Computer Associates®

Java and the Wild Wild Web
Crash Course No.1

Maria Sarikos
maria.sarikos@ca.com

Agenda

- The Web’s Impact on the DB2 Techie
- Java and Web Speak – A quick translation
- EJB’s (Enterprise Java Beans)
- JDBC
- SQLJ

Agenda

- The Web’s Impact on the DB2 Techie
- Java and Web Speak – A quick translation
- EJB’s
- JDBC
- SQLJ

The Web's Impact on the DB2 Techie

- **Opening up to the big wide world**
  - Traditional users → internet users
  - 24x7 is critical
  - Recoverability (application and disaster recovery)
  - Data Sharing
  - Lots of new performance tuning opportunities!!
- **E-business is about integration**
  - 20-30 years of IT investment on the mainframe
  - Leverage your existing IT assets
  - 70 – 80% of world's data reside on IMS and DB2

Computer Associates®

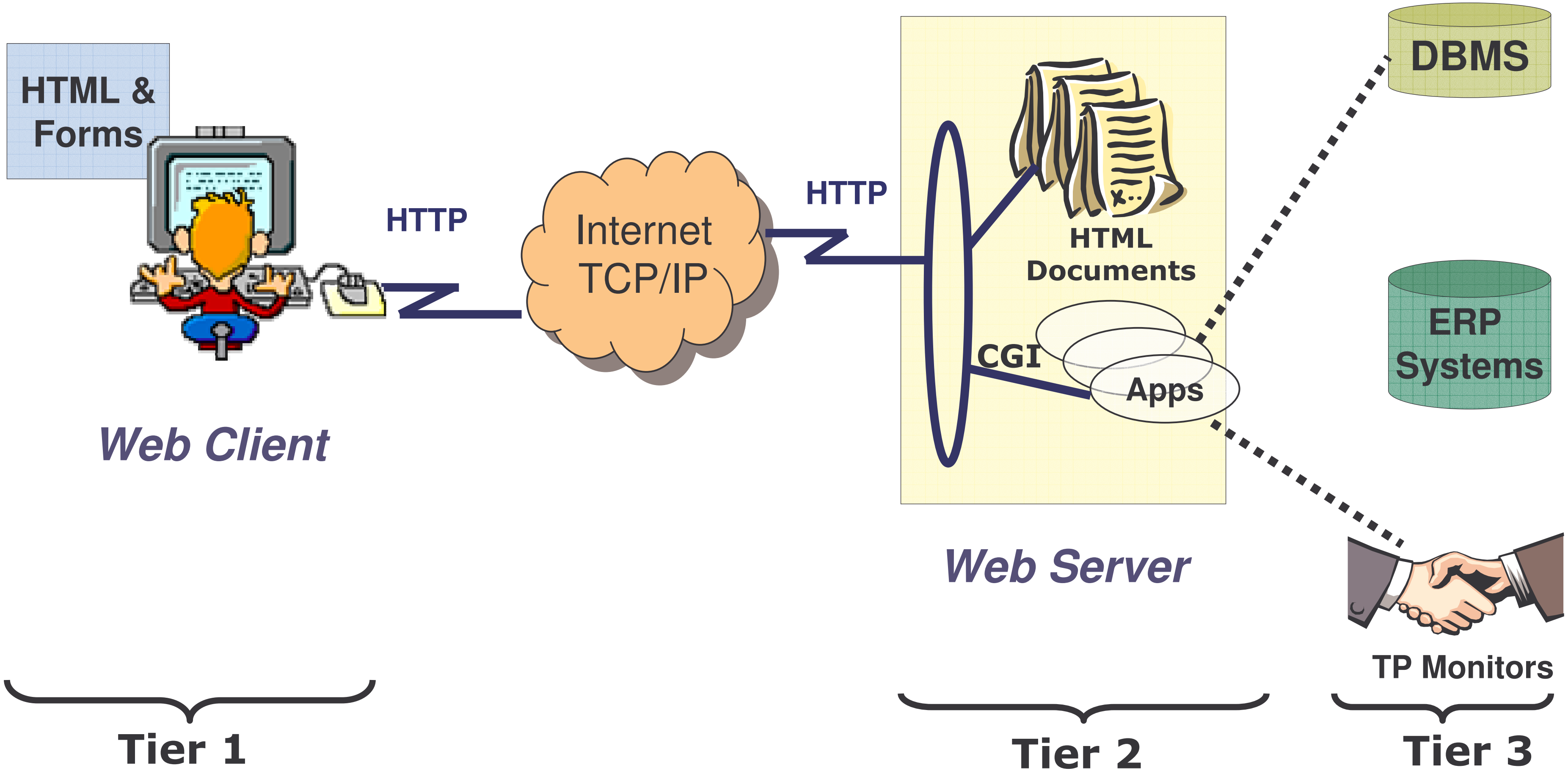
The Web's Impact on the DB2 Techie

- **How often do you re-boot your mainframe ?**
- **30 years of solid infrastructure**
- **Takes less Man power to manage a mainframe**
  - How many people does it take to look after a mainframe vs 1000 servers?

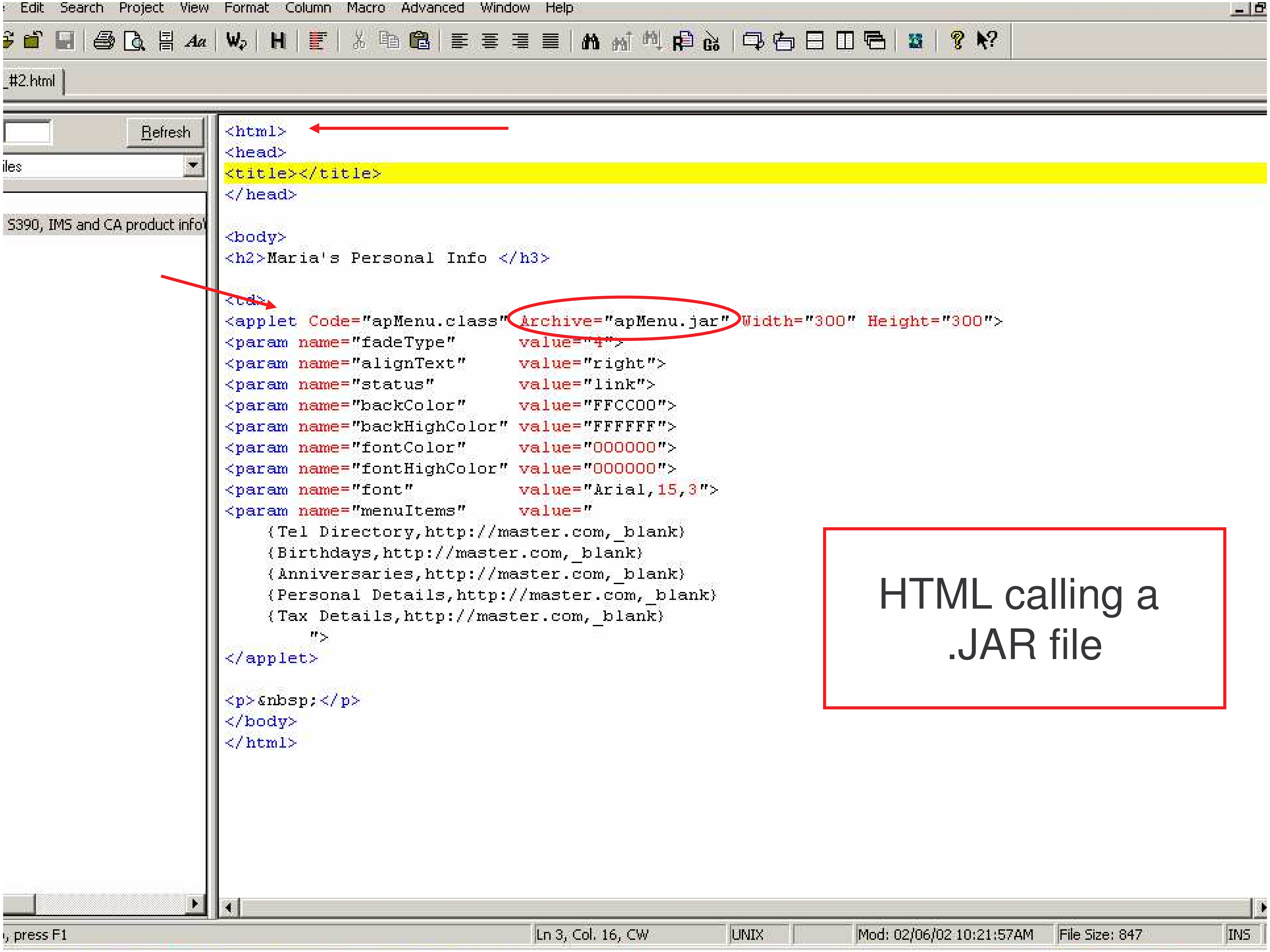
Agenda

- The Web’s Impact on the DB2 Techie
- **Java and Web Speak – A quick translation**
- EJB’s (Enterprise Java Beans)
- JDBC
- SQLJ

Web Components



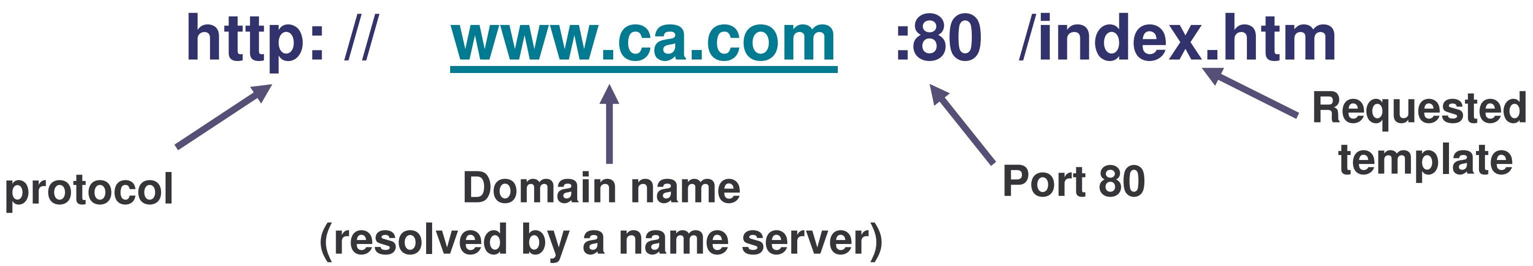
Computer Associates®


Edit Search Project View Format Column Macro Advanced Window Help
_#2.html
Refresh
iles
S390, IMS and CA product info
, press F1
Ln 3, Col. 16, CW
UNIX
Mod: 02/06/02 10:21:57AM
File Size: 847
INS

```html
<html>
<head>
<title></title>
</head>

<body>
<h2>Maria's Personal Info </h3>

<td>
<applet Code="apMenu.class" Archive="apMenu.jar" Width="300" Height="300">
<param name="fadeType"       value="4">
<param name="alignText"      value="right">
<param name="status"         value="link">
<param name="backColor"      value="FFCC00">
<param name="backHighColor"  value="FFFFFF">
<param name="fontColor"      value="000000">
<param name="fontHighColor"  value="000000">
<param name="font"           value="Arial,15,3">
<param name="menuItems"      value="
    {Tel Directory,http://master.com,_blank}
    {Birthdays,http://master.com,_blank}
    {Anniversaries,http://master.com,_blank}
    {Personal Details,http://master.com,_blank}
    {Tax Details,http://master.com,_blank}
        ">
</applet>

<p>&nbsp;</p>
</body>
</html>
```

HTML calling a .JAR file

URL (uniform resource locator)



http: // www.ca.com :80 /index.htm

protocol

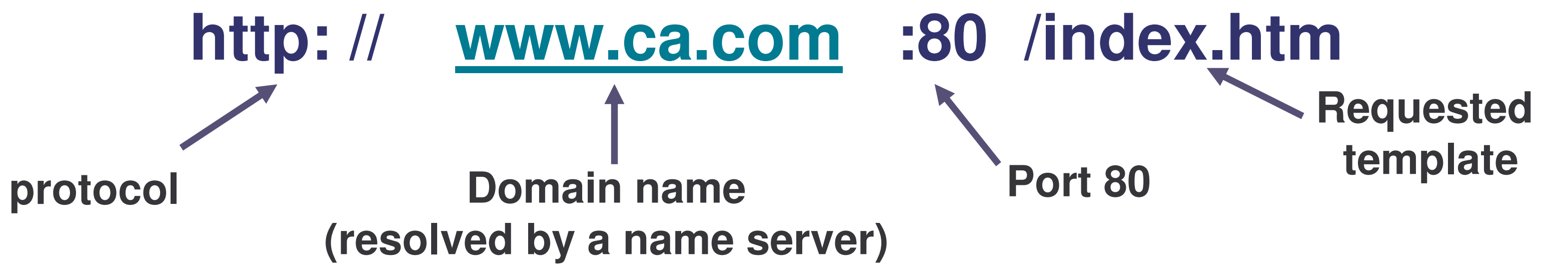
Domain name (resolved by a name server)

Port 80

Requested template

- Global address for documents and other resources on the world wide web (www)
- Browser breaks the URL into three parts
    - Protocol (http, ftp , Gopher, FTP, News, Mailto etc …)
    - Server name (www.ca.com)
    - File name
- Browser then “talks” with a name server to translate the server name into an IP address

URL (uniform resource locator)



http: // www.ca.com :80 /index.htm

protocol

Domain name
(resolved by a name server)

Port 80

Requested
template

- Browser then connects to the web server at the IP address using Port 80
- The browser using the HTTP protocol, sends a “get” request to the server for file “www.ca.com/index.htm”
- The server then responds to the browser, by sending the HTML text for the web page
- The browser reads the HTML tags and formats the page on the browser screen

HTTP and HTTPS

- **HTTP**
  - Application level protocol for distributed, collaborative, hypermedia information systems
  - It is generic and “stateless” object-orientated protocol
    - “stateless” means it does not keep track of connections
  - Based on a request/response system algorithm
  - Web browsers use HTTP to communicate with Web Servers

- **HTTPS**
  - Secure Sockets Layer provides security
    - Many web sites use SSL to get confidential info eg. credit card numbers.
    - URLs that use an SSL connection start with *http**s**:* instead of *http*:

Computer Associates®

CGI (Common Gateway Interface)

- Specification for transferring information between the www and CGI program
- A CGI program is ANY program designed to accept and return data that conforms to the CGI specification.
- The program could be written in any programming language ie. C, Perl, Java, Visual Basic
- CGI programs are the most common way for Web servers to interact dynamically with users
- Many HTML pages that contain forms, use a CGI program to process the form's data once it's submitted
- The use of CGI is a *server-side* solution because the processing occurs on the Web server
- One problem with CGI is that each time a CGI script is executed, a new process is started. For busy web sites, this can slow down the server noticeably
- A more efficient solution is to use Java Servlets

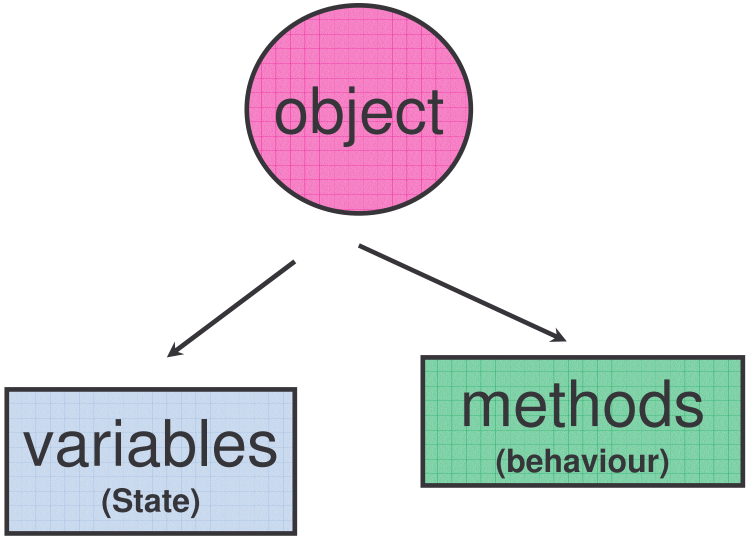
Object Orientated Programming

state

- Amphibian
- Eat insects
- Four legs
- Skin acts as camouflage
- Sticky tongue
- Little boys love them
- Little girls hate them

behaviour

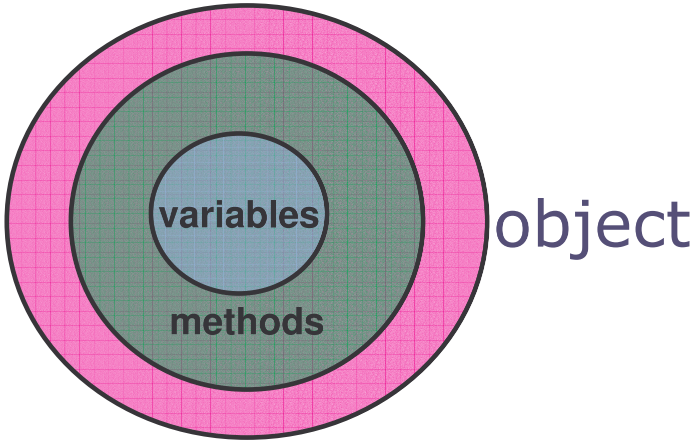
- Jumps
- Eats
- Secretes fluids
- Reproduces



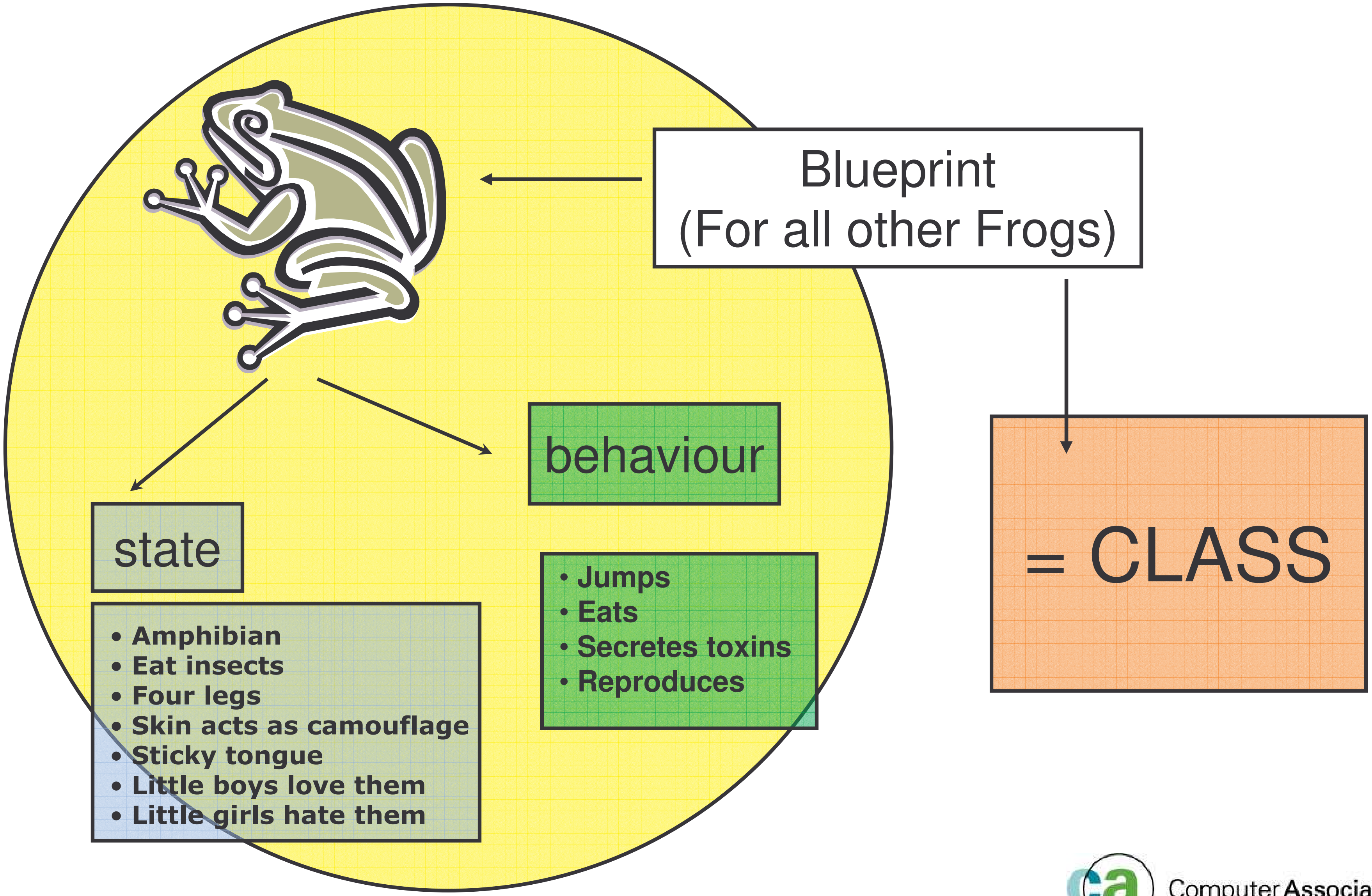
object

variables
(State)

methods
(behaviour)



Object Orientated Programming



Blueprint
(For all other Frogs)

= CLASS

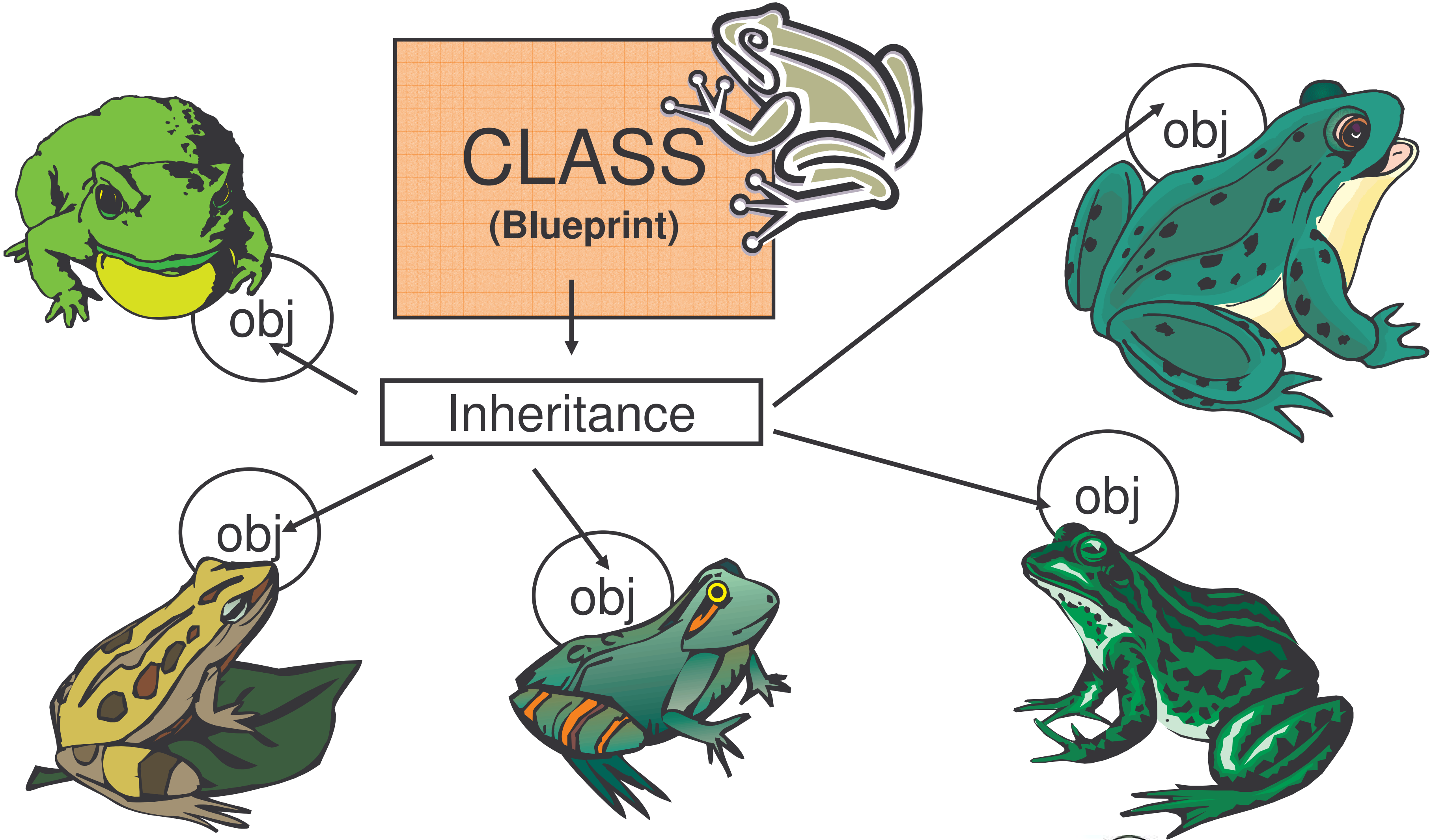
state

- Amphibian
- Eat insects
- Four legs
- Skin acts as camouflage
- Sticky tongue
- Little boys love them
- Little girls hate them

behaviour

- Jumps
- Eats
- Secretes toxins
- Reproduces

Object Orientated Programming



Computer Associates®

JavaScript and Applets

- **JavaScript (Netscape) and JScript for (Microsoft)**
  - Cross Platform object-oriented language
  - Ability to interact with HTML forms
  - Validate user input
  - Improve client side performance, by reducing the number of requests flowing over the network
  - Not recommended to use JavaScript/JScript on the server side due to variations between Microsoft and Netscape

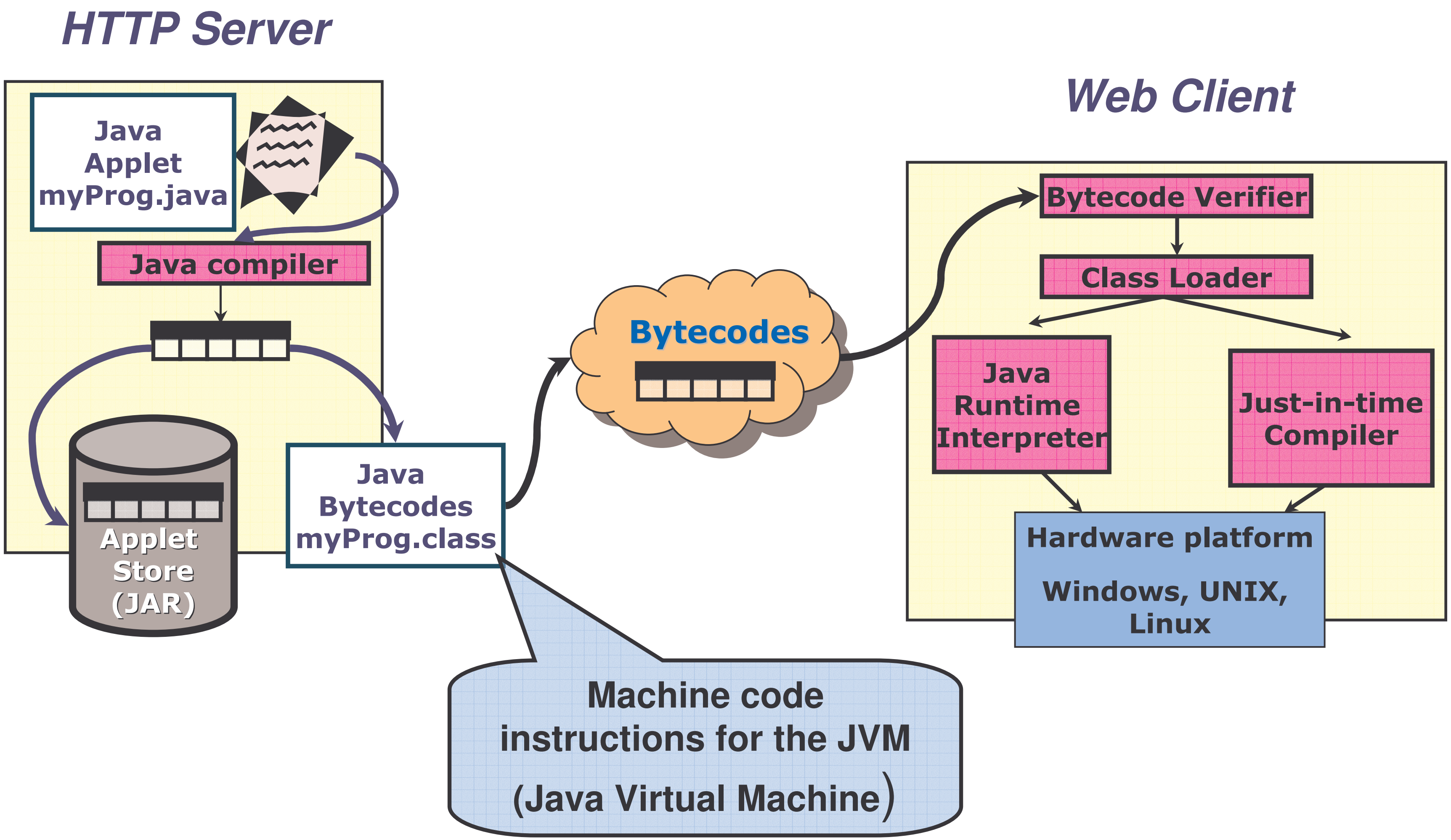
- **Java Applet**
  - Java program downloaded from the web server and runs on the browser
  - Applets rarely consist of one CLASS file
  - JARs (Java Archive File) – packages the class files into one package (hence reducing the number of requests to the server)

Servlet, JSP (Java Server Page)

- **Java Servlet**
  - Replace CGI based techniques in web programming
  - Only run on Web Server

- **JSP (Java Server Page)**
  - Simplify the process of creating web pages
  - JSPs contain HTML
  - Insert dynamic content into web pages. For example
    - Answer to a search
      - (All book titles beginning with “Once upon a time”)
    - List the last ‘n’ products viewed
  - Contain some Java code which encapsulates the logic and generates the page content
  - The java code, may call “beans” to access re-usable components and back-end data

Java Bytecodes



CORBA and LDAP

- **CORBA (Common Object Request Broker Architecture)**
  - Set of conventions, standards and protocols for interprocess communication
  - Developers can write applications for many different Operating System's at once in any number of languages
  - Any application that matches that has the defined interfaces and protocols is allowed to communicate with another CORBA implementation

- **LDAP (Lightweight Directory Access Protocol)**
  - Repository to retrieve data, resources, addresses etc.
  - These repositories are known as "directories"
  - Finding information for distributed systems without directories would not be easy
  - Each vendor may implement their own version
  - X.500 (adopted by ISO – International Standards Organisation) is a directory standard of choice

Agenda

- The Web’s Impact on the DB2 Techie
- Java and Web Speak – A quick translation
- **EJB’s (Enterprise Java Beans)**
- JDBC
- SQLJ

What is an EJB ? (Enterprise Java Bean)

- **Emerging as the preferred architecture for Java programmers**

- **EJB extends Java with the following:**
  - Security
  - Transaction management
  - Remote method invocation
  - Service improvements

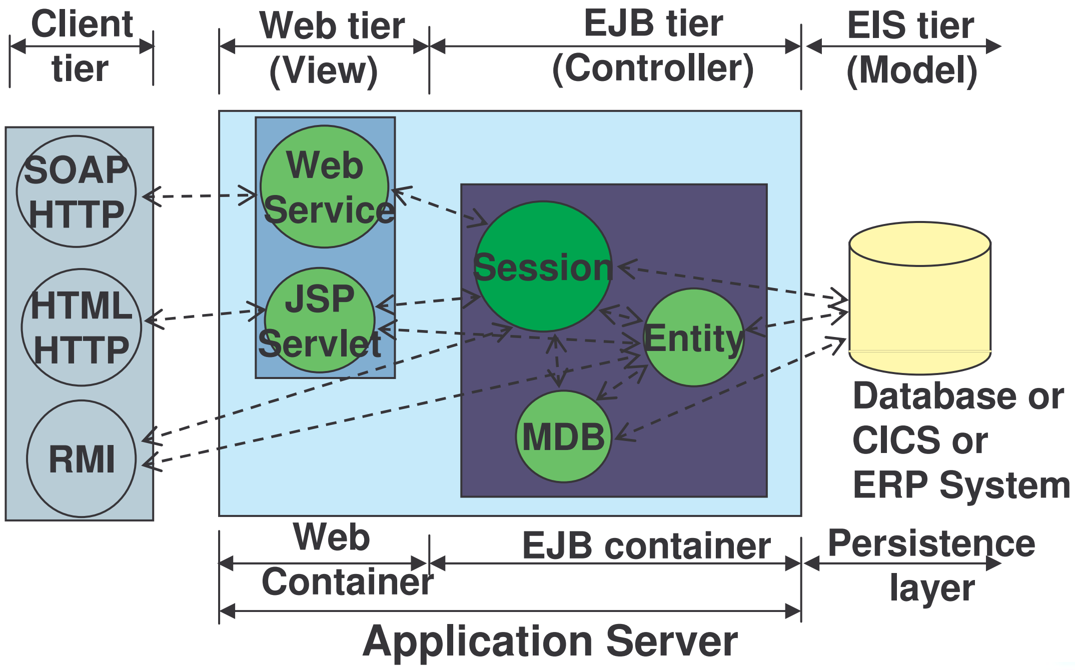
- **What do EJB's provide?**
  - Unparalleled portability
  - Supports CORBA standard
  - Provides connectors with existing programs/data
  - Access to local and remote objects/methods

What is an EJB ? (Enterprise Java Bean)

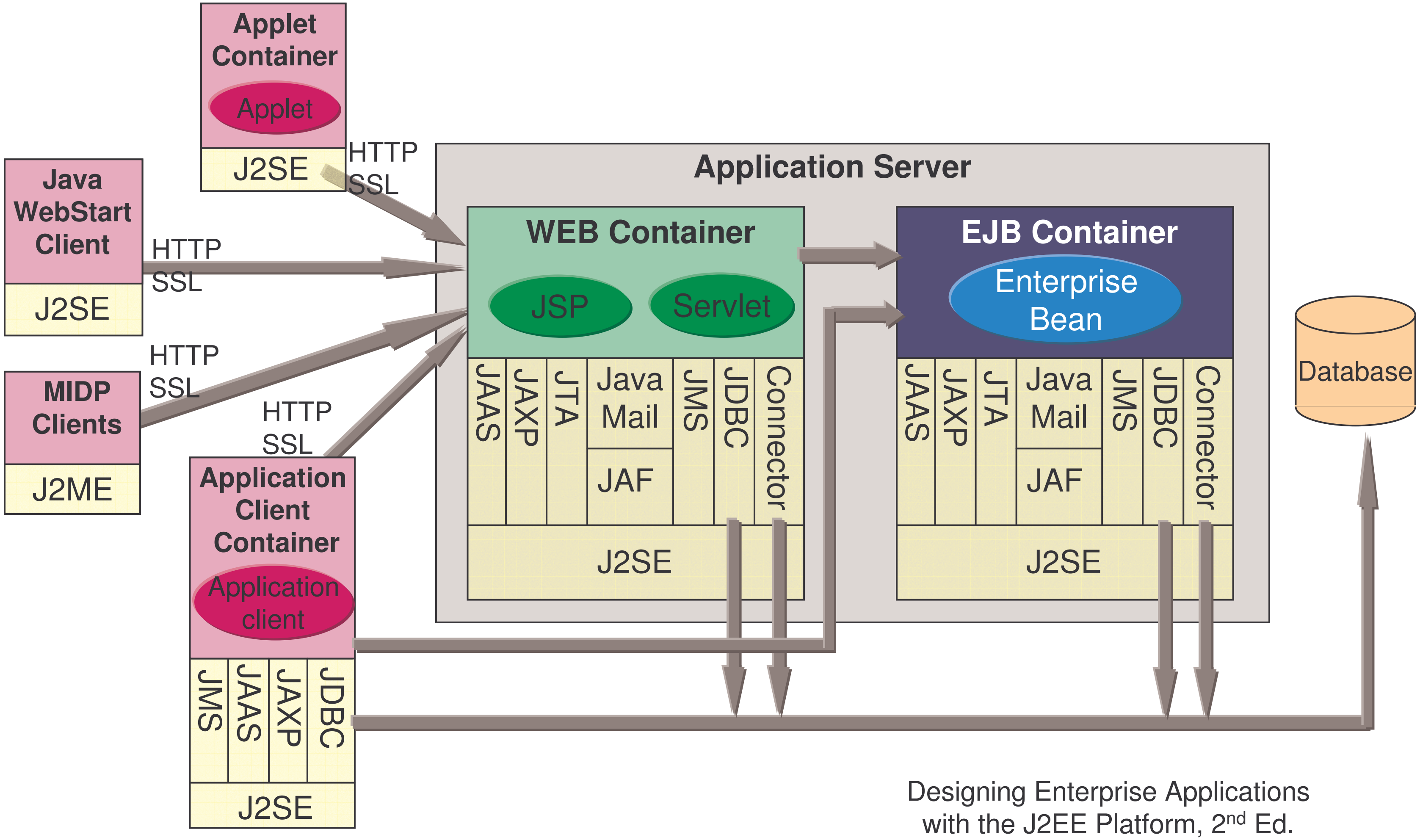
- **In a nutshell …**
  - Portability
  - Standardisation
  - Keep business logic and SQL “inside” the firewall
  - Separate the business logic from the system support side
  - Give you an “object view” of the data

Application Servers

- convergence of component based systems and OLTP monitors
- Connection pooling for external resources.
- EJBs placed in container that conforms to standard
- application server provides services to container
- EJB runs in any application server that conforms to J2EE.



J2EE Components and Containers



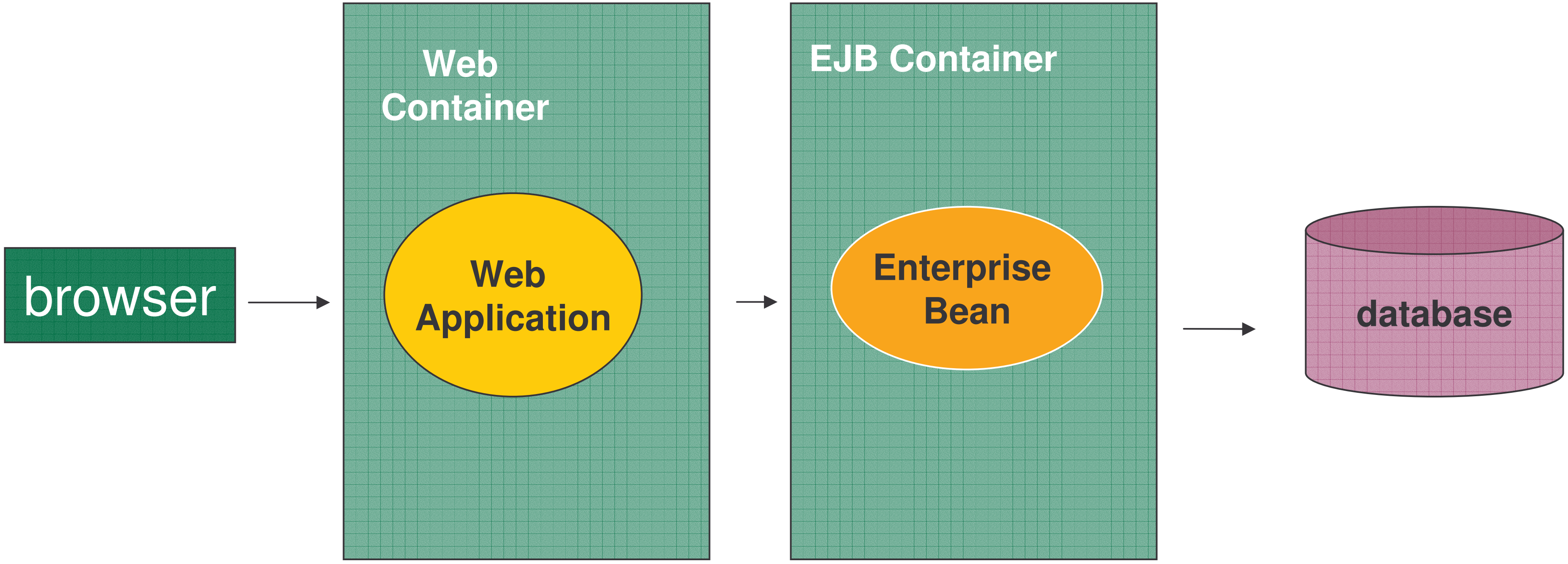
Designing Enterprise Applications
with the J2EE Platform, 2nd Ed.

Computer Associates®

J2EE Containers

- Standardized runtime environment that provides services to components.
  - performance impact in how it provides those services
- Abstract all resource access.
- You code to the J2EE standard.
- EJB communicates with the container.
- Don’t care how app server implements resources such as persistence, transactions, security.
- Drop your EJB into the container, and the app server handles connection to the implemented resource.
  - Auto-generates a substantial amount of code
- Provides automated life cycle management.
- “Specialised containers can provide additional services beyond those defined by the EJB specification”
  - EJB 2.0 specification 4.1.1

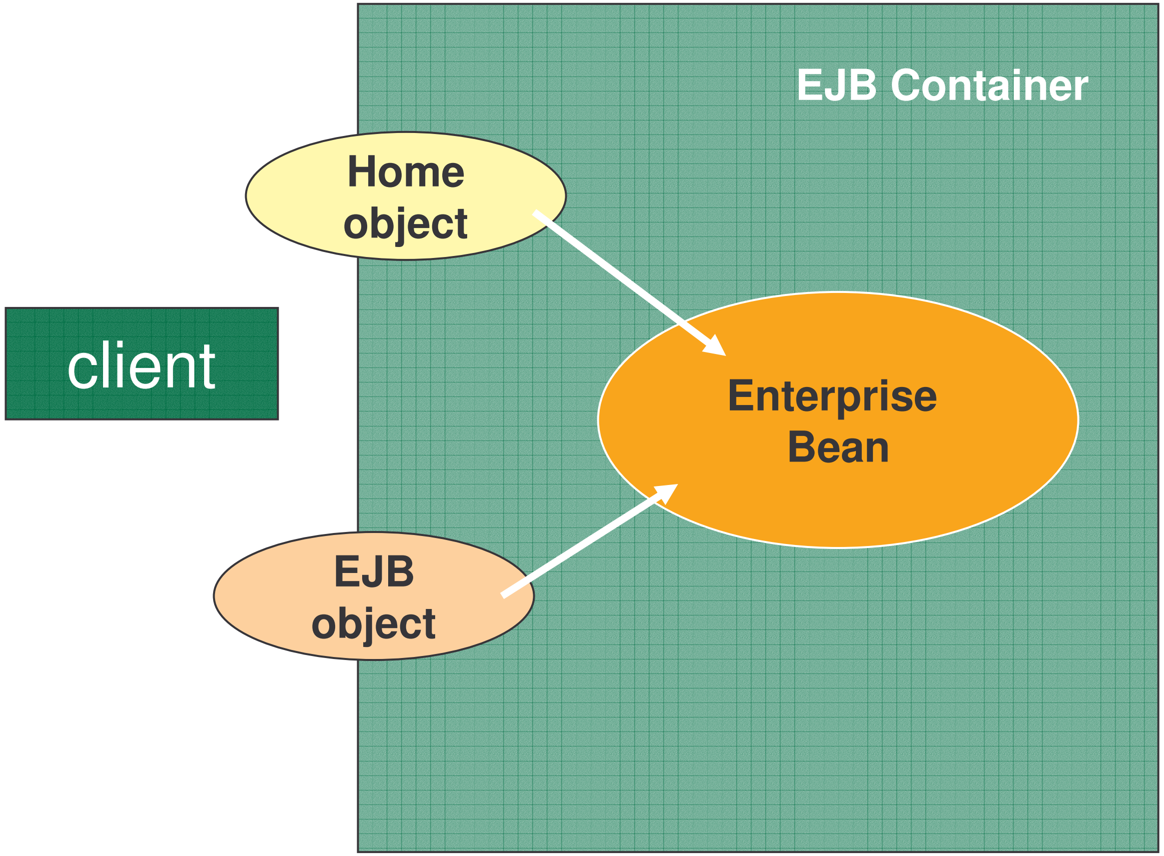
EJB Architecture



EJB Architecture

The client can only see the home and remote interfaces of the EJB

Access to an EJB is done through JNDI (Java Naming Directory Interface)



EJB Architecture - 4 Components

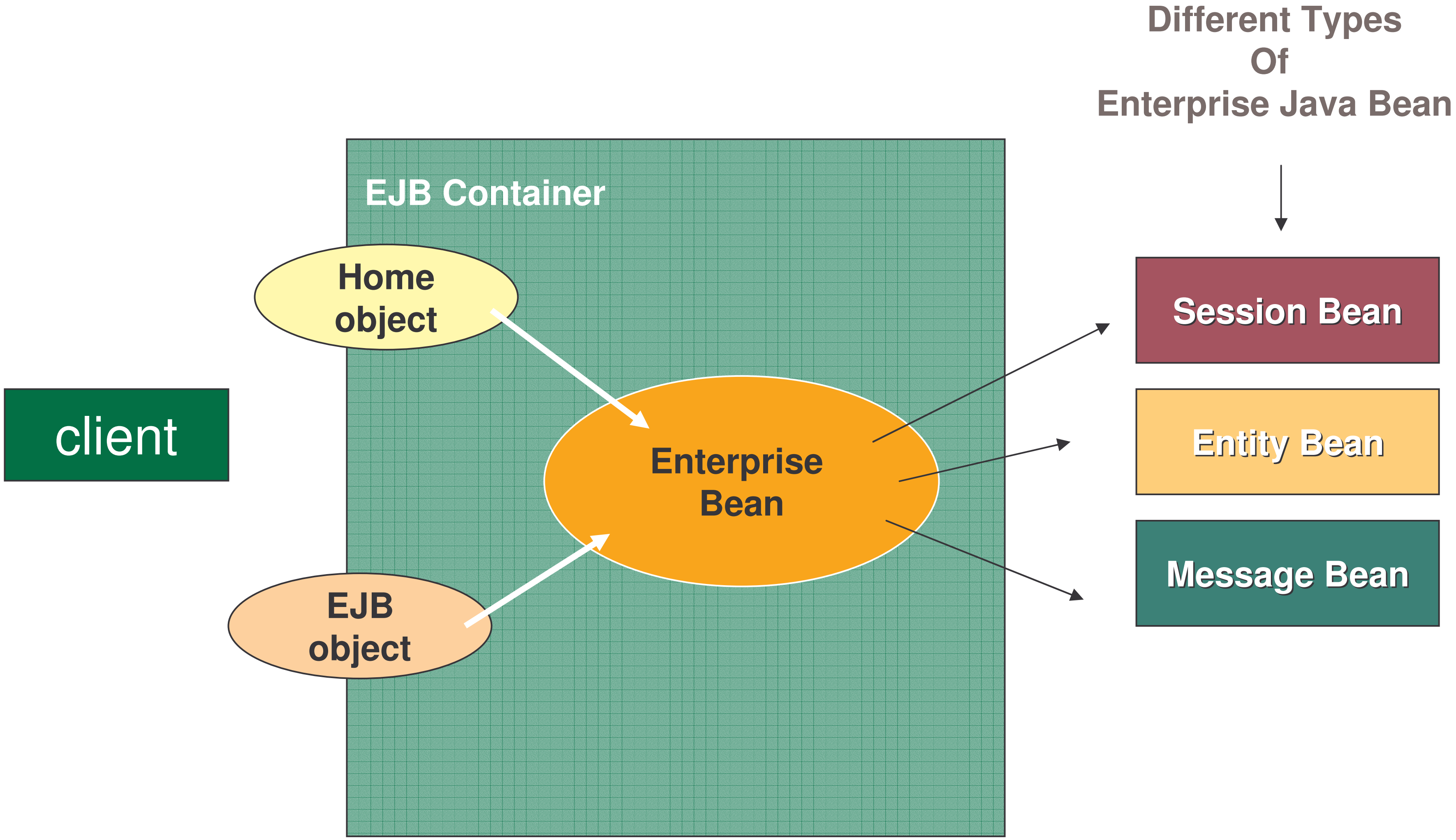
- **EJB client** ***(component - No.1)***
  - Process that requires the service provided by the bean
  - Client does not need to store the location of the EJB

- **Application Server** ***(component - No.2)***
  - Provides ALL the underlying services required by the EJB. These include :
    - *Transaction Services*
    - *Naming Services*
    - *Database Access Services*
    - *Security Services*
    - *Life Cycle and thread management services*

EJB Architecture - 4 Components

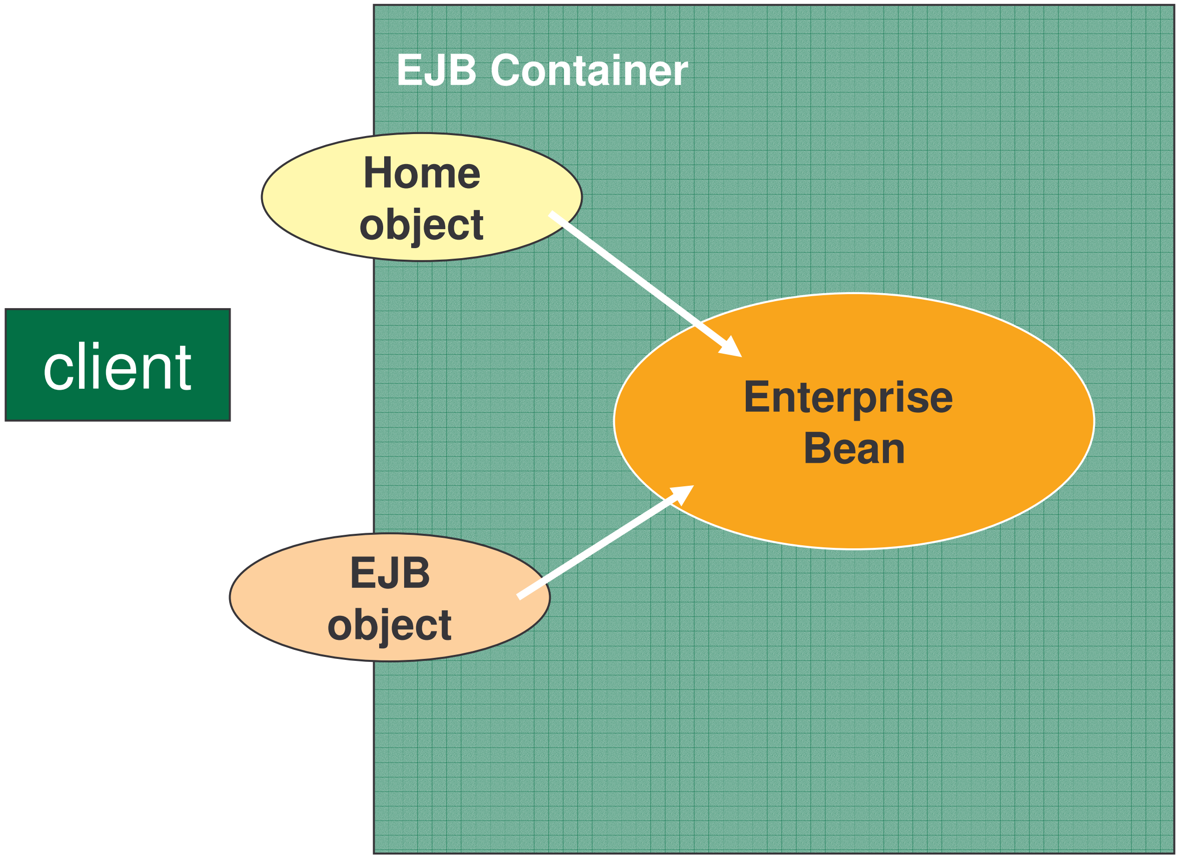
- **EJB Container** ***(component - No.3)***
  - Provides the environment in which the EJB runs. The container manages the following:
    - *Advertising the EJBs are available in this container*
    - *The Life Cycle of the EJB*
    - *Persistence of the EJBs*
    - *Authenticating Clients*

EJB Architecture



Computer Associates®

EJB Architecture



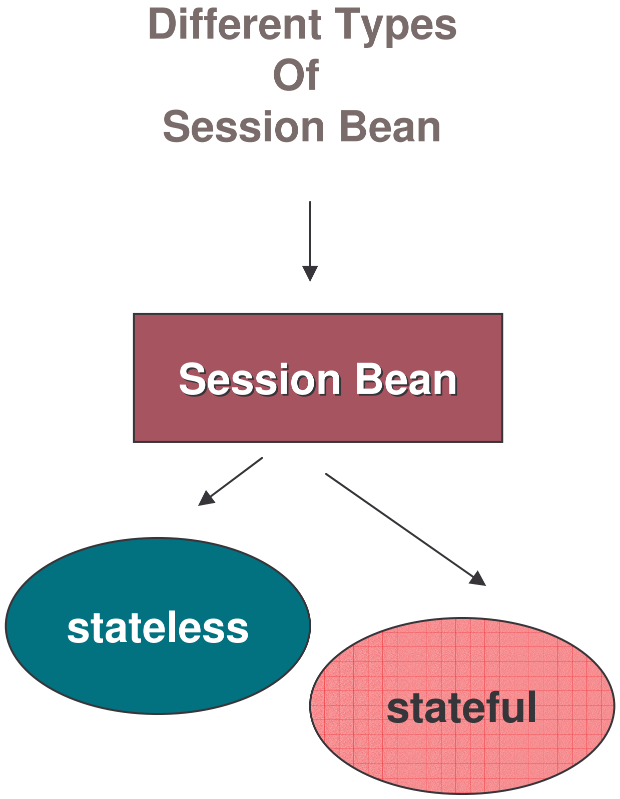
client

EJB Container

Home object

Enterprise Bean

EJB object



Different Types Of Session Bean

Session Bean

stateless

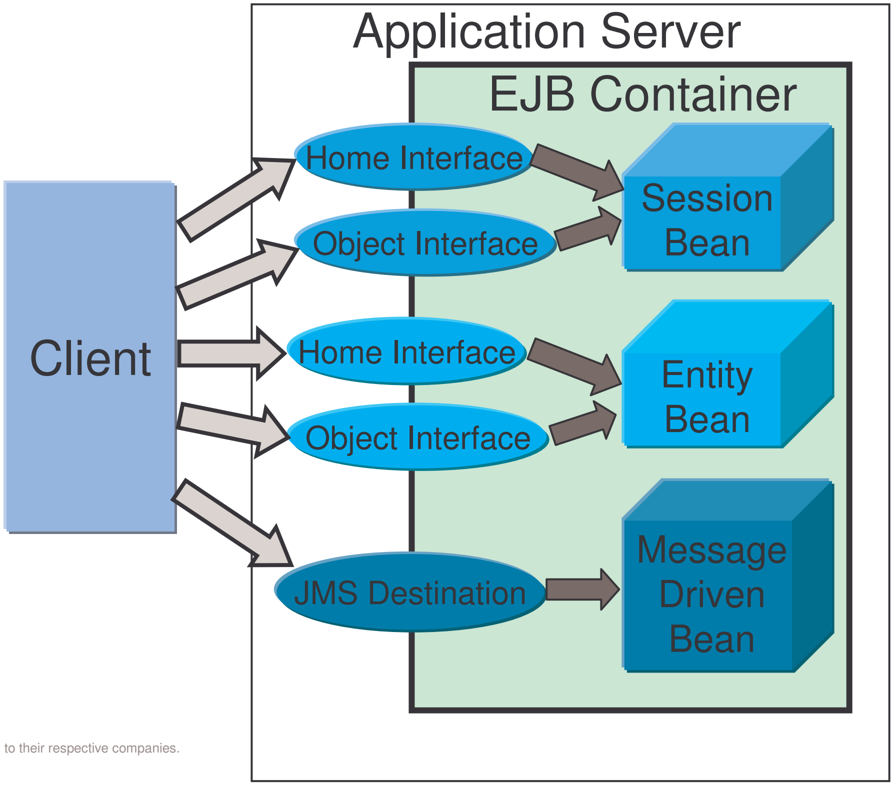
stateful

EJB Architecture - 4 Components

And finally the EJB *(component - No.4)*

- The EJB component is the Java class (or classes) that represents the business-logic component

- There are three types of EJB's
  - ***Session Beans***
  - ***Entity Beans***
  - ***Message Beans***



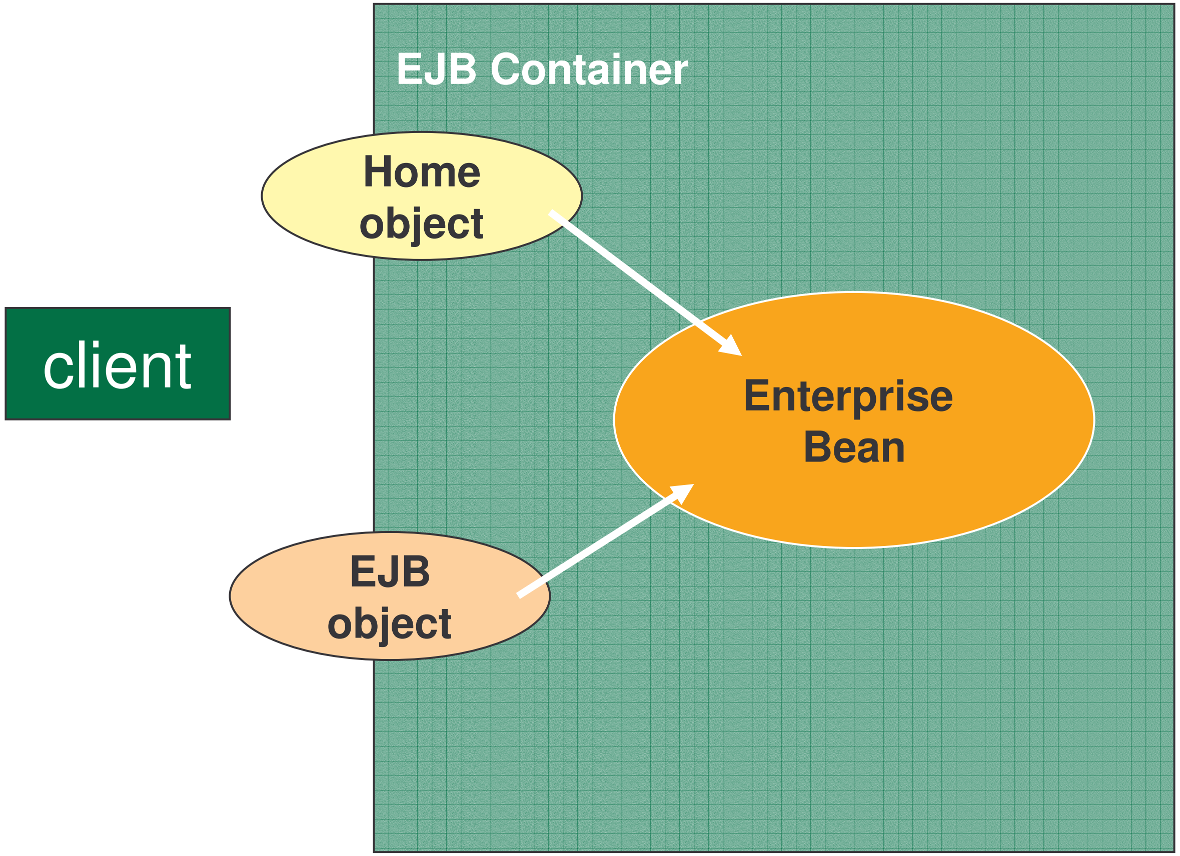
EJB Architecture - 4 Components

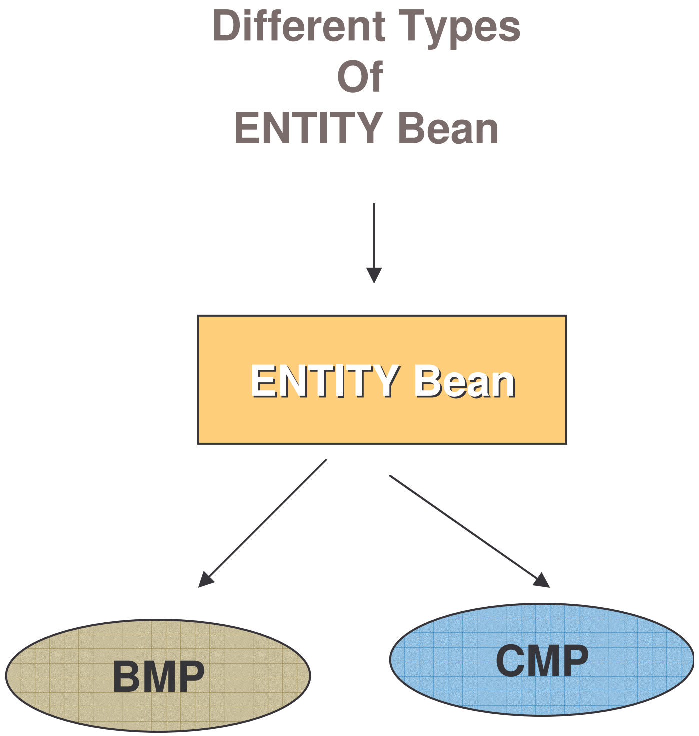
EJB - Session Beans *(component - No.4)*

- Represent a process that will be performed on the server
- The Client requests a service from a session bean
- Each client has its own instance of the bean
  - Instances of session beans cannot be shared among multiple clients

- **Two types of SESSION BEAN**
  - **Stateless** – eg. Check whether a stock code is valid
  - **Stateful** – eg. Keep track of a value within a counter

EJB Architecture - 4 Components





Computer Associates®

EJB Architecture - 4 Components

EJB – Entity Beans *(component - No.4)*

- Map a Java class to a data source
- Source can be a single row in
  - a database;
  - an entire table
  - some type of data not stored in a database
- Each Entity Bean has a primary key associated with it, which identifies the data within
- It is difficult to control changes to multiple copies of the same data
- So only one instance of an entity bean exists for any given primary key

EJB Architecture - 4 Components

EJB – Entity Beans *(component - No.4)*

- **Two types of ENTITY BEAN**
  - ***CMP*** *(Container managed persistence)*
    - *Simplest form and relies on the container to provide all database access calls*
  - ***BMP*** *(Bean managed persistence)*
    - *Provides all the database access calls within the bean itself. The disadvantage this, is that the bean is very closely tied to the underlying architecture*

EJB Query Language

- Automatic persistence is a good feature of CMP entity beans

- But BMP beans are preferable because you can create "more complex finder methods"

- With BMP beans you CAN write a complex SQL stmt for a finder method – with CMP you CANNOT!!
  - So EJB Query Language solves this dilemma

- EJB Query Language is similar to SQL

- For performance the EJB Query language can be compiled to a target language of a database

BMP entity beans vs CMP entity beans

BMP entity beans are most appropriate when you want to :

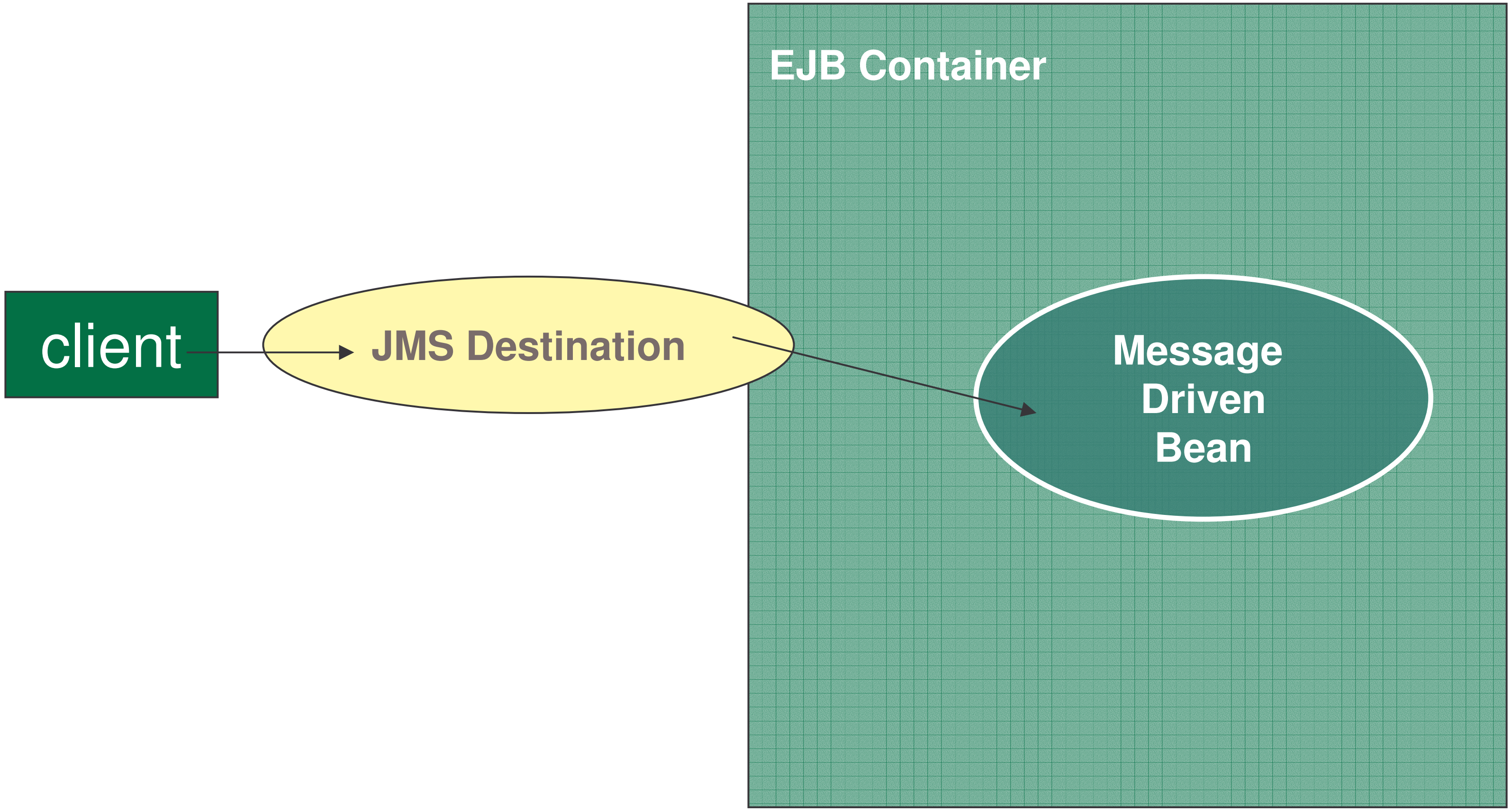
- Wrap a non-relational backend (eg. CICS, IMS)
- Map your entity EJB onto several tables
- Have SQL joins over several tables
- Use SQLJ instead of JDBC to access DB2

CMP entity beans are most appropriate when you :

- Do not want to write the persistence code of the EJB manually
- Only want to access DB2
- Have a one-to-one mapping (between an entity bean and a database table)
- Don’t require the added performance benefit and security that SQLJ can offer

Computer Associates®

EJB Architecture



EJB Architecture - 4 Components

EJB – Message Beans *(component - No.4)*

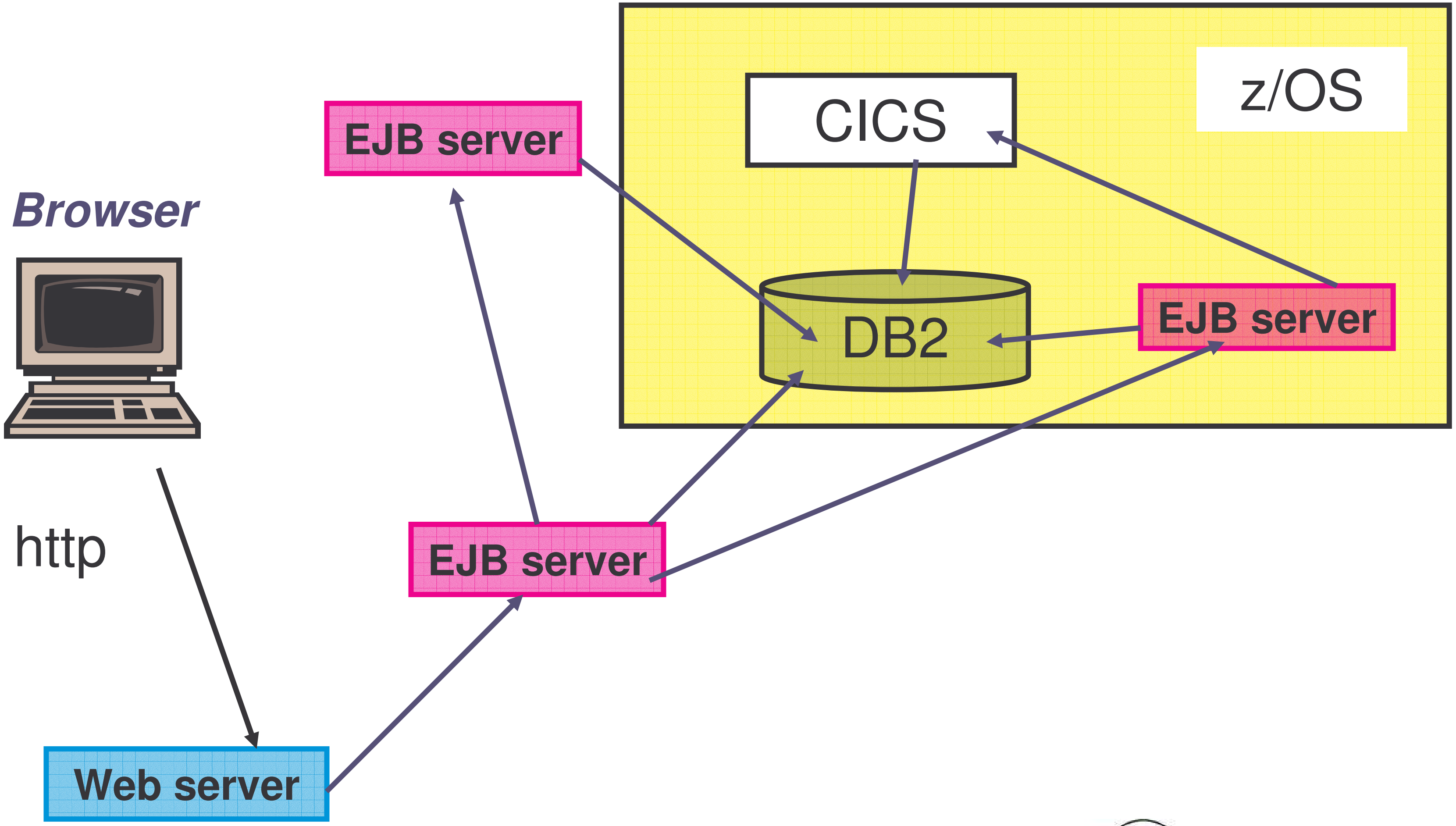
- Messaging is a service that provides communication between applications or software components
- Any application or software that uses a messaging service is called a “messaging client”
- A messaging client can send and receive messages
- An EJB Messaging bean can be asynchronously invoked to handle the processing of incoming JMS messages
  - “ Asynchronous message consumer”

EJB Architecture - 4 Components

EJB – Message Beans *(component - No.4)*

- Asynchronously consume JMS messages
  - Stateless
  - Transactions can be either Bean Managed or Container Managed
  - Queue/Topic can be configured by deployment descriptor
  - Can be used to provide control of workflows
    - A session bean to process the message
- New in EJB 2.0

DB2 and EJB Transactions



Agenda

- The Web’s Impact on the DB2 Techie
- Java and Web Speak – A quick translation
- EJB’s (Enterprise Java Beans)
- **JDBC**
- SQLJ

JDBC

- Generic interface for writing platform independent applications that can access any SQL database

- The APIs are defined within 16 interfaces and classes that support
  - Connecting to a Database
  - Executing SQL statements
  - Processing Results

- DB2's Java support includes JDBC a vendor-neutral dynamic SQL interface that provides access to your app, via standardised Java methods

- JDBC is similar to CLI – it does not require pre-compiling or binding

- ***An application using JDBC ONLY uses dynamic SQL***

Advantages of using JDBC

- Generic for all DBMS therefore, with Java and JDBC you can write SQL and port it to any platform that has the JDK (Java Development Toolkit)
  - “Write once, execute anywhere”

- JDBC applications do not require pre-compiling, hence they are easier to maintain in that respect

- JDBC allows more flexible SQL statements

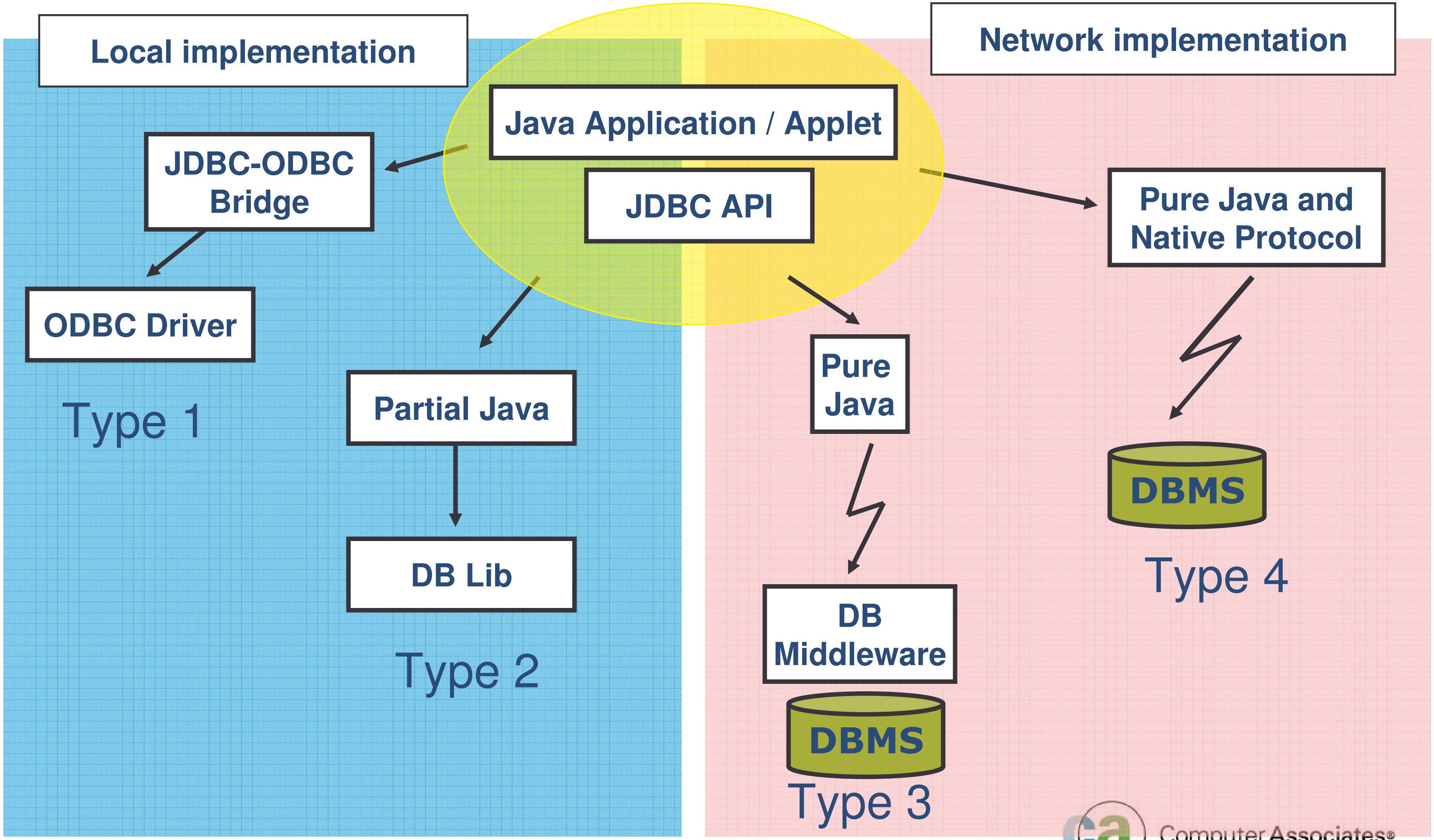
BEWARE!! With JDBC access to the underlying tables must be given to the user, so that the data can be accessed

- With STATIC SQL access is only given to the plan/package

- There is an alternative **DYNAMICRULES(BIND)** – but this could create a
  security exposure depending on how it is implemented!!

Computer Associates®

JDBC – Driver Types



Local implementation

Network implementation

Java Application / Applet

JDBC API

JDBC-ODBC Bridge

ODBC Driver

Type 1

Partial Java

DB Lib

Type 2

Pure Java

DB Middleware

DBMS

Type 3

Pure Java and Native Protocol

DBMS

Type 4

Agenda

- The Web's Impact on the DB2 Techie
- Java and Web Speak – A quick translation
- EJB's (Enterprise Java Beans)
- JDBC
- SQLJ

SQLJ

- **Embeds SQL in Java programs**

- **An application using SQLJ can only use static SQL**

- **The SQLJ environment consists of:**
  - Embedded SQL
  - Translator (written in Java)
  - A runtime environment

- **Set of Java classes that implement SQLJ's runtime support**

- **SQLJ – when using SQLJ to access a DB2 server, the program will execute SQL statements using the privileges assigned to the user who created the database package**

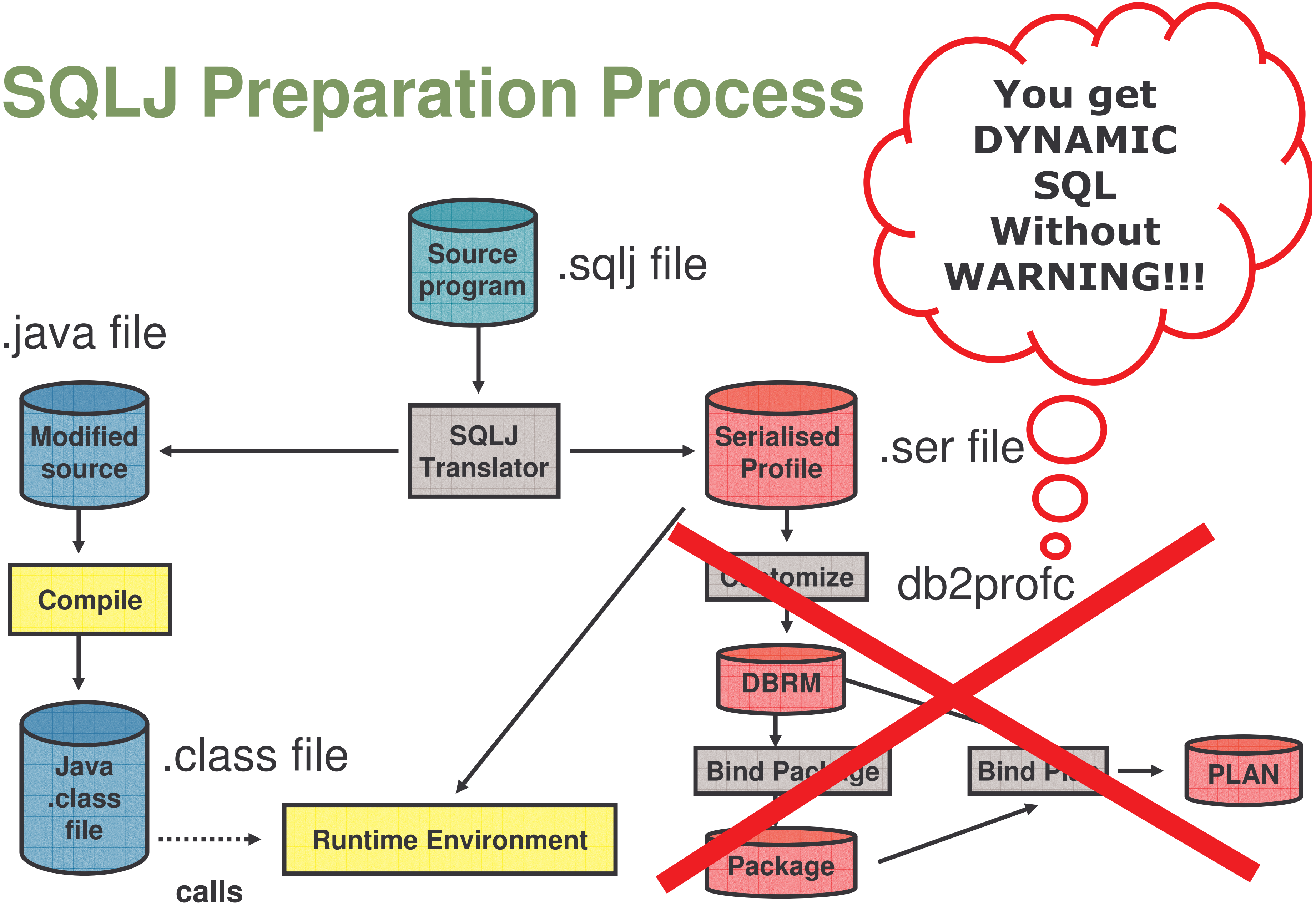
SQLJ

- **SQLJ was designed with specific goals in mind**
  - Provide simple extensions to Java
  - Allow rapid development
  - Easy maintenance of Java apps that use embedded SQL
- **DB2 support and restrictions**
  - DB2 SQLJ support is provided by the DB2 Application Development Client
  - Along with the JDBC support provided, the DB2 client allows you to
    - Create
    - Build
    - Run embedded SQL for Java apps, applets, Stored Procs, UDFs

SQLJ

- **SQLJ provides**
  - **SQLJ translator** – SQLJ replaces embedded SQL stmt in the SQLJ pgm with Java source stmts and generates a serialized profile containing info about the SQL found in the pgm - Sqllib/java/sqlj.zip.file
  - **SQLJ run-time classes** available in sqllib/java/runtime.zip
  - **DB2 SQLJ profile customiser – db2profc** – precompiles the SQL stmts stored in the generated profile and generates a package in the DB2 database
  - **SQLJ profile auditor installer - profdb** – installs/uninstalls debugging class auditors into an existing set of binary profiles. Once installed all *RTStatement* and *RTResultSet* calls made during the application run time are logged to a file. (Only those calls made to the underlying *RTStatement* and *RTResultSet* call interface at run time are audited)
  - **SQLJ profile conversion tool - profconv** – converts a serialised profile instance to class bytecode format. Some browsers do not yet have support for loading a serialised object from a resource file associated with the applet. As a work around you use this util to perform the conversion

SQLJ Preparation Process



SQLJ

- **SQLJ is easier to code**
  - SQLJ is easier to code, read and maintain – because its not an API
  - Coded purely in SQL syntax without the need to wrap SQL in Java methods

- **SQLJ catches errors sooner**
  - JDBC is a pure Java API
    - which means that the Java compiler does not know anything about SQL stmts
    - They only appear as arguments to method calls
    - If one of the stmts is in error, it will not be caught until runtime when the database complains

SQLJ

- **SQLJ allows for better monitoring** -
  - Static SQL pkgs give you the names of the pgms that are running at any given time.
  - Very useful for studying CPU consumption, locking issues (deadlocks etc)

- **SQLJ is better from a security point-of-view**

JDBC vs SQLJ

JDBC

```
java.sql.PreparedStatement ps =

con.prepareStatement("SELECT
BIRTHDATE FROM EMP WHERE
NAME=?");

ps.setString(1, name);

java.sql.ResultSet rs =
ps.executeQuery();

rs.next();

birthdate = rs.getString(1);

rs.close();
```

SQLJ

```
#sql [con] { SELECT
BIRTHDATE INTO :birthdate
FROM EMP

WHERE NAME=:name };
```

JDBC vs SQLJ

JDBC

- More portable
- **Dynamic** SQL only
- SQL prepared at run time
- JDBC manages its own connections to DB2
- Security ("uses userid privileges to access DBMS tables")
- Dynamic SQL statements are cached until they are invalidated

SQLJ

- Easier to code
- **Static** SQL only
- Better performance
- Security ("uses bind owner privileges to access DBMS tables")
- Static SQL is "persistent" ie. They last as long as package exits
- Robustness

- ***SQLJ and JDBC can be used in the same application***
- ***How often will the SQL statement be executed?***

Computer Associates®

Conclusion

Conclusion

- World Wide a very high percentage of new applications are being written in Java
- For most very large companies – the majority of their important data is in DB2
- Java and DB2 - will be the future
  - Understanding of Java and the Web is vital

Bibliograpghy

- **http://java.sun.com/**
- **http://java.sun.com/docs/books/tutorial/**
- **http://java.sun.com/products/ejb/**
- **http://www.ibm.com/developerworks/xml/newto/**
- **http://alphaworks.ibm.com/xml**

- http://www.ibm.com/developerworks/xml/
- http://www.xml.org/
- http://www.xml.com/
- www.w3.org/TR/SOAP
- www.uddi.org

- WebSphere V5.1 Application Developer 5.1.1 Web Services Handbook (SG24-6891-01
- Squeezing the most out of dynamic SQL (SG24-6418-00)
- DB2 UDB e-business Guide (SG24-6539-00)
- DB2 for z/OS and OS/390: Ready for Java (SG24-6435-00)
- Design and Implement Servlets, JSPs and EJBs for IBM WebSphere Application Server (SG24-5754-00)
- Enterprise JavaBeans for z/OS and OS/390 WebSphere Application Server V4.0 (SG24-6283-00)
- Client/Server Survival Guide – 3rd Edition by Orfali, Harkey, Edwards
- Using XML on z/OS and OS/390 for Application Integration (SG24-6285-00)
- DB2 UDB for z/OS and OS/390 V7 Administration Guide
- DB2 UDB for z/OS and OS/390 V8 Administration Guide
- DB2 UDB for z/OS and OS/390 V7 Application Programming and Reference for Java
- DB2 UDB for z/OS and OS/390 V8 Application Programming and Reference for Java
- DB2 UDB for z/OS and OS/390 V7 Application Programming and SQL Guide
- DB2 UDB for z/OS and OS/390 V8 Application Programming and SQL Guide
- DB2 UDB for z/OS and OS/390 V7 XML Extender and Administration Programming

Computer Associates®

Java and the Wild Wild Web
Crash Course No. 1

Maria Sarikos
maria.sarikos@ca.com