JDK 6 di komputer pembaca.

- Klik ganda berkas **alzip.exe** yang merupakan berkas yang diperlukan untuk melakukan ekstraksi berkas terkompresi.
- Selanjutnya pembaca dapat mengekstraksi berkas **eclipse-win32-Callisto** ke suatu folder tertentu (nama folder terserah pada pembaca, misalnya **Eclipse Callisto**). Jika pembaca melakukannya dengan benar, masuklah pada folder tempat pembaca mengekstrak **eclipse-win32-Callisto** maka pembaca akan menemukan berkas **Eclipse Callisto** (dalam bentuk **eclipse.exe**) yang dapat langsung diaktifkan dengan mengklik ganda ikonnya. Sebagai tambahan: Agar pembaca tidak selalu harus mengaktifkan Eclipse Callisto melalui Windows Explorer, pembaca dapat mengklik kanan ikon Eclipse Callisto (dalam bentuk **eclipse.exe**) dan memilih menu **Create Shortcut** dan memindahkannya ke desktop. Suatu saat kelak jika pembaca ingin mengaktifkan Eclipse Callisto kembali, pembaca dapat mengklik ganda ikon itu di desktop.
- Agar dapat menjalankan aplikasi J2EE, pembaca dapat menginstalasi server aplikasi **JBoss**. Ekstraklah berkas **jboss-4.0.5.GA** ke folder yang digunakan pembaca untuk mengekstraksi Eclipse Callisto di atas. Jangan di direktori *plug-in* yang dimiliki oleh Eclipse Callisto.
- Agar pembaca dapat tuntas memanfaatkan aplikasi JPA (*Java Persistence API*) dan teknologi Hibernate, ekstraklah berkas **HibernateTools-nodeps-3.2.0** di direktori *plug-in* Eclipse Callisto.
- Selanjutnya, karena buku ini juga berusaha konsisten dengan teknologi *open-source* untuk mengembangkan antarmuka pengguna, termasuk untuk pengembangan berkas JSP, HTML, JavaScript, XML, dan sebagainya, menggunakan teknologi Aptana, pembaca juga harus memasang *plug-in*-nya. Caranya mudah! Ekstraklah berkas **aptana_studio_setup_windows** ke dalam direktori *plug-in* Eclipse Callisto dan kemudian instalasilah browser **Firefox** (dalam bentuk berkas **Firefox Setup 2.0.0.3**) yang diperlukan Aptana untuk *men-debug* dan menjalankan berkas JSP, HTML, JavaScript, XML, dan sebagainya tadi.
- Terakhir, jika pembaca ingin membuat diagram UML dari dalam IDE Eclipse Callisto, ekstraklah berkas **eclipseUML_E320_freeEdition** di direktori *plug-in* Eclipse Callisto.

```

C:\Documents and Settings\... [Program Files]
  [WIN XP] [Eclipse Callisto] [oracle]
  [NVIDIA] [ACROREAD] [Sun]
  [TCWIN45] [FFPC]
clapton1.jpg      1 File(s)      205.632 bytes
                  .10 Dir(s)    15.060.107.264 bytes free

D:\>cd Eclipse Callisto

D:\Eclipse Callisto>dir/w
Volume in drive D is WINXP
Volume Serial Number is 15F8-2437

Directory of D:\Eclipse Callisto

[.]                [..]                [tftp]                [eclipse]                [Tomcat 5.5] [JBoss]
[workspace]
                0 File(s)      0 bytes
                7 Dir(s)    15.060.107.264 bytes free

D:\Eclipse Callisto>cd eclipse
D:\Eclipse Callisto\eclipse>eclipse.exe -clean
D:\Eclipse Callisto\eclipse>

```

KP-2

Agar dapat memanfaatkan IDE Eclipse Callisto secara menyeluruh, periksalah *plug-in* yang telah ditambahkan ke Eclipse Callisto tadi. Caranya adalah dengan mengaktifkan Eclipse Callisto kemudian memilih menu **Help** → **Eclipse SDK** → **Plug-In Details**. Periksalah apakah semua *plug-in* sudah benar-benar dikenali oleh Eclipse Callisto. Perhatikan Gambar KP-1. Jika pembaca telah melakukan semua langkah di atas dengan benar, namun beberapa *plug-in* belum dikenali Eclipse Callisto, pembaca mungkin perlu mengikuti langkah berikut ini (Perhatikan Gambar KP-2):

- Pilih menu **Program** → **Accessories** → **Command Prompt** di sistem operasi yang pembaca gunakan.
- Arahkan prompt Windows ke direktori tempat pembaca mengekstrak Eclipse Callisto.
- Ketikkan `eclipse.exe -clean`.
- Tekan **ENTER** dan periksalah kembali *plug-in*-nya dengan memilih menu **Help** → **Eclipse SDK** → **Plug-In Details**.

Sebagai catatan, jika komputer pembaca memiliki memori yang cukup besar (di atas 256 MB) dan pembaca ingin memanfaatkannya demi kinerja Eclipse Callisto secara maksimal, cobalah buka IDE Eclipse Callisto menggunakan perintah `eclipse -vmargs -Xmx<memory`

size> dengan <memory size> yang diatur sesuai kapasitas memori yang ada. Sebagai contoh, untuk menjalankan IDE Eclipse Callisto di komputer penulis yang berprosesor **Pentium IV 3,2GHz** dan bermemori **1GB**, penulis menggunakan perintah `eclipse -vmargs -Xmx 512MB` dan IDE Eclipse Callisto berkinerja cukup baik. Perintah ini cukup dilakukan satu kali saja sebab pemanggilan IDE Eclipse Callisto di waktu-waktu berikutnya akan secara otomatis mengikuti pengaturan di atas.)

Dalam CD penyerta buku ini pembaca juga akan mendapatkan beberapa *driver* JDBC yang bermanfaat untuk menghubungkan aplikasi J2EE yang kita kembangkan ke berbagai sistem basis data relasional yang banyak digunakan di Indonesia saat ini. Saat pembaca masuk ke Bab IV, yang membahas akses aplikasi J2EE ke sistem basis data relasional, pembaca mungkin membutuhkannya sebagai sarana berlatih.

Belajar adalah proses tanpa akhir hingga akhir hidup ini. Oleh sebab itu bagi pembaca yang ingin lebih jauh mempelajari pengembangan aplikasi Java, khususnya Java EE, pembaca dapat mempelajari lebih dalam melalui dokumentasi dari vendor yang juga disertakan dalam CD penyerta buku ini. Semuanya penulis dapatkan dari jaringan internet secara cuma-cuma.

PENUTUP

Sebagai penutup, ijinkanlah penulis mengucapkan terima kasih pada beberapa orang dan lembaga yang secara langsung maupun tidak langsung mendukung penyelesaian buku *Pemrograman Java Menggunakan IDE ECLIPSE CALLISTO: Dalam Penerapannya pada Pengembangan Aplikasi JavaEE dengan Konsep Enterprise Java Bean dan Web Service* ini. Kepada Bpk. Andrian The di pusat pelatihan INIXINDO di Yogyakarta yang telah memperkenalkan konsep pemrograman J2EE pada penulis. Penulis juga mengucapkan banyak rasa terima kasih kepada seluruh karyawan di Penerbit ANDI OFFSET Yogyakarta yang telah menerbitkan buku ini serta mendistribusikannya hingga ke seluruh penjuru tanah air sehingga buku ini tampil di hadapan para pembaca.

Pada rekan-rekan kerja di Universitas Kristen Satya Wacana di Salatiga, Jawa Tengah, terutama Bpk. Hendro Tampake dan Bpk. Theophilus Herman Wellem, yang banyak memberi masukan dan kritik

membangun, serta pada bapak dan ibu Pudjosudjarwo, Nuniek, Eni, Heiman, Sonda, serta Falin dan Dhira yang kecil-lucu-menggemaskan, yang mendorong penulis menyelesaikan buku ini dengan tuntas, penulis juga mengucapkan rasa terima kasih yang sebesar-besarnya. Untuk seorang mahasiswi UKSW yang secara tidak langsung membuat penulis bersemangat menyelesaikan buku ini, penulis mengucapkan terima kasih yang tulus. Akhir kata, kiranya TUHAN memberkati kita semua! Selamat mempelajari teknologi Java EE menggunakan buku *Pemrograman Java Menggunakan IDE ECLIPSE CALLISTO: Dalam Penerapannya pada Pengembangan Aplikasi JavaEE dengan Konsep Enterprise Java Bean dan Web Service* ini!

Ergo Sum Via Veritas et Vita

Salatiga, 19 Agustus 2008
Penulis

ADI NUGROHO, ST., MMSI