

BAB 3

Studi Kasus SOA

Sebagai pembuktian integrasi berbasis SOA maka akan dibangun aplikasi bisnis fiktif berupa kredit perbankan yang dinamakan aplikasi WHUBANK. Aplikasi WHUBANK ini merupakan proof of concept dari integrasi berbasis SOA. Teknologi yang digunakan dalam membangun aplikasi integrasi berbasis SOA ini adalah Open ESB (Barrett, 2009)

Open ESB merupakan software open source yang dibangun diatas arsitektur Java Business Integration (JBI) mengikuti spesifikasi JSR 208. Open ESB merupakan basis teknologi baru yang diproduksi oleh Sun Microsystem yang sekarang telah diakuisisi oleh Oracle. Glassfish ESB merupakan bagian dari fitur Open ESB. Tujuan dari buku ini adalah menyediakan pengantar yang lengkap dalam memanfaatkan fitur JBI menggunakan Glassfish ESB. Topik-topik yang akan dibahas mencakup :

- Web services yang dibangun menggunakan modul EJB dan aplikasi web.
- Pengembangan skema XML dan WSDL
- Orkestrasi BPEL dari web services menggunakan BPEL service engine
- Deployment dan pengujian aplikasi komposit
- Perancangan interface user dengan web
- Penyimpanan data dalam basisdata melalui JDBC binding component

3.1 Skenario

Skenario menggunakan WHUBANK yang merupakan institusi keuangan yang beroperasi di Salatiga. Pemilik mempunyai ide menggunakan SOA sebagai strategi TI. Pemilik menginginkan pegawainya memahami SOA dengan menggunakan web services dan aplikasi komposit. Karena itu akan dibuat aplikasi sebagai proyek proof of concept dari penerapan SOA.

Penetapan Bunga Pinjaman: Penetapan bunga pinjaman akan diimplementasikan menggunakan web services, yang dibangun dari EJB stateless yang menyediakan logik untuk menetapkan bunga pinjaman.

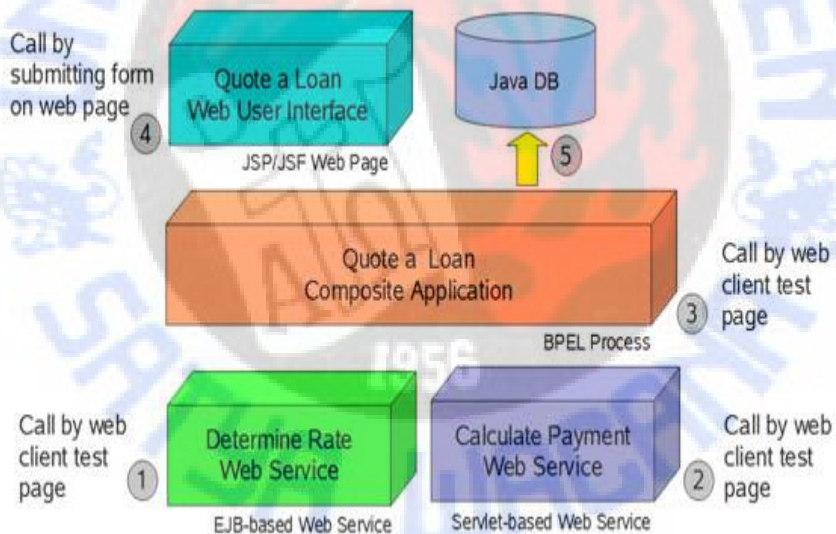
Menghitung Pembayaran Pinjaman.: Web services lainnya akan mengambil input jumlah pinjaman, tingkat bunga dan periode dan menghitung jumlah pembayaran bulanan. Rumus menghitung pembayaran menggunakan rumus umum dalam keuangan. Web services akan mengimplementasikan algoritma dari rumus tersebut. Akan digunakan aplikasi web untuk menyediakan logik penghitungan pembayaran..

Proses Peminjaman: Proses Business Process Execution Language (BPEL) digunakan untuk melakukan orkestrasi (orchestration) dengan menggunakan dua web services yang telah dibuat sebelumnya. BPEL akan menerima paramter input jenisPinjaman, jumlah, dan periode dan melakukan serangkaian eksekusi dari dua web services tersebut untuk menghasilkan output berupa pembayaranPinjamanBulanan. Pada saat runtime, BPEL service engine akan menyediakan dukungan untuk melakukan eksekusi WS-BPEL 2.0. Aplikasi ini kemudian akan diperluas dengan memanfaatkan

JDBC binding component yang akan menyimpan pinjaman ke sebuah basisdata.

User Interface: Untuk melakukan pengujian proses BPEL, akan dibangun sebuah user interface berbasis web browser menggunakan JavaServer Pages (JSP) yang dihosting dengan komponen Java Server Faces (JSF) sehingga dapat berinteraksi dengan proses peminjaman melalui interface web services.

Selanjutnya akan ditangani rangkaian skenario dari lima proyek yang dibangun berurutan. Gambar berikut ini merupakan ilustrasi komunikasi blok bangunan setiap proyek.



Akan dimulai proyek 1 dan 2 dengan membangun dua blok bangunan dasar dari web services. Dalam proyek 1 akan dibuat web services berbasis EJB yang akan digunakan untuk menetapkan tingkat bunga dari pinjaman.

Proyek 2 akan dibangun web services berbasis servlet yang akan digunakan untuk menghitung jumlah pembayaran pinjaman secara bulanan.

Dalam proyek 3, ke dua blok bangunan web services yang telah dibangun pada proyek 1 dan 2 tersebut akan diorkestrasi ke dalam sebuah aplikasi menggunakan BPEL. Proyek 4 akan dibangun user interface menggunakan web browser yang user friendly. Proyek 5 akan digunakan untuk menyimpan data peminjaman ke basisdata menggunakan komponen JDBC binding component.

Sebelum menangani 5 proyek tersebut akan dikerjakan proyek 0, yang merupakan pengantar cara instalasi dan konfigurasi software yang akan digunakan

Untuk mengerjakan ke 5 proyek tersebut, diperlukan komputer dengan spesifikasi paling tidak 2GB RAM dan sekitar 1 GB Hardisk. Walaupun JDK dan Netbeans dapat digunakan untuk sistem operasi Linux maupun Mac, namun dalam pembangunan ke 5 proyek tersebut menggunakan sistem operasi Windows XP SP3.

Selain itu juga diperlukan JDK6 dan Glassfish ESB. JDK dapat didownload dari <http://java.sun.com/javase/downloads/index.jsp> sedangkan Glassfish ESB versi opensource tidak lagi dikembangkan oleh Oracle tetapi dikelola komunitas, dan dapat didownload di alamat.

3.2 Instalasi Software

Dalam proyek ini, akan didownload JDK dan software Glassfish ESB dan kemudian diinstalasi.

Download dan install JDK (bukan JRE) sebelum melakukan instalasi GlassFish ESB. Jika belum punya JDK6, maka download dari <http://java.sun.com/javase/downloads/index.jsp>. Pengerjaan proyek ini menggunakan *JDK 6 Update 20*.

Setelah mempunyai JDK maka lakukan instalasi dengan mengklik dua kali pada file exe dari JDK6, kemudian ikuti perintah-perintahnya. Jika masih level pemula, maka ikuti default nya saja selama instalasi JDK. Instruksi instalasi dapat ditemukan di: <http://java.sun.com/javase/6/webnotes/install/index.html>

OpenESB (Glassfish ESB) tidak lagi dikembangkan oleh Oracle tetapi dilanjutkan oleh komunitas open source. Karena itu OpenESB didownload di alamat <http://hudson.openesb-dev.org/hudson/view/dev/job/openesb-installers-legacy/>

Pilih GlassFish ESB versi terakhir (pada buku ini digunakan Glassfish ESB V2.1)



GlassFish ESB v2.1 Release Candidate 2

Description This is the release candidate of the GlassFish ESB v2.1 release. It there are only a few known issues.

Notes [\(click to expand\)](#)

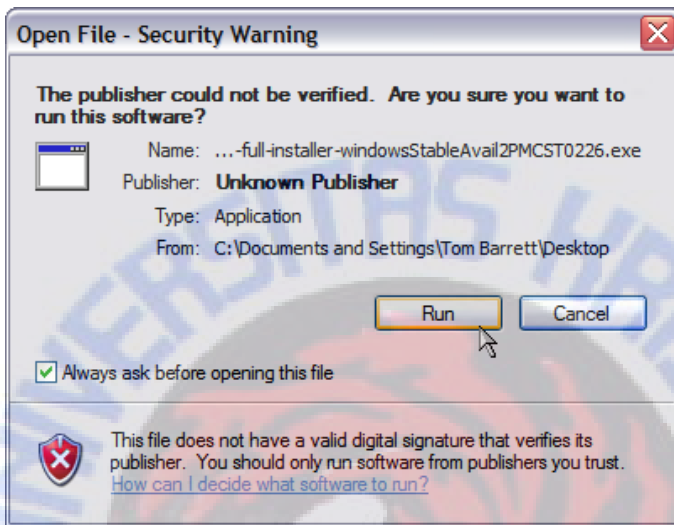
Contents [\(click to expand\)](#)

Installers

Full install, 188 Mb: (includ

Components only, 66 Mb:

Setelah download selesai, lakukan instalasi dengan melakukan klik 2X pada file kemudian klik Run.



{Lakukan klik-double pada glassfishesb-full-installer-windows.exe.

Uninstall GlassFish ESB juga mudah. Klik saja pada uninstall.exe pada direktori install netbeans.

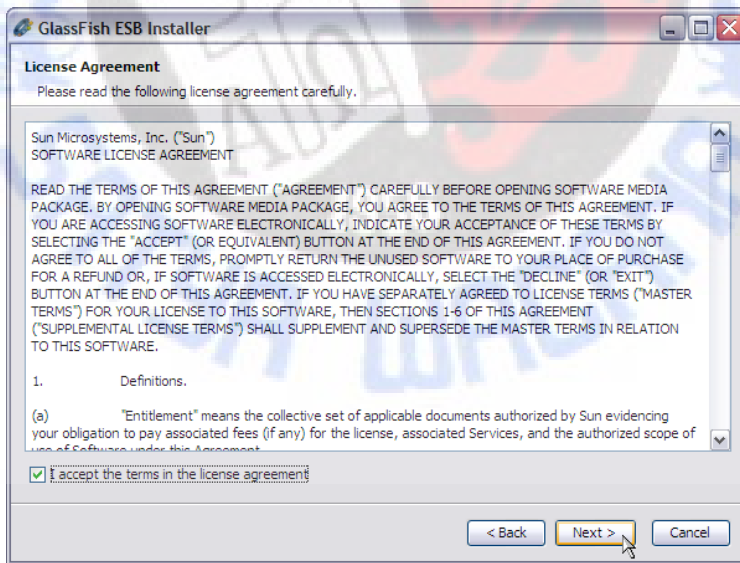
- *Akan ditandai apakah uninstall juga menghapus server Glassfish juga*
- *Jika uninstall lengkap termasuk GlassFish ESB, maka harus check box sebelum menekan tombol Uninstall. }*

Klik *Run* untuk melanjutkan. Lihat gambar:

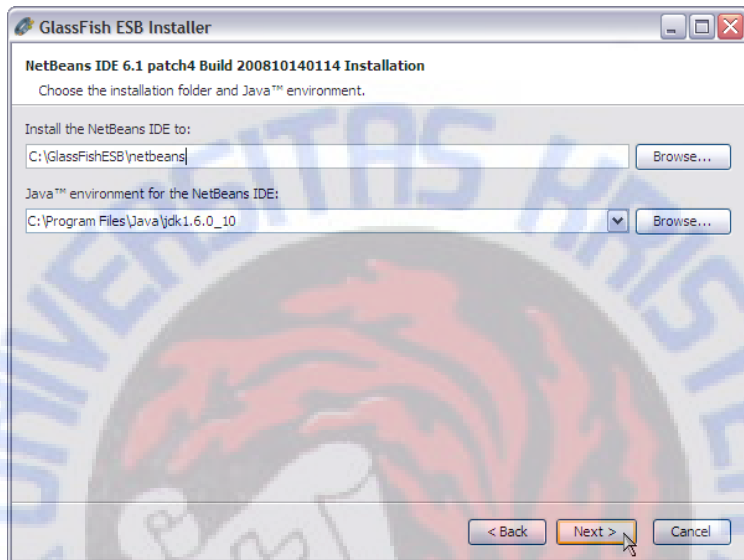


Klik Next untuk melanjutkan.

Statement lisensi muncul. Respon lisensi dengan mengklik Accept, kemudian klik Next lagi untuk melanjutkan instalasi.



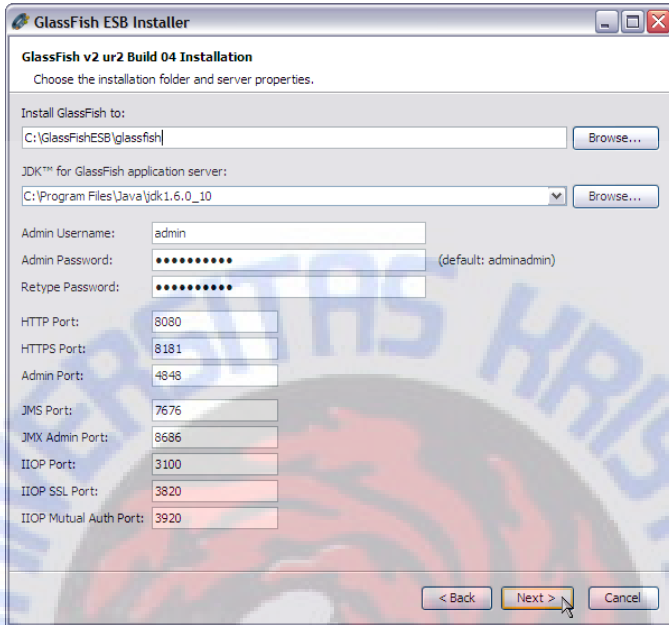
Prompt JDK dan NetBeans muncul:



{Perhatikan bahwa installer secara otomatis menemukan JDK 6 Update 20 pada mesin.}

Wizard instalasi akan menemukan instalasi JDK6 secara otomatis. Jika tidak maka gunakan tombol Browse dan temukan folder tempat instalasi JDK sebelumnya. Terima default direktori Netbenas, kemudian klik Next untuk melanjutkan.

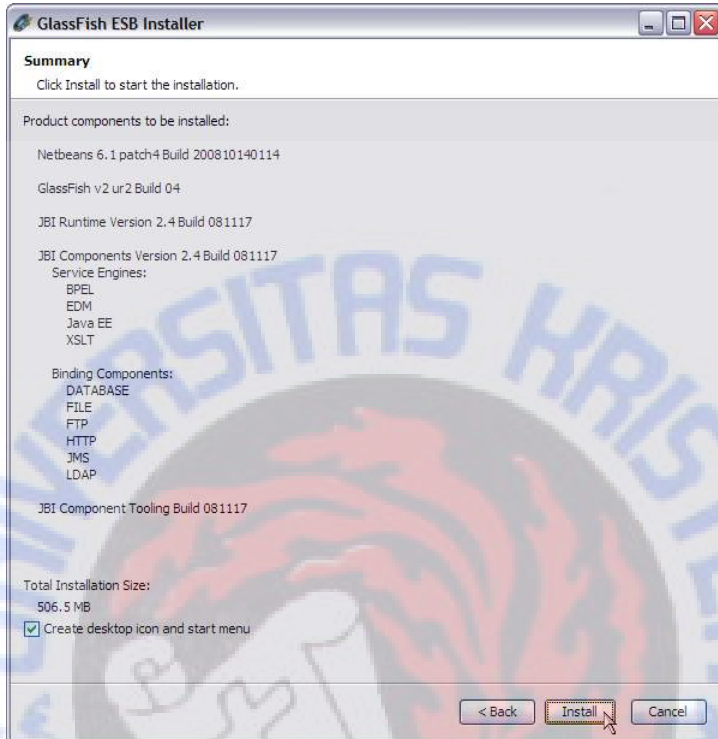
Prompt yang berkaitan dengan server aplikasi Glassfish akan tampak :



{Perhatikan apakah port 8080 sudah digunakan untuk server lain. Jika tidak ingin mengganti port 8080 maka terima saja default.}

Terima saja defaults, dan klik Next.

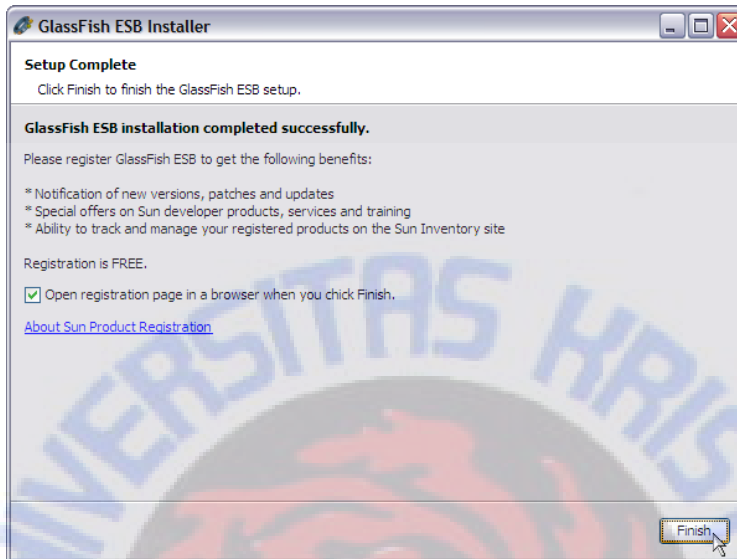
Kemudian ringkasan konfigurasi akan tampak.



{Ini merupakan ringkasan yang menunjukkan JBI service engines dan binding components yang terdistribusi dengan GlassFish ESB.

Jika menggunakan GlassFish ESB V 2.1, akan tampak pula Scheduler BC.

Perhatikan progress bar untuk melihat ketika instalasi telah lengkap. Saat instalasi selesai maka akan terlihat gambar berikut. {Progress bar instalasi akan memakan waktu 8-10 menit untuk instalasi yang lengkap.}



Klik *Finish* dan dengan demikian proses instalasi telah lengkap.

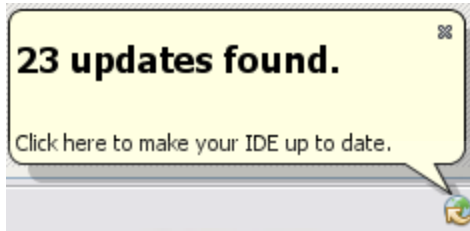
Halaman registrasi akan tampak di browser dan menanyakan registrasi Glassfish ESB.

{Jika melakukan register, maka ada keuntungan menjadi anggota komunitas dan menerima update penting tentang perkembangan teknologi ini.}

3.3 Memulai Glassfish ESB

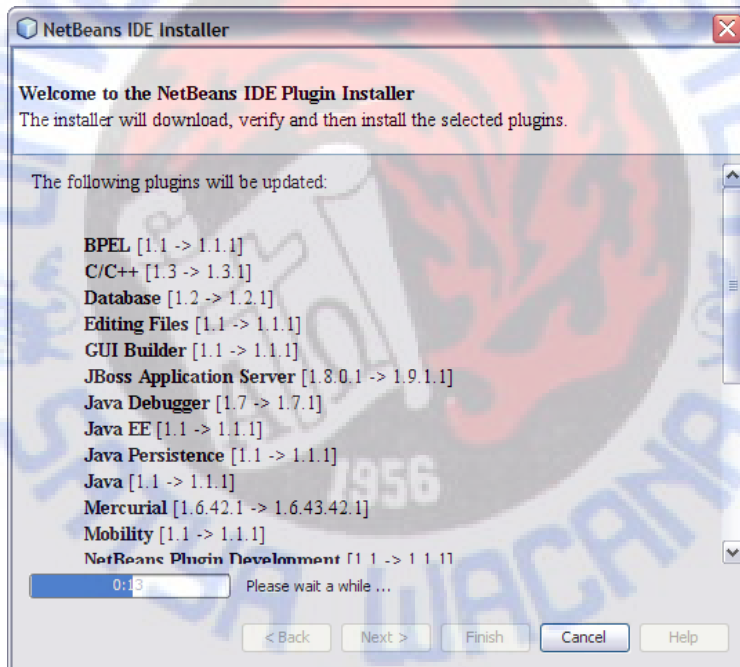
Start GlassFish ESB dengan menggunakan *Start / AllPrograms / GlassFish ESB* pada menu. {Start dapat dilakukan juga dengan menjalankan file *start_netbeans.bat* pada direktori *C:\GlassFishESB.*}

Setelah NetBeans start, dan jika ada koneksi internet, maka software akan melakukan pengecekan update yang tersedia. Akan tampak pesan seperti gambar berikut.



Ketika pesan tampak, klik pada pesan pop-up dan ikuti langkah untuk melakukan update Netbeans.

Lihat daftar update seperti yang terlihat yang dapat didownload.



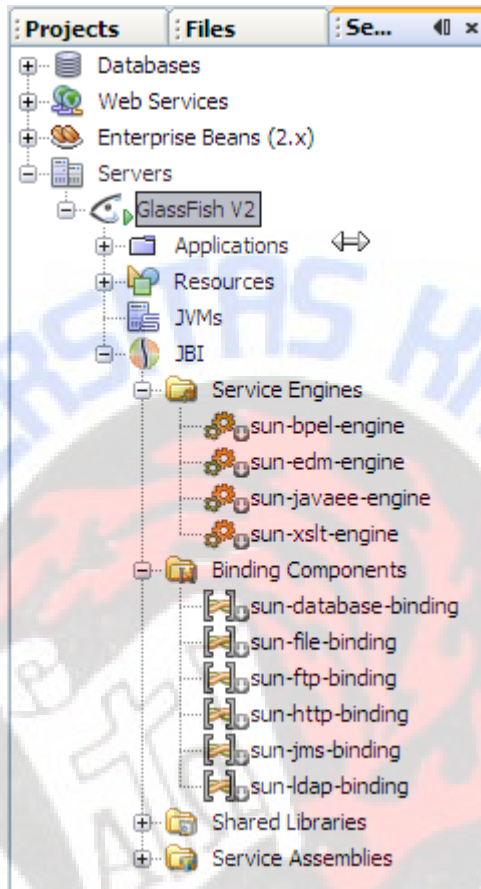
{Jika tidak terlihat daftar plug- ins, ini menunjukkan bahwa NetBeans sudah up-to-date.}

Jika pesan pop-up yang menunjukkan update tidak terlihat maka gunakan pilihan menu *Help / Check for Updates* untuk memulai pengecekan update.

Setelah instalasi selesai, maka lihat panel *Services* dan start aplikasi server Glassfish dengan melakukan klik kanan pada node *GlassFish V2* dibawah *Servers* dan pilih *Start* .

Setelah Glassfish start, cek JBI service engines dan binding components yang terpaket dengan GlassFish ESB dengan melakukan ekspansi hirarki GlassFish V2:





{SOA middleware yang ada pada GlassFish ESB didasarkan pada standard Java Business Integration 1.0 (JSR 208). JBI menetapkan bagaimana memperluas platform integrasi melalui penambahan komponen. JBI menetapkan model kontainer untuk integrasi. Ada dua jenis komponen pada JBI yaitu service engines dan binding components. Keduanya mini-containers.

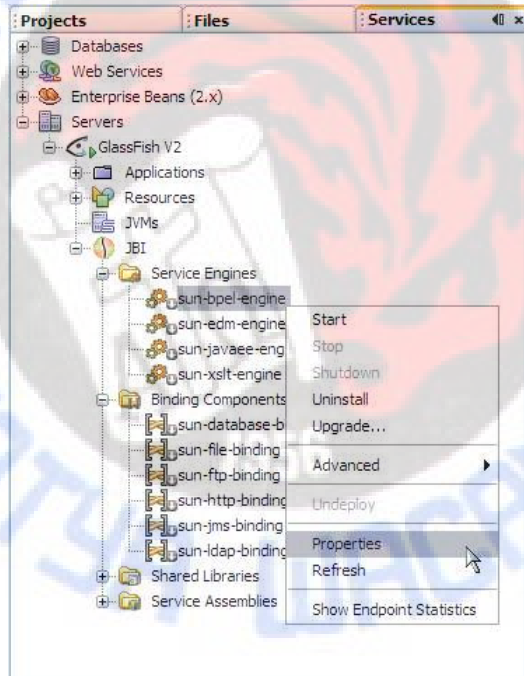
Service engines (SE) bertindak sebagai business logic dan binding components (BC) sebagai proxy dengan dunia luar.

Ikatan longgar (loose coupling) antara SE dan BC disediakan oleh sebuah in- memory bus yang disebut Normalized Message Router (NMR) }


Tampak bahwa ada empat services engines dan enam proxy komunikasi (binding components) pada runtime GlassFish ESB.

Sangat menarik mengetahui cara melihat status dan properti komponen JBI. Sebuah komponen mempunyai tiga status yaitu *shutdown*, *started* atau *stopped*.


Klik kanan pada node *sun-bpel-engine* dan pilih *Properties*:




*{JBI component mempunyai siklus hidup:
Shutdown – Tidak tersedia*

 sun-file-binding

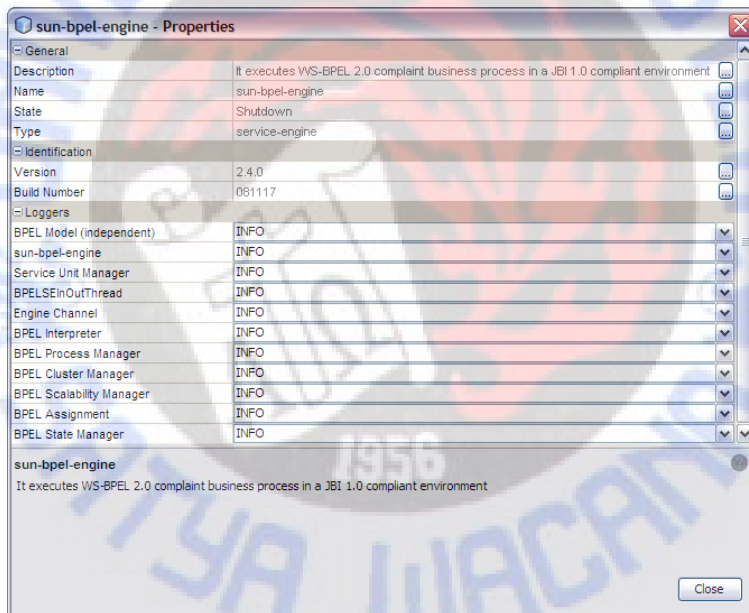
Started – Siap menerima pesan:

 sun-file-binding

Stopped – Sementara tidak tersedia, tetapi dapat start lagi dengan perintah start secara cepat:

 sun-file-binding }

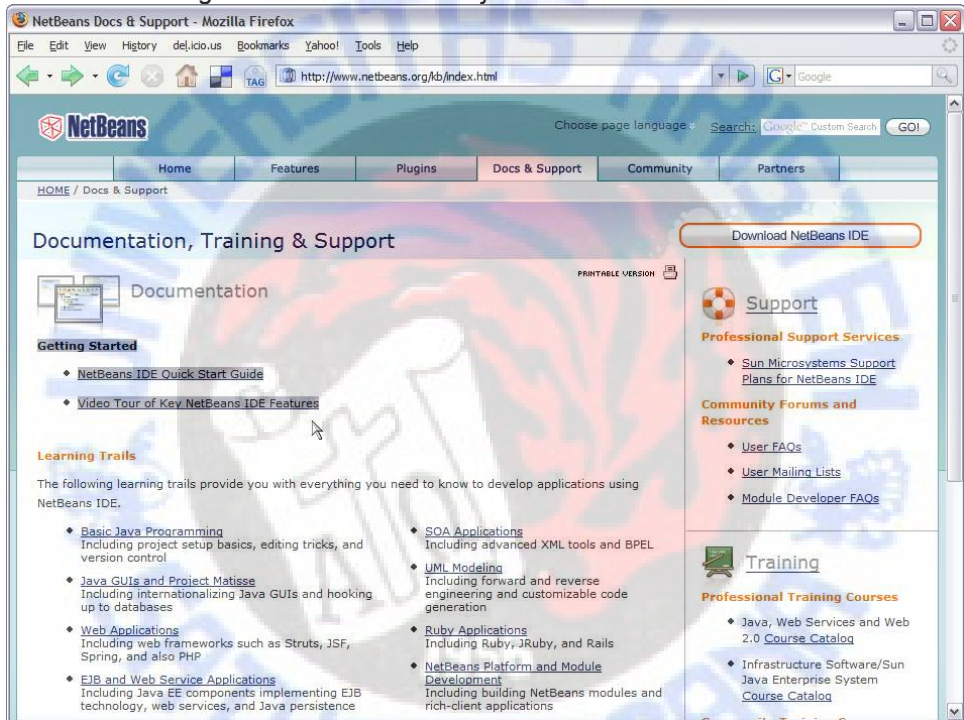
Lihat properties pada komponennya. Selama troubleshooting, akan membantu mengetahui soal property ini sehingga debugging dapat diketahui lebih rinci.



{Perhatikan bahwa dapat dikondisikan level logging JBI component. Nilai default, adalah level INFO.}

Setelah GlassFish ESB diinstall dengan NetBeans IDE, jika mempunyai akses web, maka dapat dilakukan pendalaman dengan melihat link tutorial “Getting Started” pada : <http://www.netbeans.org/kb/index.html>

{Jika GlassFish ESB belum start, maka dapat melalui Windows desktop menggunakan opsi menu Start / All Programs / GlassFish ESB.}



Link *Getting Started* akantampak diatas. Setelah memahami lingkungan lingkungan Netbeans maka dapat pindah dengan memulai proyek.