# COOL:Joe

**C**OOL:Joe is an EJB development environment integrated with architecture design and component modeling capabilities. Java development teams are able to architect, create, deploy and manage reliable and scalable enterprise applications using EJBs. Using COOL:Joe's built-in smart features, developers can automatically generate EJBs from component specifications, test and debug business logic, and deploy to an application server. Developing distributed systems is effortless because COOL:Joe generates the EJB infrastructure code and frees the developers to concentrate on developing business logic.

## Who should use COOL:Joe

- IT architects who need to design large-scale distributed eBusiness systems.
- Development teams who want to create and deploy server-side components with Enterprise JavaBeans (EJBs) technology.
- Java specialists that want to be insulated from the underlying Java technology while taking advantage of EJBs.

## Features of COOL:Joe

**Component Management** Manage components as corporate assets, providing a foundation for re-use.

**Component Modeling** Visualize a component architecture and collaborations among component interfaces.

**Smart Options** Provide ways to get the most out of your developer resources.

- **Smart Expansions** Provide common code fragments to be built and shared across the development team.
- **Smart Macros** Extend the COOL:Joe development environment to automate processes for your team.
- **Smart Methods** Avoid coding errors by leveraging the power of an object repository.

**Editor** Quickly create and edit business logic and related files with advanced editing capabilities.

**Object repository** All visualization and implementation objects created and defined once and used as needed.

**UML Class Diagram** Implementation classes modeled using auto-layout.

**Java Standards** Full support for Sun Microsystems' Java 2.

**Advanced Wizards** Automate the majority of the development effort so generalists and specialists can generate server side EJB components.

- **Specification to Implementation Wizard** Generate all the necessary classes to support implementing your business components.
- **Component Wizard** Automatically converts Java classes to EJB components without specification.
- **EJB Generation Wizard** Apply the Enterprise JavaBeans framework to your business logic.
- **Persistence Generation Wizard** Automatically support persistence by generating the object to relational mappings and data management methods.
- **Test Harness Wizard** Automatically generates user interfaces for testing.
- **Build Wizard** Compiles component classes and creates JAR files.

**Deployment Wizard** Automatic deployment to leading EJB application server environments.

**Interactive Debugger** Set breakpoints and step through Java business logic.

*COOL:Joe links component specifications to an implementation and deployment environment that ensures quality results.*

*COOL:Joe supports system architects in designing applications, which is critical for meeting the needs of eBusiness solutions.*

*COOL:Joe provides automatic test generation facilities, which speeds up deployment of business systems.*

*COOL:Joe illustrates component inventories and interfaces, which are necessary for building distributed systems.*

*COOL:Joe employs repeatable processes, which promote better use of scarce development resources.*

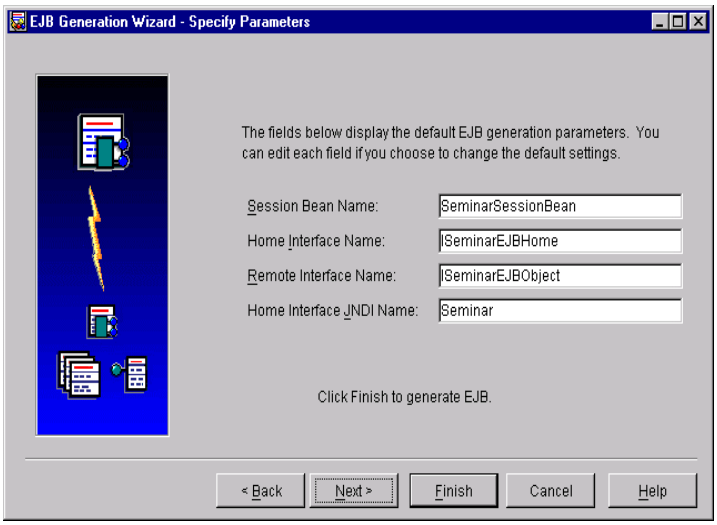


*Figure 1. Specify Parameters Dialog of the EJB Generation Wizard*

The EJB Generation Wizard will create a session bean for a component. Upon executing the wizard, the component implementation is wrapped with the EJB framework without requiring knowledge of the complexities of EJB technology. However, the EJB framework is stored in the COOL:Joe model repository, allowing the developer to extend it for unique environmental requirements. Once the EJB is generated and built, it can be deployed to an EJB application server using the Deployment Wizard. The Deployment Wizard will automatically create the EJB server container and enable communications with the server machine and database. The Test Harness Wizard can be used to fully test the EJB either locally or remotely with source level tracing, providing a complete test environment.

## Workflow in COOL:Joe

1. Model business processes and relationships to build a component architecture, which includes the individual component specifications, needed to meet new systems requirements.
2. Use the Specification to Implementation Wizard to transform a component specification into Java classes.
3. Generate a set of persistence support classes to access a relational database with the Persistence Generation Wizard. Use the DDL Generation Wizard to generate the database definition language.
4. Use the Editor to specify business logic and use the Class Diagram to model class structures.
5. Use the EJB Generation Wizard to generate the code to implement the component as an Enterprise JavaBean.
6. Compile component classes and create an executable JAR file with the Build Wizard.
7. Use the Deployment Wizard to deploy the EJB JAR file to the Enterprise JavaBean server.
8. Use the Test Harness Wizard to generate all the code necessary to test the component. Two interfaces may be generated for testing – a Java desktop application and HTML servlets.
9. Test the EJB.
10. EJB implemented successfully.

## Open Integration

COOL:Joe includes Component Manager 1.2, which provides support for multiple component models. For example, component specifications implemented in COOL:Gen can also be used in COOL:Joe to generate native EJBs. In addition, COOL:Gen's Java Proxy can be used to link EJBs created in COOL:Joe with COOL:Gen server transactions. Any Java source can be imported in to COOL:Joe, providing open integration with Java tools.

**System Requirements for COOL:Joe:**

Pentium II or III 300 MHz microprocessor or higher recommended • CD-ROM drive • minimum 128 MB RAM; recommended 256 MB RAM • SVGA monitor resolution or better • 100 MB storage. Operating System: MS Windows NT 4.0 with Service Pack 4 with Year 2000 updates. MS Network services • Mouse • MS Word 97 or later. Relational database such as MS SQL Server 7.0, DB2/2 5.0, or Oracle 8i /8.1.5. or higher with JDBC drivers. **Provided**: Enterprise JavaBean specification classes 1.0 • JavaServer Web Development Kit 1.0 (JSWDK) • ObjectStore 5.1 and ObjectStore Java Interface 3.0 • Java 2 SDK, Standard Edition (includes Java 2 Runtime environment) 1.2.2 • Java 2 SDK documentation 1.2.2 • Java Naming and Directory Interface 1.1.2 (JNDI). **Other**: Web browser • Java Activator Plug-ins • EJB Application Server (WebLogic 4.0.3 or 4.5.1 on Windows NT/2000).

For more information on Sterling Software and the COOL products:

www.sterling.com/cooljoe
1-877-SSW-COOL
+44 (0)1932 587-000

STERLING SOFTWARE