

WebSession Java Development Kit

WebSession Development Kit is a Java-class library that provides easy and convenient access to 390-class host mainframe applications. Use the API to write applications, applets or servlets that access your mainframe legacy applications and databases. Your Java program can create sessions and do all of the things a human operator can do.

WebSession Java Beans are also available with the WebSession Java Development Kit. Java Beans are reusable components that provide WebSession Java Development Kit's power and easy of use in any visual builder tool or IDE. For more information on WebSession Java Beans, please see *WebSession 3270 Beans for WebSession Development Kit*.

Your Java application or applet acts as a programmed operator. Repetitive or complex tasks are easily automated because the program can run without human intervention. Any well-defined sequence of 3270 session activities is a candidate for a WebSession application, such as:

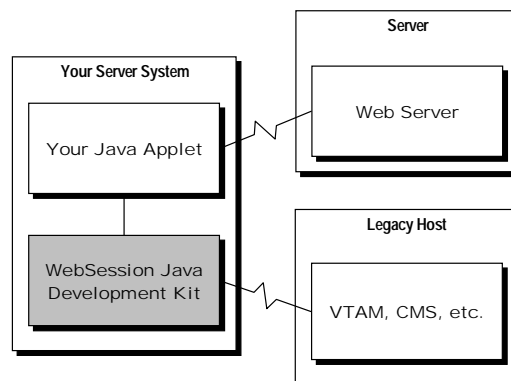
- query a mainframe database and copy selected data;
- monitor mainframe performance or activity
- automate a series of file uploads or downloads
- copy data from one mainframe session or application to another
- transfer files between mainframes using download and upload
- and much more

Your Java program creates host sessions by invoking the API session object Session3270. This object uses the TN3270E protocol to communicate with your mainframe host. You must implement the SessionListener

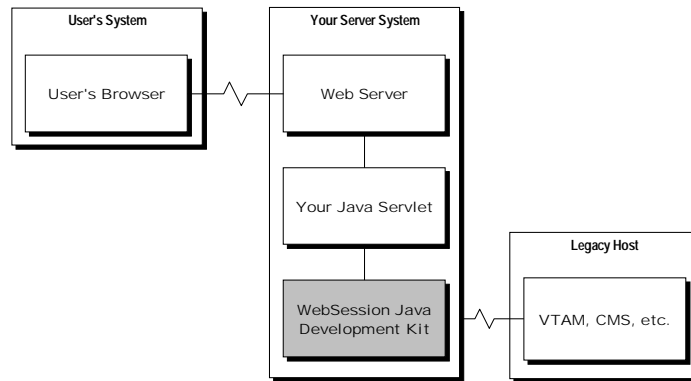
interface in order to be notified of asynchronous session events and errors. The SessionAdapter class is provided for your convenience in implementing the SessionListener interface.

Your Java program has complete control of each host session by calling Session3270 methods. Sample programs are supplied to illustrate the use of these methods.

WebSession programs can start multiple host sessions. Each



Using WebSession Java Development Kit
in a Java applet



Using WebSession Java Development Kit to write a Java servlet

instance of the Session3270 object can be connected to a different host.

Features

WebSession programs can make sessions visible or invisible to an operator. Although intended as an aid to program debugging, the visible session window has interactive terminal functionality. Therefore your program can make a session visible dynamically when operator assistance is required and invisible during normal operation.

When a session is visible, your WebSession program can dynamically allow and disable operator interaction.

WebSession programs can start a file transfer and then do other processing while the transfer is in progress.

Your WebSession program can create a graphical user interface using both the Java AWT classes and/or the Java Foundation Classes. This allows your program to create portable, modern, graphical “front end” interfaces for legacy host applications.

Requirements

The WebSession Development Kit consists of a number of Java classes. It is delivered in a single file, `javaapi.jar`, which requires a Java Runtime Environment, Version 1.1 or later.

The WebSession Development Kit comes with standard Java documentation as well as many sample Java applications. Development is quick and easy, with support available from I/O Concepts support.

Sample Applications	Deliverables
<p>Sample WebSession programs are located in a samples directory, and include:</p> <ul style="list-style-type: none"> • Ispf Sample demonstrates how to write a state machine that can automatically logon to a mainframe TSO session for a user. • File Transfer Sample demonstrates how to send a file to a mainframe host and how to fetch a file from a mainframe host. • Servlet HTML Sample is a simple servlet that connects to a mainframe host using our WebSession API, and performs 3270-to-HTML conversion. 	<p>The following files will be installed in your WebSession installation directory:</p> <ul style="list-style-type: none"> • <code>release.txt</code> Release notes • <code>docs</code> A directory containing javadoc documentation <p>The following files should be installed in the <code>javaapi</code> subdirectory:</p> <ul style="list-style-type: none"> • <code>api*.gif</code> Pictures used in the <code>apiReadme.html</code> page • <code>javaapi.jar</code> A JAR file containing the WebSession Java API • <code>samples</code> A directory containing WebSession Java API sample programs