IBM

The CICS Transaction Gateway: Web and Java access to CICS

Chris Smith
smithch@uk.ibm.com

Fort Mitchell, 12 - 16 April 2002

Trademarks

- The following terms are trademarks of International Business Machines Corporation in the United States and/or other countries:

AIX	MVS/ESA	TXSeries	WebSphere
CICS	OS/2	VSE/ESA	z/OS
CICS/VSE	OS/390	VTAM	
DB2	S/390	VisualAge	

- Java, J2EE, Solaris are trademarks of Sun Microsystems, Inc
- Windows, Windows 95, Windows 98, Windows 2000, and Windows NT are trademarks of Microsoft Corporation, Inc
- Other company, product, and service names may be trademarks or service marks of others

Agenda

- Introduction
- Structure
- Terminal Servlet
- Network Protocols
- Connectivity to CICS
- Security Considerations
- Application Programming Interfaces
- Connector Architecture Support
- Further Information
- Summary

Agenda

- **Introduction**
- Structure
- Terminal Servlet
- Network Protocols
- Connectivity to CICS
- Security Considerations
- Application Programming Interfaces
- Connector Architecture Support
- Further Infomation
- Summary

The CICS Transaction Gateway

- Provides an interface to CICS from Java and the Web....
  - from a Web Browser
  - from any Java execution environment
    - Applets
    - Servlets
    - Enterprise JavaBeans
    - Other Java Applications
- Allows Java programs to....
  - invoke CICS application programs
  - drive 3270 based CICS transactions
- Is a strategic IBM e-business Connector
  - CICS connector for WebSphere

The CICS Transaction Gateway

- Comprises
  - Java Gateway Daemon
  - Client Daemon
  - Java Class Library
  - Terminal Servlet
  - Configuration Tool
- Runs on several platforms
  - Windows NT, Windows 2000
  - AIX, Solaris, HP-UX
  - Linux for zSeries, Linux for S/390
  - OS/390, z/OS
- Supports multiple concurrent users and CICS connections
- Latest Version is V4.0.2

The CICS Transaction Gateway

CICS Transaction Gateway V3

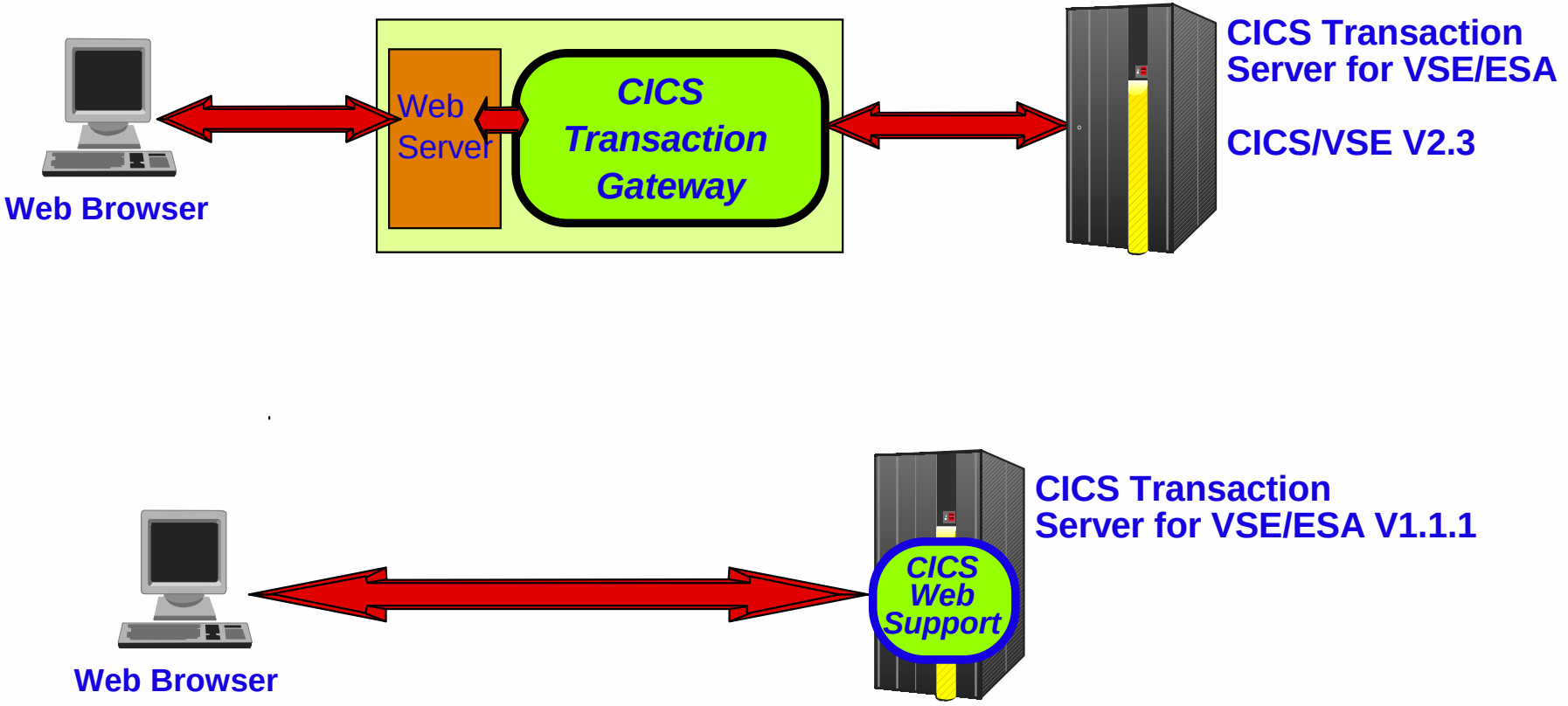
- Was delivered and licensed with
  - CICS Transaction Servers
  - TXSeries
  - WebSphere Enterprise Edition
  - VisualAge for Java Professional and Enterprise Editions
- Now withdrawn
  - Fee-based service is available until 31 January 2003

The CICS Transaction Gateway....

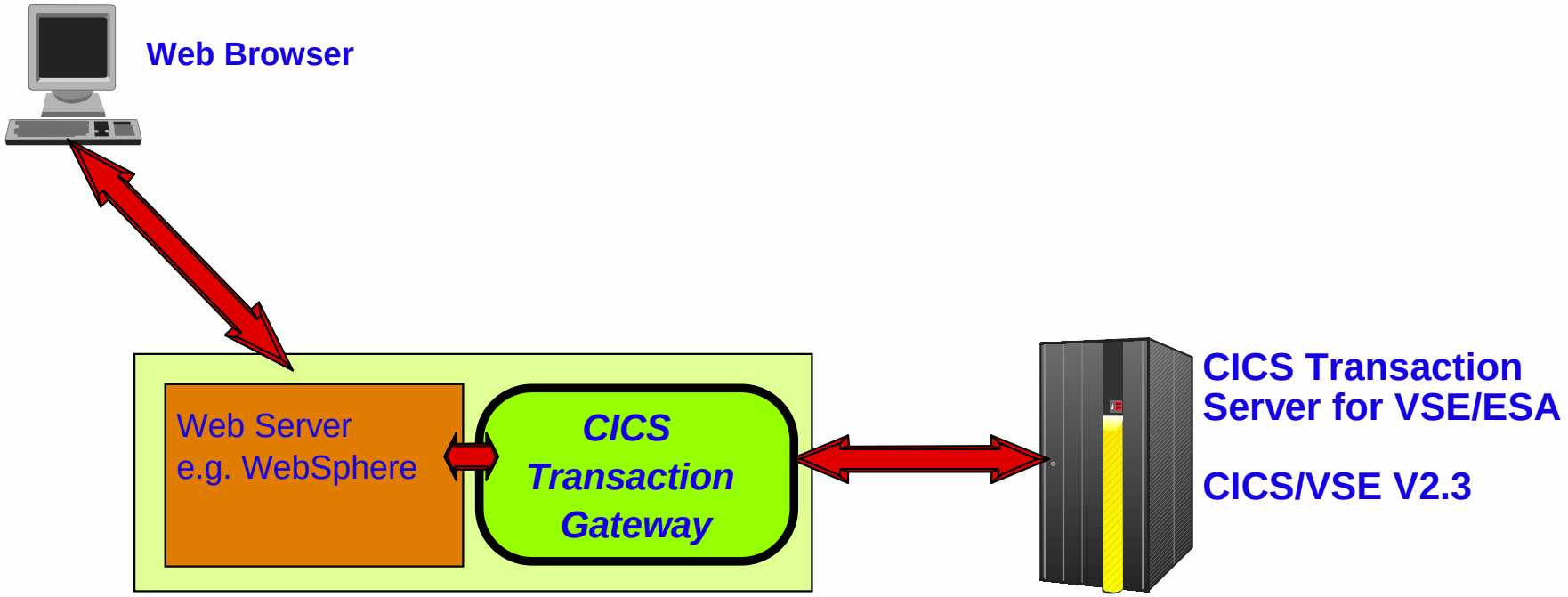
CICS Transaction Gateway V4

- Separately ordered and priced product
- New functions
  - Additional platform support
    - Windows 2000
    - HP-UX support
    - Linux for zSeries and S/390
  - J2EE Connectors
  - Support for JDK V1.3
  - TCP62 enhancements
  - API enhancements
  - New sample programs

Relationship to CICS Web Support



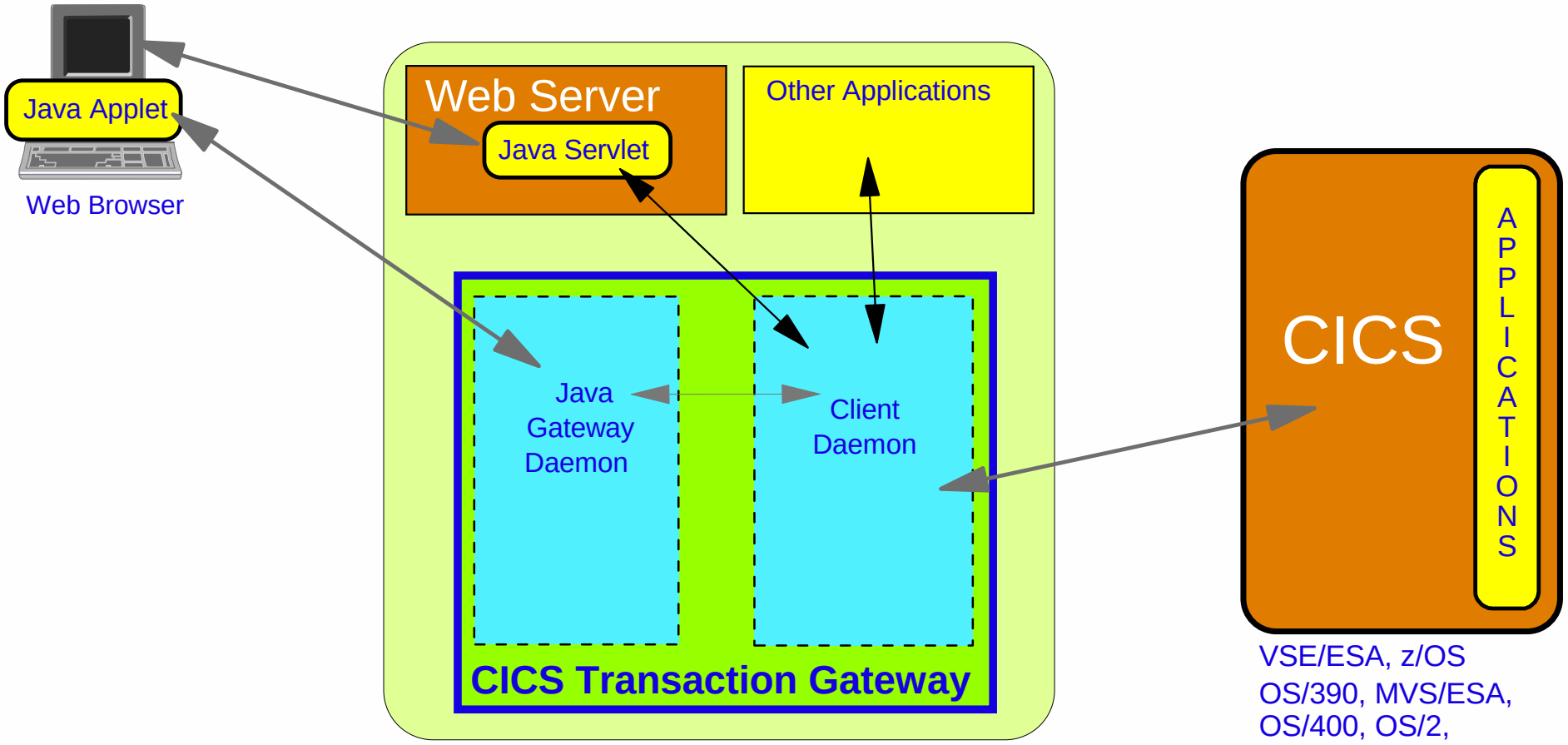
The CICS Transaction Gateway



Agenda

- Introduction
- **Structure**
- Terminal Servlet
- Network Protocols
- Connectivity to CICS
- Security Considerations
- Application Programming Interfaces
- Connector Architecture Support
- Further Information
- Summary

Structure



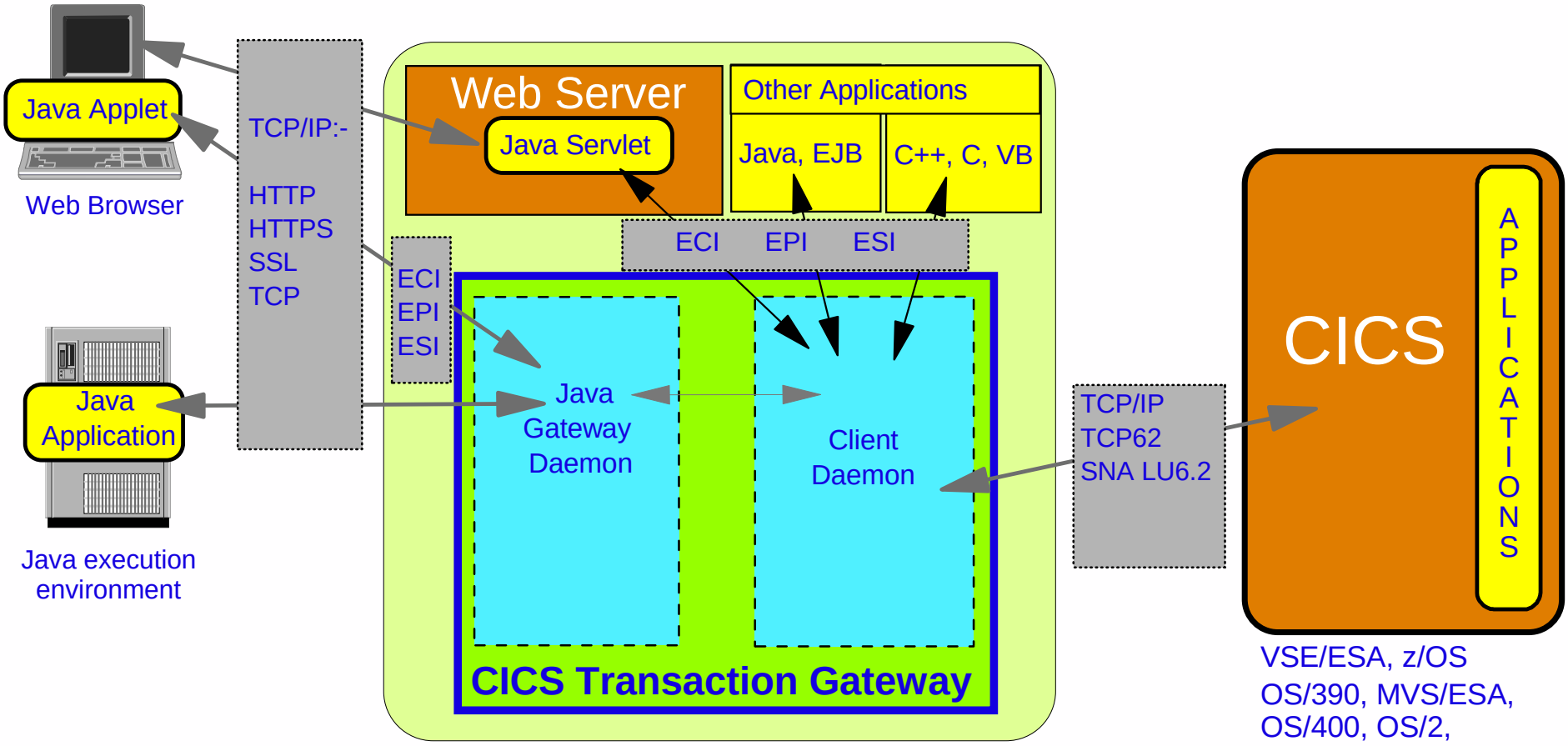
Windows NT, Windows 2000, AIX, Solaris, HP-UX, z/OS, OS/390, Linux for zSeries & S/390

VSE/ESA, z/OS OS/390, MVS/ESA, OS/400, OS/2, AIX, Solaris, HP-UX, Windows NT, Windows 2000

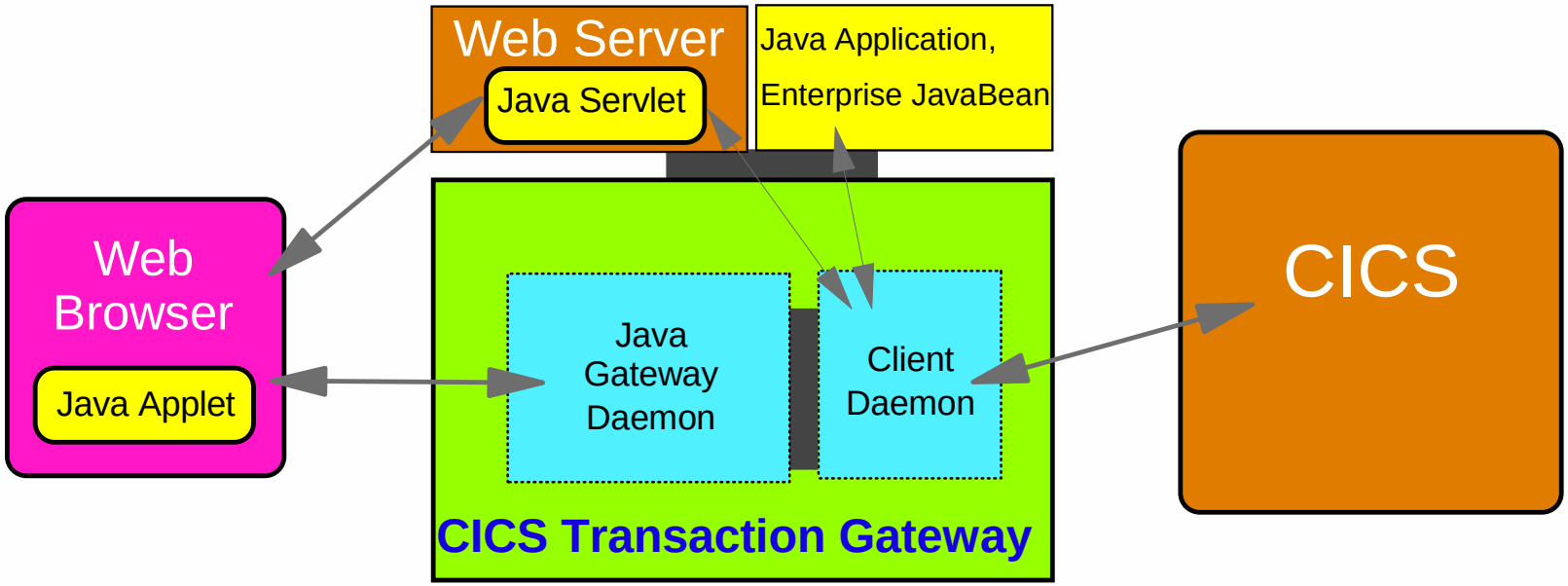
Structure

- The Java Gateway Daemon.....
  - handles connectivity to the Java client programs
  - interfaces to the Client Daemon
  - is a Java application
- The Client Daemon.....
  - is the CICS Universal Client
    - CICS Universal Client is a separate product
    - Integrated within the CICS Transaction Gateway
  - provides access to CICS systems
    - Base API's
    - Connectivity

Structure

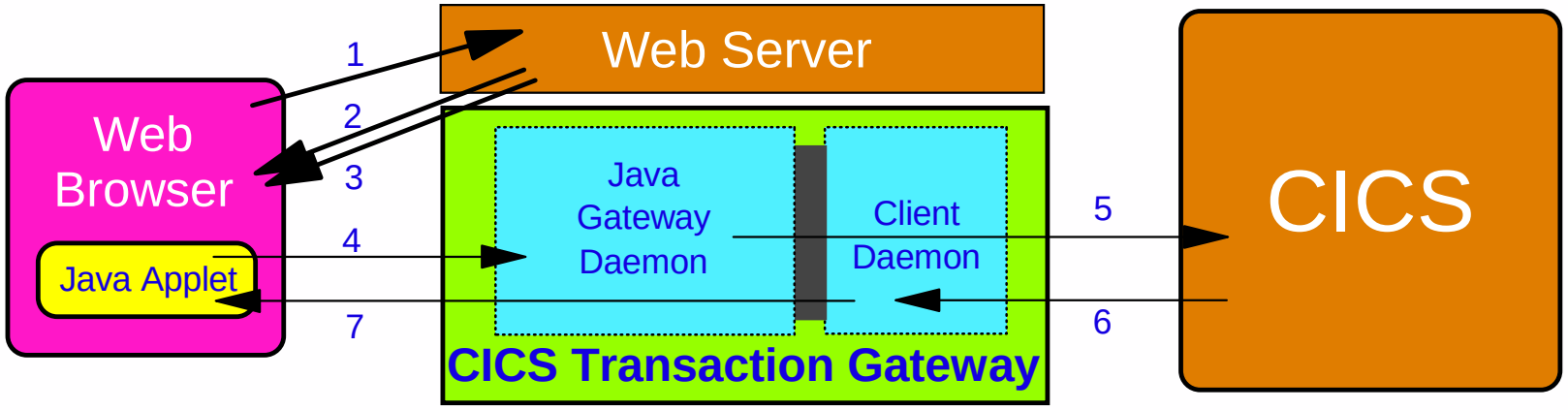


Structure



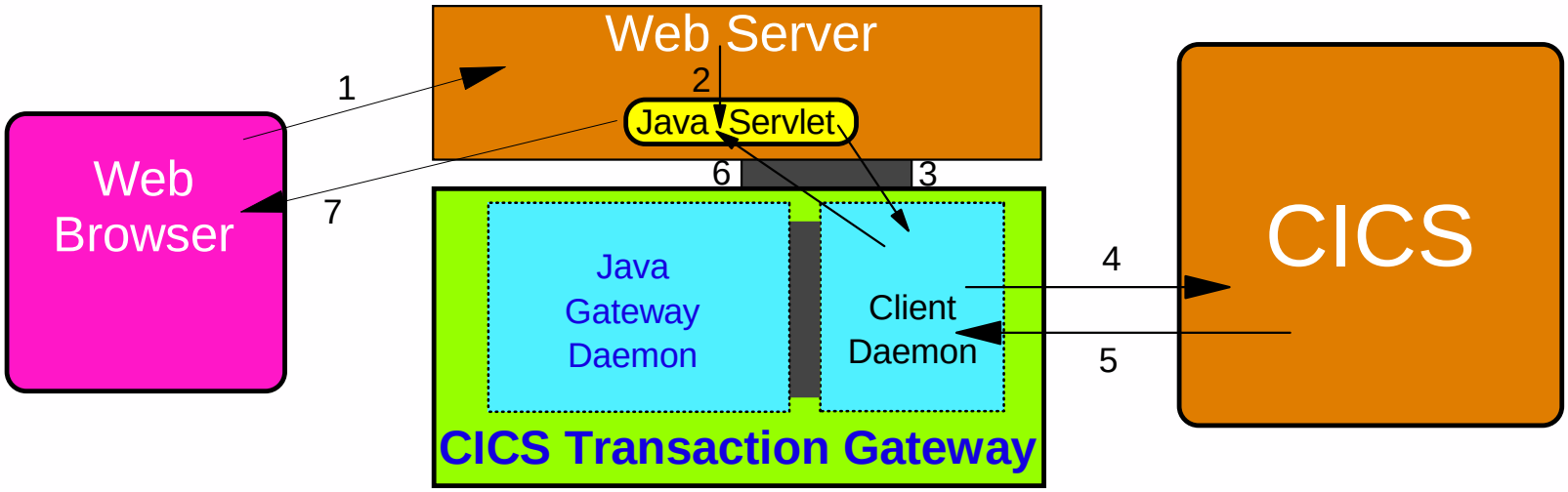
- Applets are Java applications that execute in Web browsers
- Servlets are Java applications that execute in Web servers
- Enterprise JavaBeans are Java applications that execute in Enterprise Java Servers

Applet flows



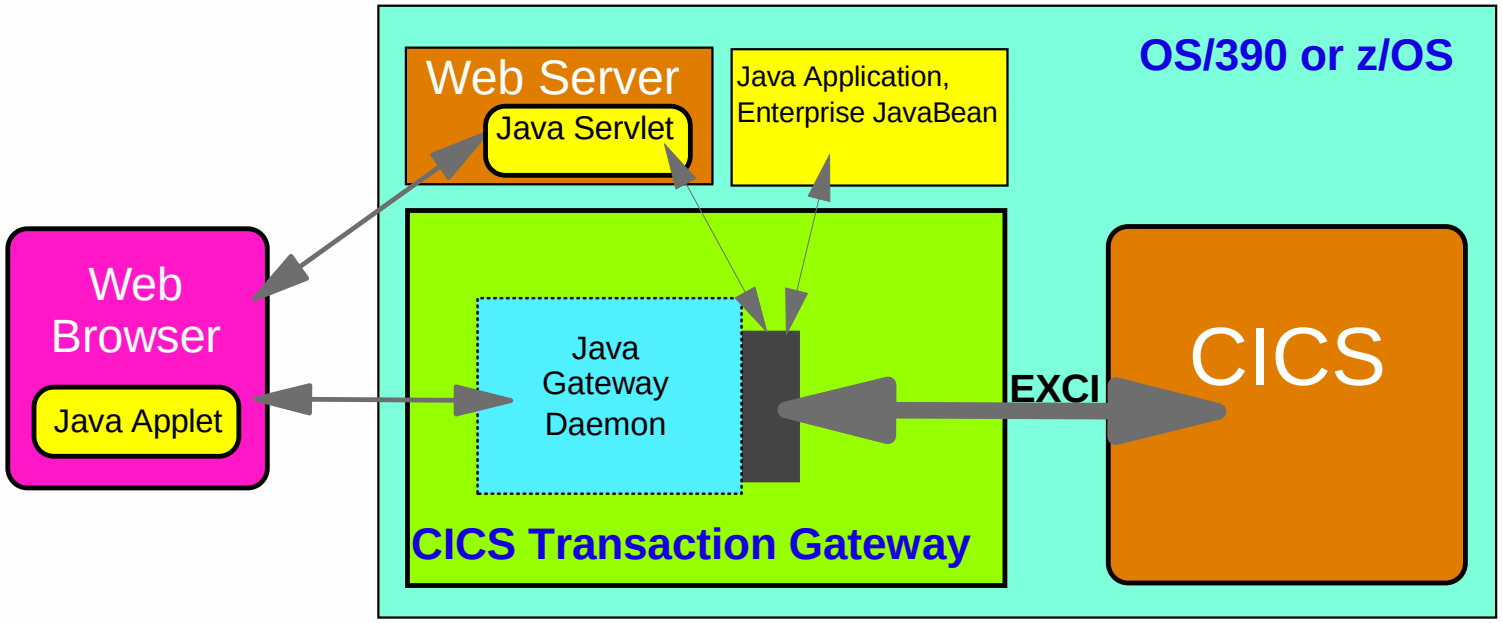
1. Web browser requests HTML page from the Web-server
2. Web server returns HTML page which identifes applet
3. Web browser downloads applet
4. Applet creates a CICS request and passes it to the Gateway
5. Gateway calls CICS Universal Client to pass request to CICS
6. CICS processes the request and returns result to CICS Client
7. Gateway gets result from CICS Client and provides to applet

Servlet flows



1. Web browser requests an HTML page from the Web-server
2. Web server loads servlet identified in HTML page
3. Servlet creates a CICS request and passes to CICS Client
4. CICS Universal Client passes the request to CICS
5. CICS processes request and returns result to CICS Client
6. Servlet receives result from the CICS Client
7. Servlet formats HTML page and web server sends to browser

Structure on OS/390



- Communication with CICS via EXCI
- ECI only - no support for EPI or ESI

Agenda

- Introduction
- Structure
- **Terminal Servlet**
- Network Protocols
- Connectivity to CICS
- Security Considerations
- Application Programming Interfaces
- Connector Architecture Support
- Further Information
- Summary

The Terminal Servlet

- Provides access to CICS ***transactions*** from Web Browsers
- Supplied as part of the CICS Transaction Gateway
- Not supported if CTG running on OS/390 or Linux for S/390
- The Terminal Servlet can....
  - Behave like a simple terminal emulator
  - Substitute data from CICS into HTML template files
  - Display CICS screen data in server-side includes
  - Map specific CICS screens to HTML pages
- Can be invoked in three ways....
  - By URL
  - With an HTML FORM
  - With a server-side include

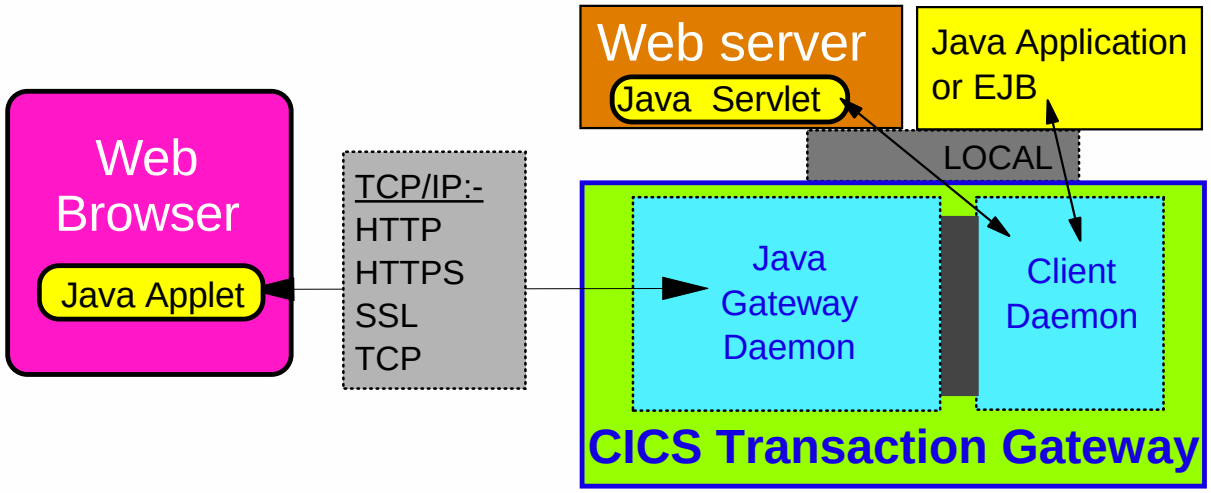
The Terminal Servlet

- Invoking the Terminal Servlet with a URL:
  - http://webserver/servlet/TerminalServlet?request=send&transaction=CECI
- Invoking the Terminal Servlet with an HTML FORM:
  - <FORM METHOD="GET" ACTION="/servlet/TerminalServlet">
    <INPUT TYPE="HIDDEN" NAME="REQUEST" VALUE="SEND">
    <INPUT TYPE="HIDDEN" NAME="TRANSACTION" VALUE="CECI"
    .....Text entry tags, buttons, etc.....
    </FORM>
- Invoking the Terminal Servlet with a server-side include:
  - <SERVLET NAME="TerminalServlet">
    <PARAM NAME="request" VALUE="send">
    <PARAM NAME="transaction" VALUE="CECI">

Agenda

- Introduction
- Structure
- Terminal Servlet
- **Network Protocols**
- Connectivity to CICS
- Security Considerations
- Application Programming Interfaces
- Connector Architecture Support
- Further Information
- Summary

Network Protocols

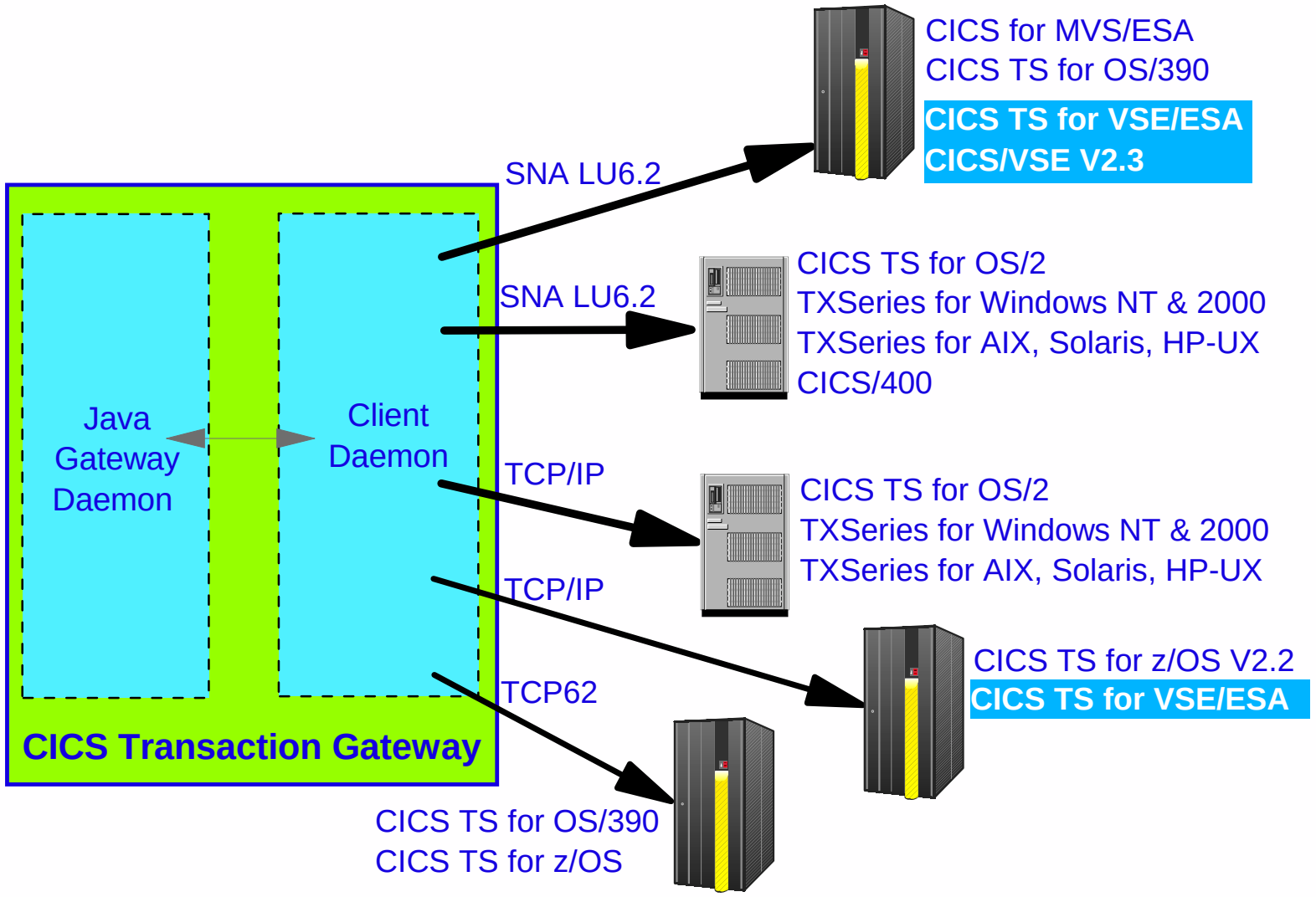


- tcp - private persistent connection protocol
- http - standard protocol used for the Web
- ssl - private persistent secure connection protocol
- https - secure protocol used for the web
- local - private protocol used on Gateway machine

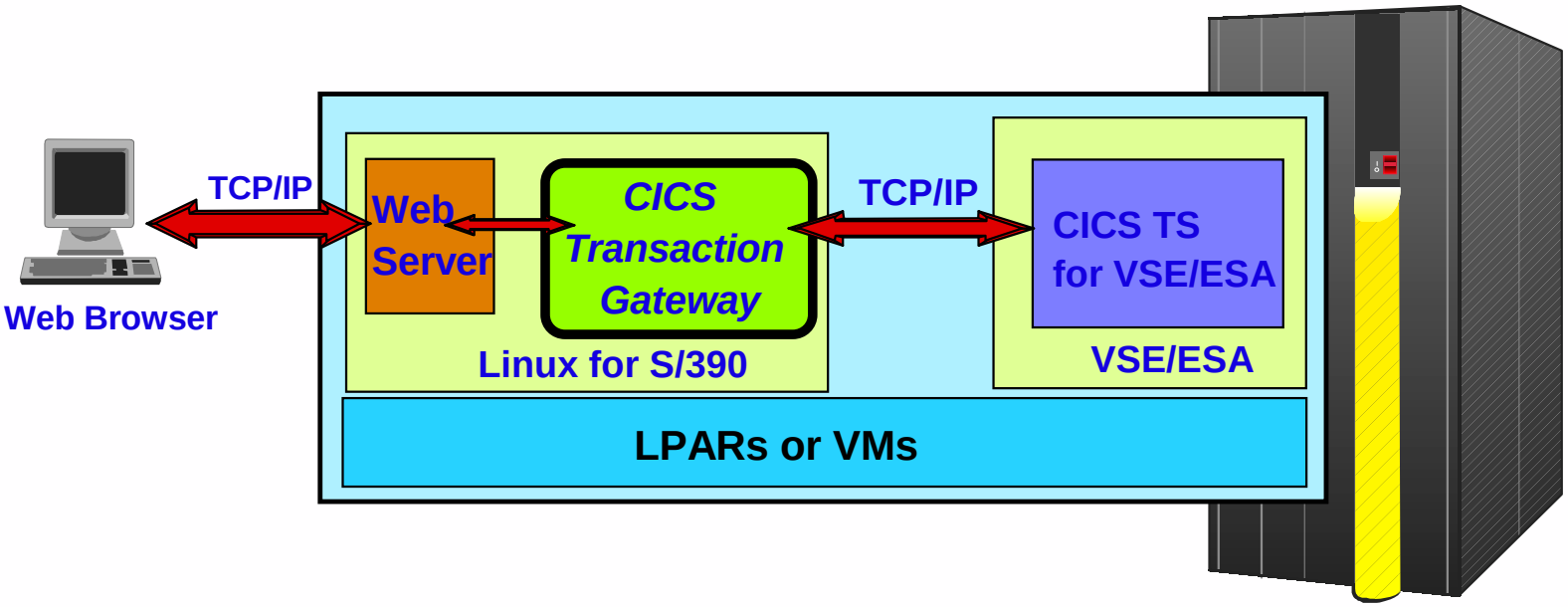
Agenda

- Introduction
- Structure
- Terminal Servlet
- Network Protocols
- **Connectivity to CICS**
- Security Considerations
- Application Programming Interfaces
- Connector Architecture Support
- Further Information
- Summary

Connectivity to CICS



Connectivity from Linux for zSeries or S/390



- CICS Transaction Gateway V4 only
- ECI support only
  - No support for EPI or ESI

Agenda

- Introduction
- Structure
- Terminal Servlet
- Network Protocols
- Connectivity to CICS
- **Security Considerations**
- Application Programming Interfaces
- Connector Architecture Support
- Further Information
- Summary

Security Considerations

- Between end-user or client application and the Gateway....
  - Via Secure Sockets Layer (SSL)
    - Support includes User Exits
- Between the Gateway and CICS on S/390....
  - LU6.2 security
    - Link
    - Session (Bind-time)
    - User (Conversation)

Agenda

- Introduction
- Structure
- Terminal Servlet
- Network Protocols
- Connectivity to CICS
- Security Considerations
- **Application Programming Interfaces**
- Connector Architecture Support
- Further Information
- Summary

Application Programming Interfaces

- Three API's
  - External Call Interface
  - External Presentation Interface
  - External Security Interface
- Java is the primary language
  - Applets
  - Servlets
  - Enterprise JavaBeans
  - Applications

NB: The other language bindings are also available on the system on which the Gateway is running

- C++, C, Visual Basic

The External Call Interface

- Usually referred to as the ***ECI***
- Allows invocation of COMMAREA-based ***applications***
- CICS application invoked via
  - Program name
  - Userid and password
  - COMMAREA
- Like a CICS Distributed Program Link
- Calls may be extended to create one logical transaction
- Calls may be synchronous or asynchronous

The External Presentation Interface

- Usually referred to as the ***EPI***
- Provides access to CICS 3270 ***transactions***
- Acts as a logical terminal
- Used to drive existing CICS 3270 applications
- No change to CICS application

The External Security Interface

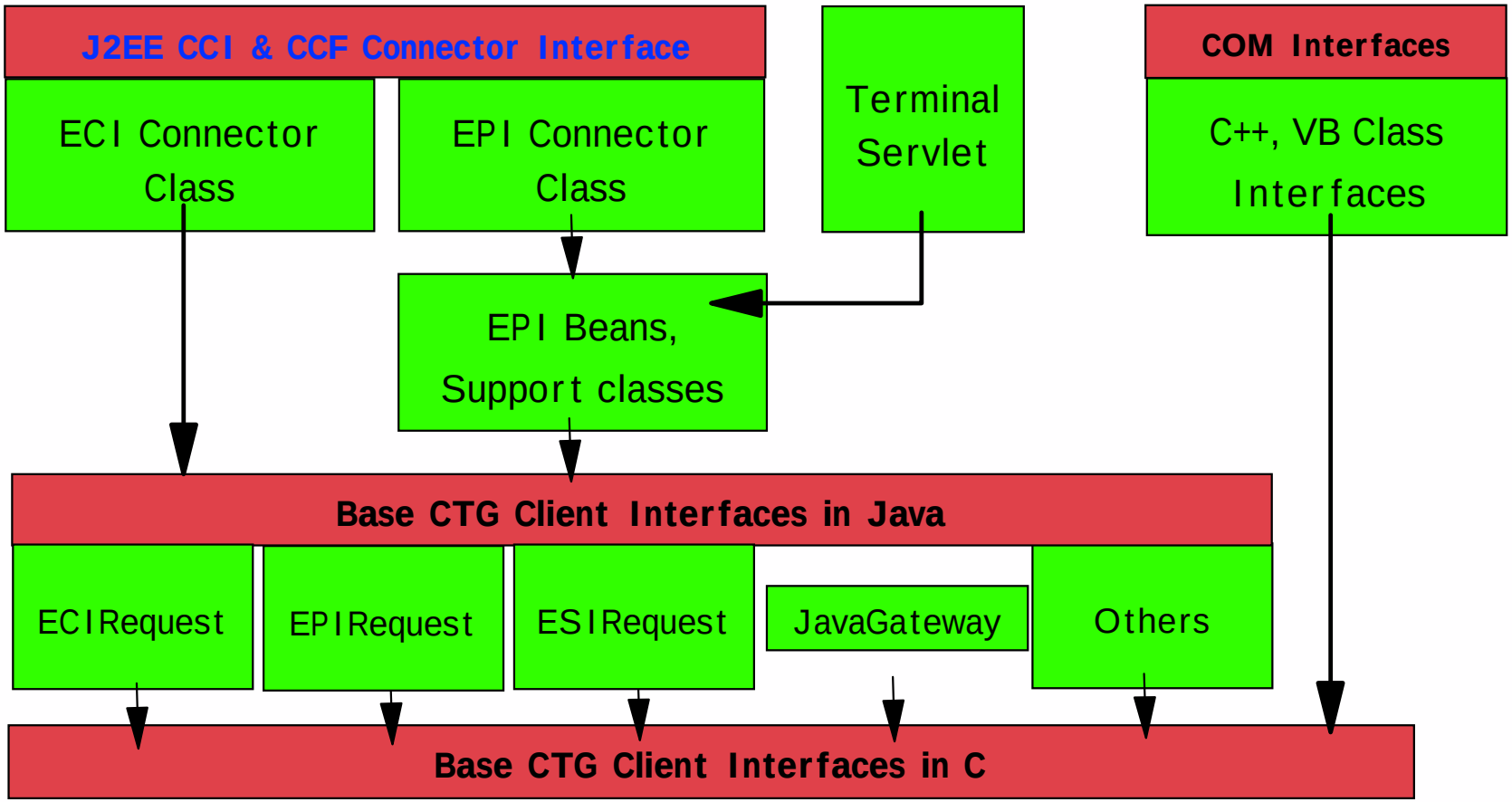
- Usually referred to as the ***ESI***
- Enables use of APPC ***Password Expiry Management (PEM)***
- Passwords can be verified or changed
- Provides audit trail information
- Requires an External Security Manager on S/390

The Java API....some basic concepts

- A ***Class*** is a collection of methods, data and interfaces common to all objects of a certain type
- A ***Method*** is the object-oriented term for a function
- An ***Object*** is created by instantiating the relevant class
  - Behaviour implemented with methods
  - State maintained in variables
- ***JavaBeans*** are self-contained re-usable Java components
  - Require no programming
  - Use any JavaBean enabled visual application builder
    - e.g. IBM VisualAge for Java, Sun BDK BeanBox

The Java API

- Several layers of Java API supported



Base Java API's

- ***JavaGateway*** object
  - Represents connection to the CICS Transaction Gateway
  - Has various properties....
    - URL
    - Network address
    - Security classes to be used
  - Core method is ***flow***
    - Sends requests to the Gateway
    - Synchronous or asynchronous

Base Java API's....

- ***ECIRequest*** object
  - Encapsulates all types of ECI request
- ***EPIRequest*** object
  - Encapsulates all types of EPI request
- ***ESIRequest*** object
  - Encapsulates all types of ESI request
- ***CicsCpRequest*** object
  - Queries code page in use
- ***Callbackable*** interface
  - Used with asynchronous calls

Java EPI Support Classes

- Hides programmer from 3270 datastreams
- Based on C++ EPI classes in CICS Universal Client
- Terminal class handles all interactions with CICS
- Terminal has associated ***Screen*** instance....
  - Contains a number of ***Fields***
    - accessed by index or screen position
- for BMS screens a ***Map*** class can be generated from BMS source and then fields accessed by name
- BMS Map classes created using supplied utility

Example ECI Java code

```
import com.ibm.ctg.client.*;                                    //  ctg classes
public class ECISamp
{
//  Invoke program using:   java ECISamp <Gateway_URL> <CICS_Server><CICS_Prog><COMMAREA_size>
   public static void main (String [ ] args)
   {
        ECIRequest ecireq = null;                               // initialise ECI request object
        int CommareaSize =  integer.parseInt(args[3 ])          // get commarea size as an integer
        byte [ ] Commarea = new  byte  [CommareaSize]           // create byte array for Commarea
       JavaGateway jgate = new JavaGateway();                   // create a  JavaGateway object
        jgate.setURL(args[0]);                                  // set URL of Gateway
        jgate.open();                                           //open connection to Gateway
                                                                // set parameters on ECI request object
        ecireq = new ECIRequest(ECIRequest.ECI_SYNC,                    //ECI call type
                        args[1], null, null,                            //CICS server, userid,password
                        args[2], null,                                  // program to be run & TranID
                        Commarea, CommareaSize,                         //COMMAREA & its length
                        ECIRequest.ECI_NO_EXTEND, 0);                   //ECI extend mode & LUW token
         jgate.flow(ecireq);                                            // flow the ECI request to CICS
         if (ecireq.Cics_Rc == 0)                                       // if good RC, show returned data in hex
           {
             System.out.println("\nProgram " + args[2]  + "returned following data:- \n");
             System.out.print("\tHex: ");
             for (int i = 0; i < Commarea.length; i++)
                { System.out.print(Integer.toHexString(Commarea[i])); }
           }
                                                                // if bad RC, display error message
        else { System.out.println("\nError from Gateway, RC:("  +ecireq.getCicsRcString()); }
        jgate.close();                                          // Close Gateway connection
    }
```

Example EPI Java code

```
import com.ibm.ctg.client.*;                                    // Client-side classes
import com.ibm.ctg.epi.*;                                       // EPI support classes
public class EPISamp
{
   public static void main (String [ ] args)                    //  Invoke program using :
   {                                                            //    java EPISamp <Gateway_URL> <CICS_Server>
      try {
         JavaGateway jgate = new JavaGateway();                 //Create a default JavaGateway
         jgate.setURL(args[0]);                                 // Set URL of remote Gateway
         jgate.open();                                          // Open the connection

         Terminal terminal = new Terminal(jgate, args[1], null, null) ;      // Add a terminal
         terminal.send(null, "CESN", null);                                  // Start CESN on the terminal

         Screen screen = terminal.getScreen();                               // Get the current screen
         for (int i = 1; i <= screen.fieldCount(); i++)                      // Loop round all fields
         {
            if (screen.field(i).textLength() > 0) {                          // Print non-empty fields
               System.out.println("Field " + i + ": " + screen.field(i).getText());
            }
         }

         screen.setAID(AID.PF3);                                             // Set the AID key to send
         terminal.send();                                                    // Return the screen to CICS
         terminal.disconnect();                                              // Disconnect the terminal
      }
      catch (Exception e) {                                                  // Handle any problems
         System.out.println(e.getMessage());
      }
   }
}
```

EPI JavaBeans

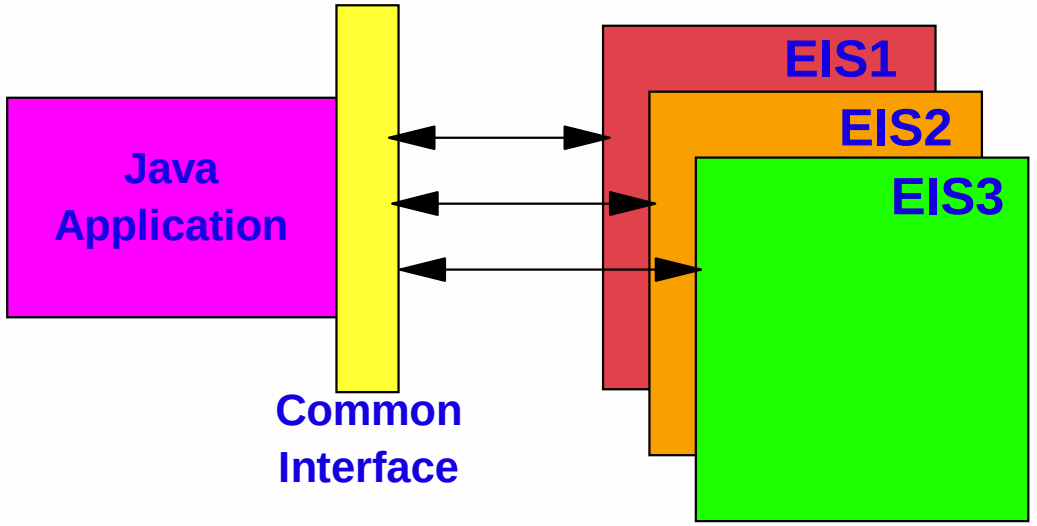
- Use to quickly create front-ends that connect to CICS
- The EPI Beans are ....
  - Built on top of the EPI Support classes
  - Fully compliant with Sun's JavaBeans API
- Four EPI Beans supplied
  - The ***EPITerminal*** bean
  - The ***EPIBasicScreenHandler*** bean
    - Specific ***ScreenHandler*** beans can also be created
  - The ***EPIScreenButtons*** bean
  - The ***EPIMonitor*** bean

Agenda

- Introduction
- Structure
- Terminal Servlet
- Network Protocols
- Connectivity to CICS
- Security Considerations
- Application Programming Interfaces
- **Connector Architecture Support**
- Further Information
- Summary

Connector Architecture Support

- Two architectures are available with the goal of providing a consistent Java client application interface for integration with existing Enterprise Information Systems (EIS)
  - IBM Common Connector Framework (CCF)
  - J2EE Connection Architecture



The IBM Common Connector Framework

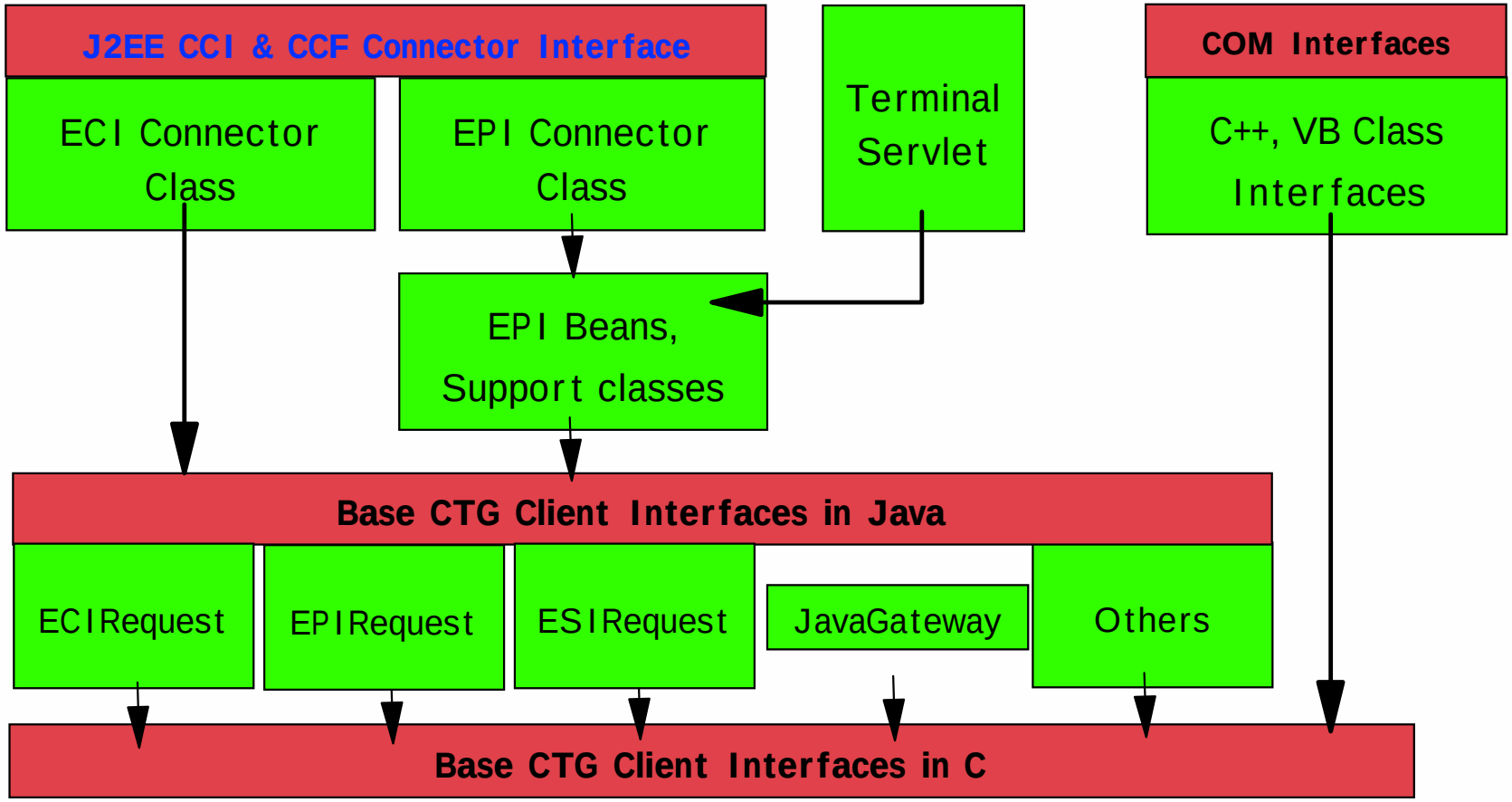
- The IBM ***Common-Connector-Framework (CCF)*** provides a consistent means of interacting with Enterprise resources from any Java execution environment
- The CICS Transaction Gateway V3 and V4 provides CCF Connectors for both ECI and EPI
- VisualAge for Java Enterprise Edition V3 and V4 provides support for CCF connectors
  - CICS, MQSeries, IMS
  - Encina, Host-on-Demand, SAP R/3
- CCF based connectors also available with VSE/ESA V2.5
  - Access VSE resources such as VSAM, Librarian, POWER

J2EE Connector Architecture

- "The J2EE Connector Architecture specifies a standard architecture for integrating Java applications with existing Enterprise Information Systems"....Sun Microsystems
- J2EE Connector Architecture heavily influenced by IBM's Common Connector Framework architecture
- CICS Transaction Gateway V4 provides J2EE Connectors for both ECI and EPI
- VisualAge for Java Enterprise Edition V4 and WebSphere Studio Application Developer provide J2EE connector support
  - CICS, MQSeries, IMS, Host-on-Demand
  - SAP R/3, PeopleSoft, Oracle, J D EDwards
- J2EE Connectors provide the strategic solution
  - Will replace CCF Connectors

Connector Interfaces

- Connector interfaces built on existing Gateway classes



Agenda

- Introduction
- Structure
- Terminal Servlet
- Network Protocols
- Connectivity to CICS
- Security Considerations
- Application Programming Interfaces
- Connector Architecture Support
- **Further Information**
- Summary

Further Information

- **Web Sites**
  - **CICS (main site)**
    - **http://www.ibm.com/software/cics**
  - **CICS Clients and Gateways**
    - **http://www.ibm.com/software/cics/ctg/index.html**
  - **CICS (SupportPacs)**
    - **http://www.ibm.com/software/cics/txppacs**
  - **Redbooks**
    - **http://www.redbooks.ibm.com**
- **Announcement Letters**
  - CICS Transaction Gateway V4: 201-187
  - CICS Transaction Server for VSE/ESA: 299-156, 200-293
  - VSE/ESA V2.6: 201-325
  - VSE/ESA V2.7 Preview: 202-038

Further Information

Publications

Title	Number
IBM CICS Transaction Gateway V4 product publications	
CICS Transaction Gateway V4.0 Windows Client Administration	SC34-5940
CICS Transaction Gateway V4.0 Windows Gateway Administration	SC34-5932
CICS Transaction Gateway V4.0 Gateway Programming	SC34-5938
IBM Redbooks	
CICS Transaction Gateway V3.1, The WebSphere Connector for CICS	SG24-6133
Revealed! Architecting Web Access to CICS	SG24-5466
Java Connectors for CICS: Featuring the J2EE Connector Architecture	SG24-6401
e-business Solutions for VSE/ESA	SG24-5662

- Accessible from the CICS and Redbooks Web sites

Further Information

- **CICS SupportPacs** - download from CICS Web site
  - CA83: CICS Gateway for Java - Test suite
  - CA88: CICS Gateway for Java - ECI/EPI test applications
  - CA89: Web access to CICS using Java Servlets
- **IBM Planning Services for CICS Web Enablement**
  - http://www.as.ibm.com/asww/offerings/mww73bE.html
- **Related Conference Sessions**
  - VSE Plays Well With Others
  - VSE Connectors - Architecture and Use
  - CICS Transaction Server for VSE/ESA: CICS Web Support
  - JavaBeans: It's More Than Just coffee!
  - Building Java Applications
  - WebSphere Application Server for VSE Users

Agenda

- Introduction
- Structure
- Terminal Servlet
- Network Protocols
- Connectivity to CICS
- Security Considerations
- Application Programming Interfaces
- Connector Architecture Support
- Further Information
- **Summary**

CICS Transaction Gateway Summary

- Enables access to CICS applications and transactions from Web Browsers and Java application environments
- Is the strategic IBM e-business Connector for CICS
- Provides the ECI, EPI and ESI programming interfaces
- Provides 3270 transaction access via the Terminal Servlet
- Supports the Common Connector Framework
- Supports the J2EE Connection Architecture
- Provides network security via industry standard SSL
- Well proven and established product
- Supports CICS TS for VSE/ESA and CICS/VSE V2.3

IBM

The CICS Transaction Gateway: Web and Java access to CICS

Chris Smith
smithch@uk.ibm.com

Fort Mitchell, 12 - 16 April 2002