

# **WebSphere Summit**

**J2EE vs. .NET**

# Friendly Finance Business Needs

We need to remain competitive by extending our business on the Internet.

We need to improve customer service and satisfaction.

We need to reduce business costs.

It's easy to offer online account management with WebSphere



**IBM**



**Business  
Manager**

# Friendly Finance - I/T Requirements

WebSphere is the best infrastructure for your needs...

It has the best support for standards, has a lower TCO, and WebSphere tools are twice as productive.



**IBM**

We need a strategic infrastructure for our ebusiness strategy.

We need to reduce I/T costs.

We need to respond quickly to new business requirements.



**CIO**

# Why are Open Standards Important?

---

- Consistency
- Avoid vendor lock-in
- Stability protects customer investments
- Skill availability
- Expert driven

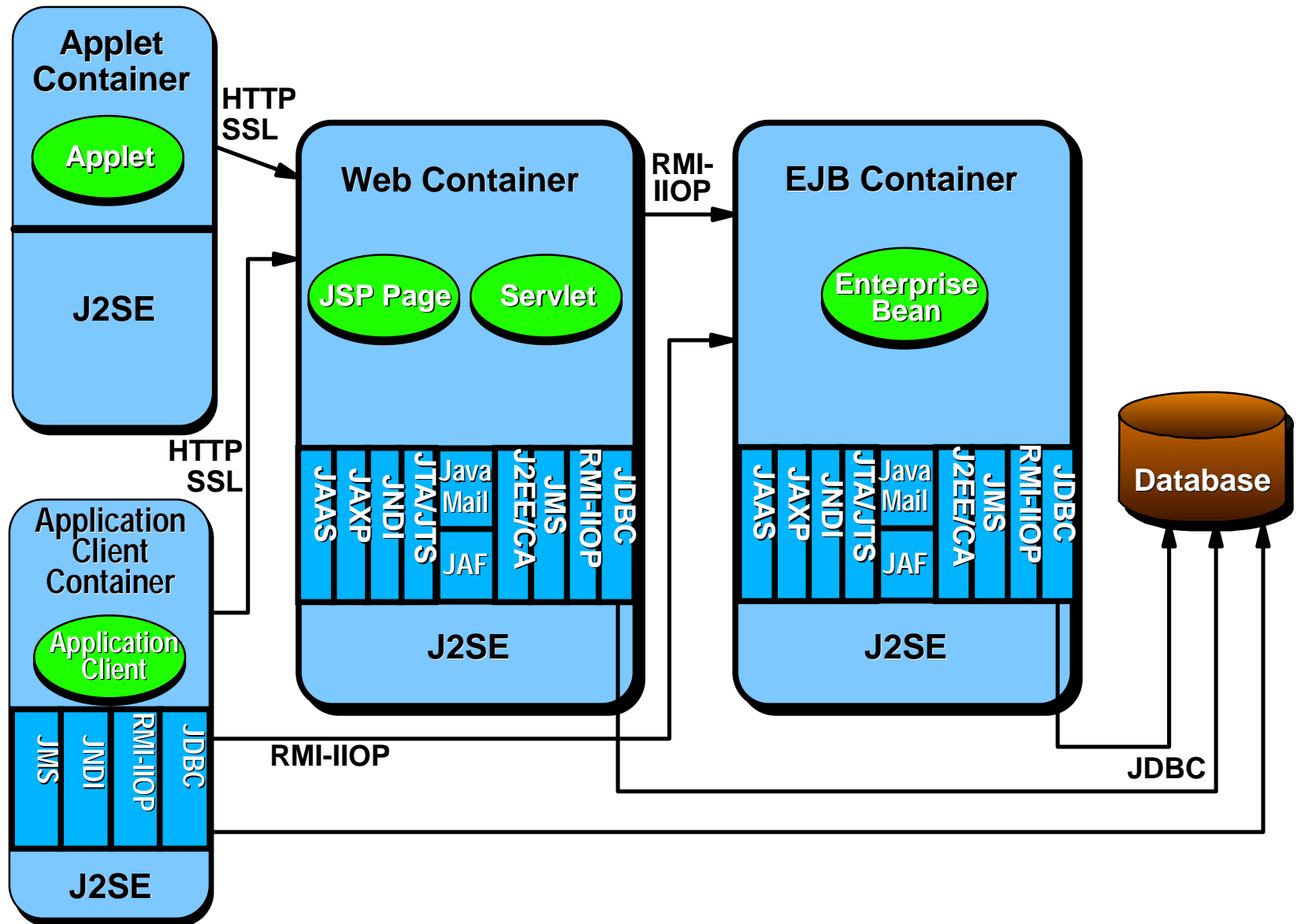
Lower costs, faster to market

# What is J2EE?

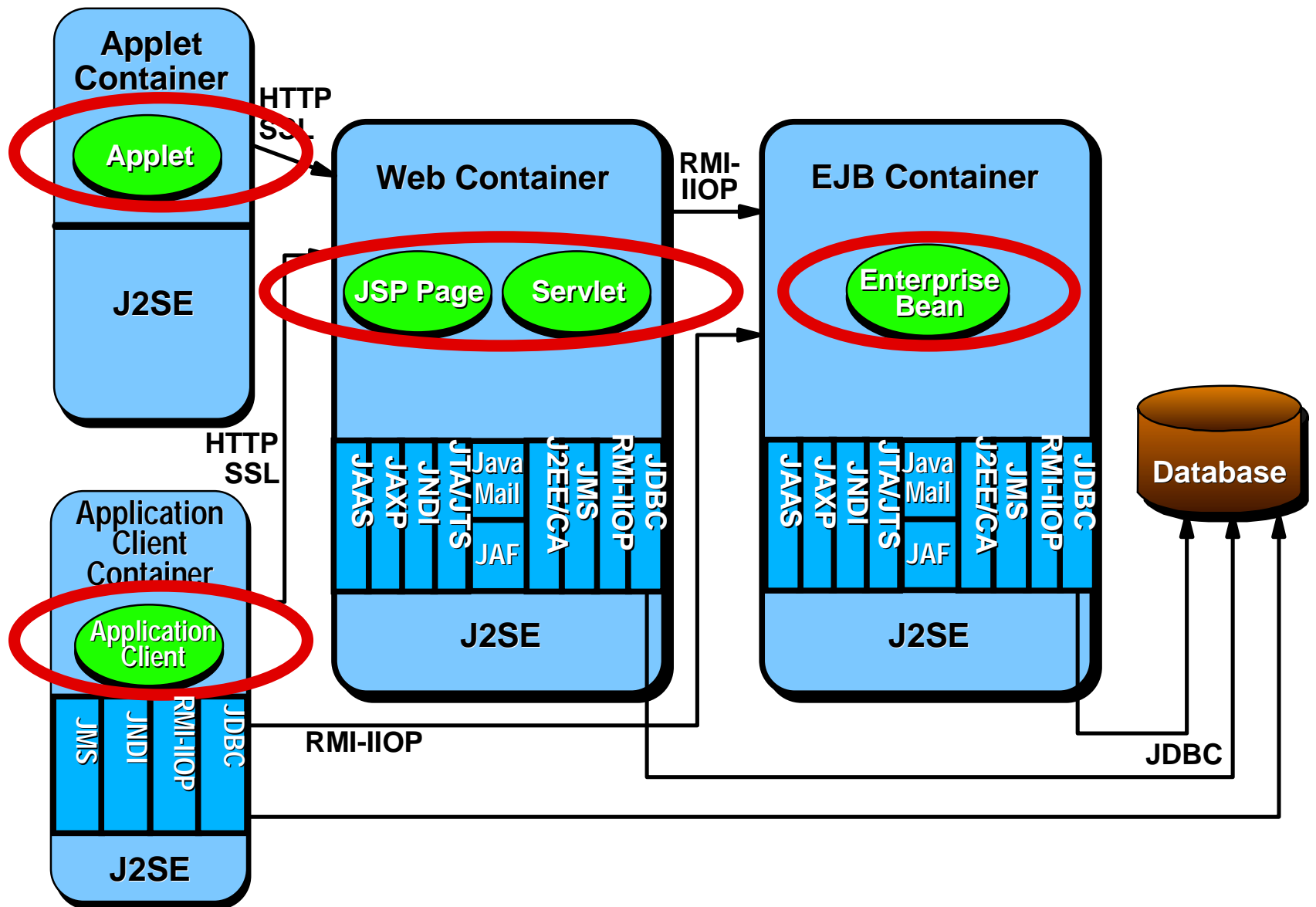
---

- Java 2 Platform, Enterprise Edition (J2EE) defines a Java platform for enterprise computing
  - ▶ Multi-tier, distributed applications
- J2EE defines a standard architecture that includes the following:
  - J2EE Platform Specification -- defines the specific Java functionality (APIs) that must be supported to host J2EE applications
  - J2EE Application Model -- shows how to design multi-tier, distributed, thin-client applications
  - J2EE Compatibility Test Suite -- used by vendors to verify that their application server complies with the J2EE specification
  - J2EE Reference Implementation -- a no-value-added example application server complying with a J2EE platform specification

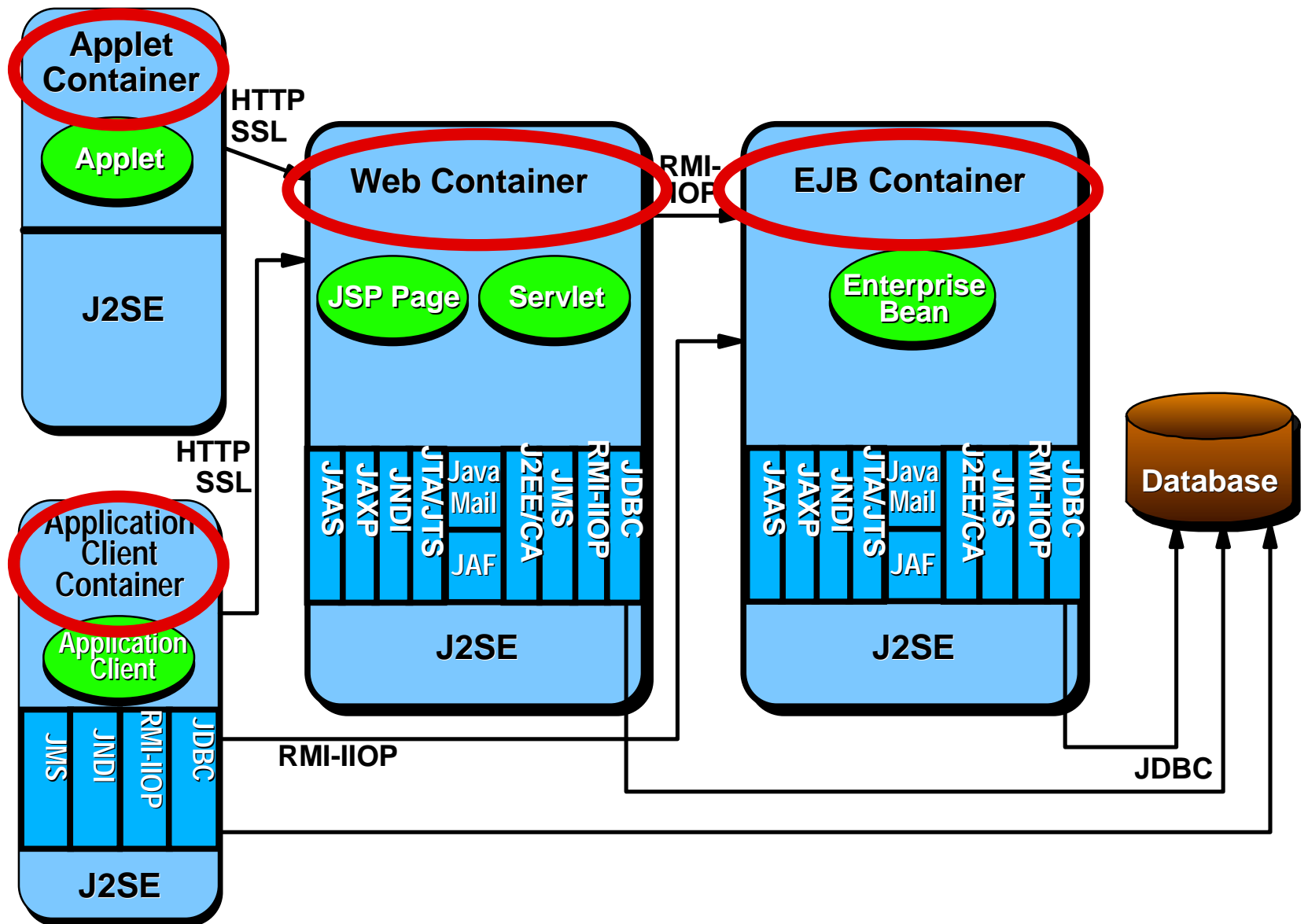
# J2EE Platform



# J2EE Application Components

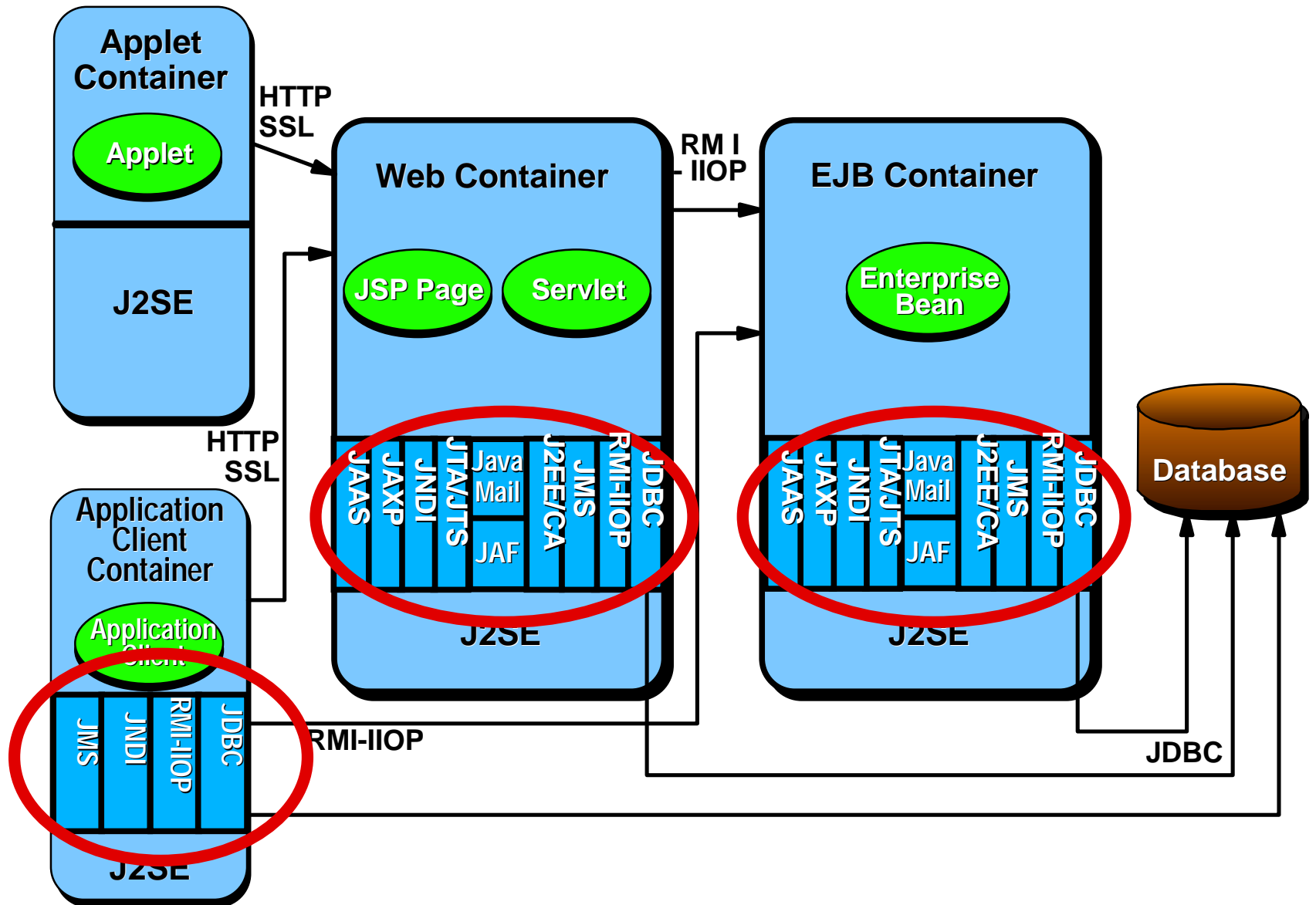


# J2EE Containers





# J2EE Standard Services



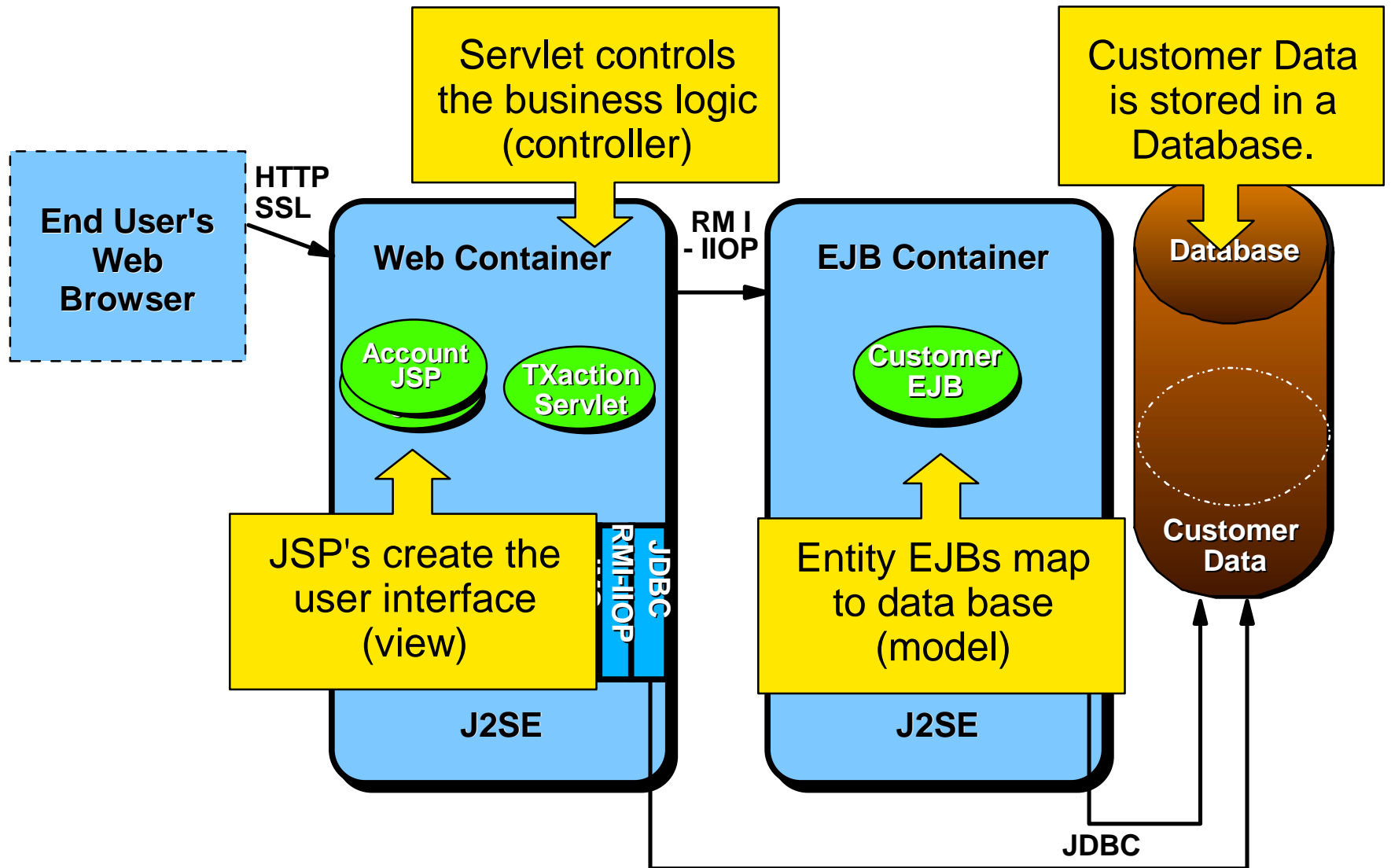
# J2EE Standards

		J2EE 1.2	J2EE 1.3
J2SE	formerly known as JDK	1.2	1.3
EJB	distributed processing	1.1	2.0
Servlets	servlets	2.2	2.3
JSP	Java server pages	1.1	1.2
JDBC	access data base	2.0	2.0
JNDI	directory	1.2	1.2
JMS	messages	1.0	1.0
JTA/JTS	transactions API and services	1.0	1.0
JavaMail	mail	1.1	1.2
JAF	application framework	1.0	1.0
RMI/IIOP	remote method invocation	1.0	1.0
JAXP	XML parsing API	NA	1.1
JAAS	authentication	NA	1.0
Connector	connector framework	NA	1.0

WAS 4.0 Certified to Support J2EE 1.2

WebSphere Technology for Developers 5.0  
Certified to Support J2EE 1.3

# Example Server based J2EE application



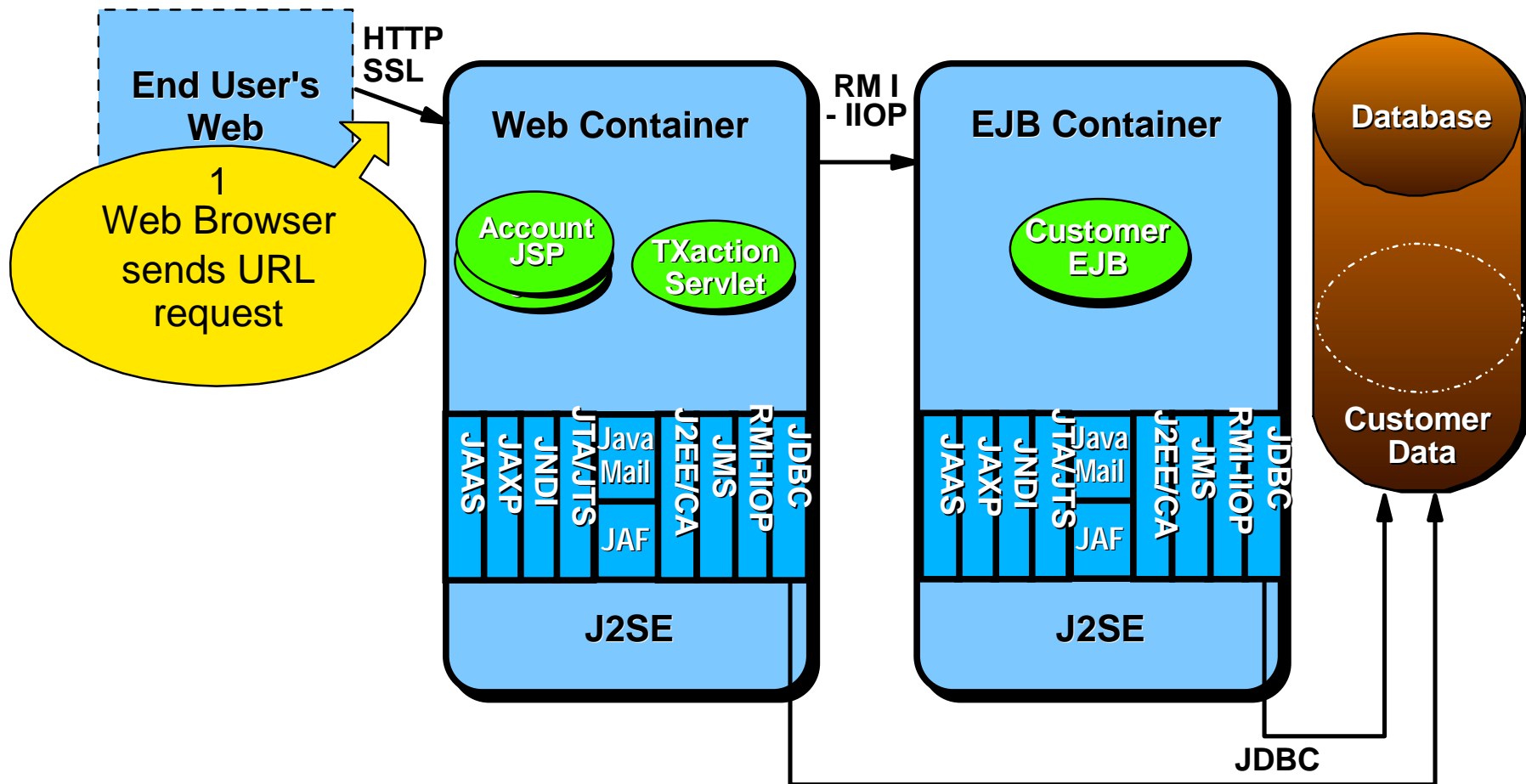
# DEMO

## Introduction to Friendly Finance

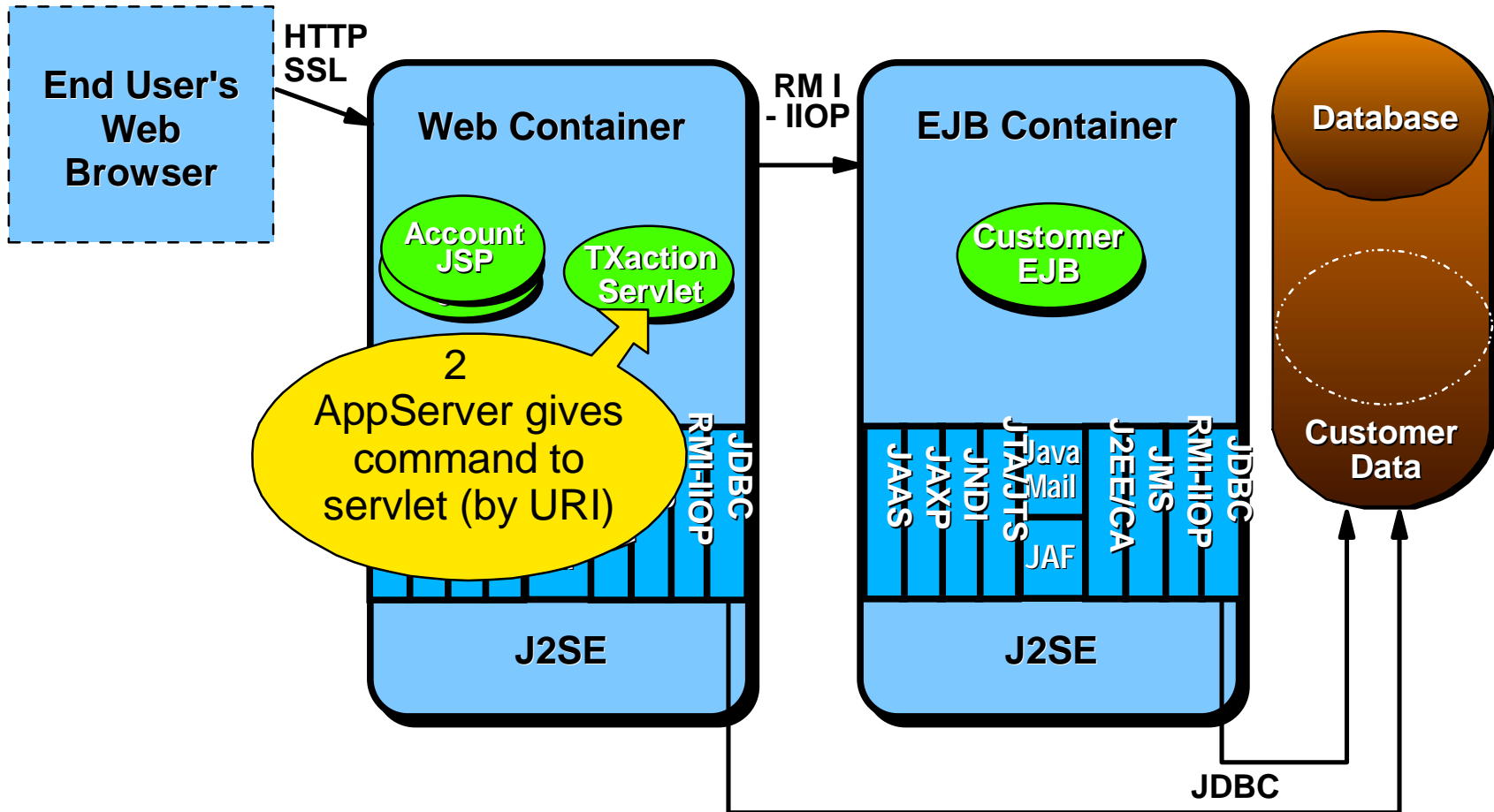
---

- A look at the web site
- Customer service functions
  - ▶ Log in
  - ▶ Account Management

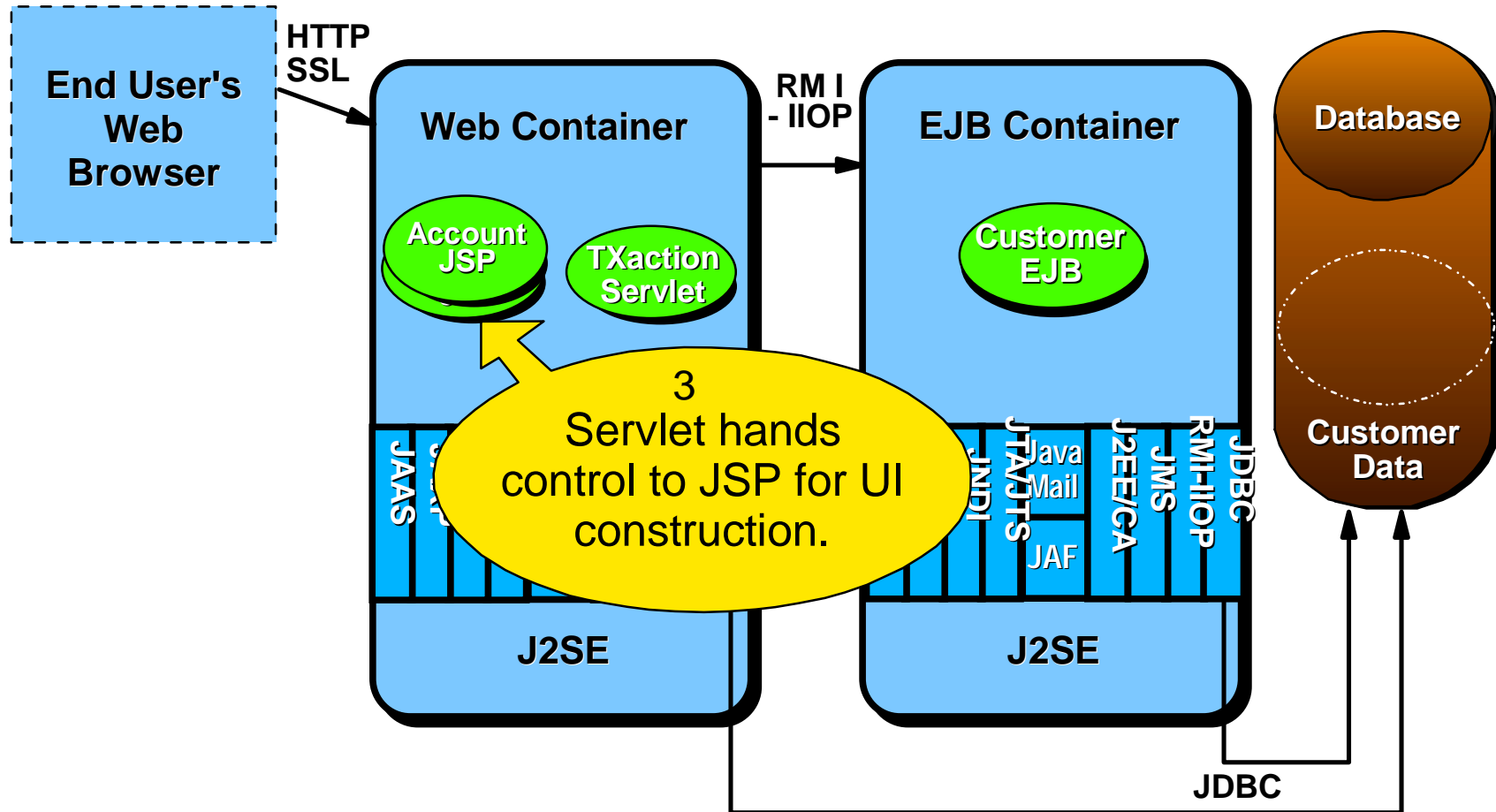
# Example flow of execution



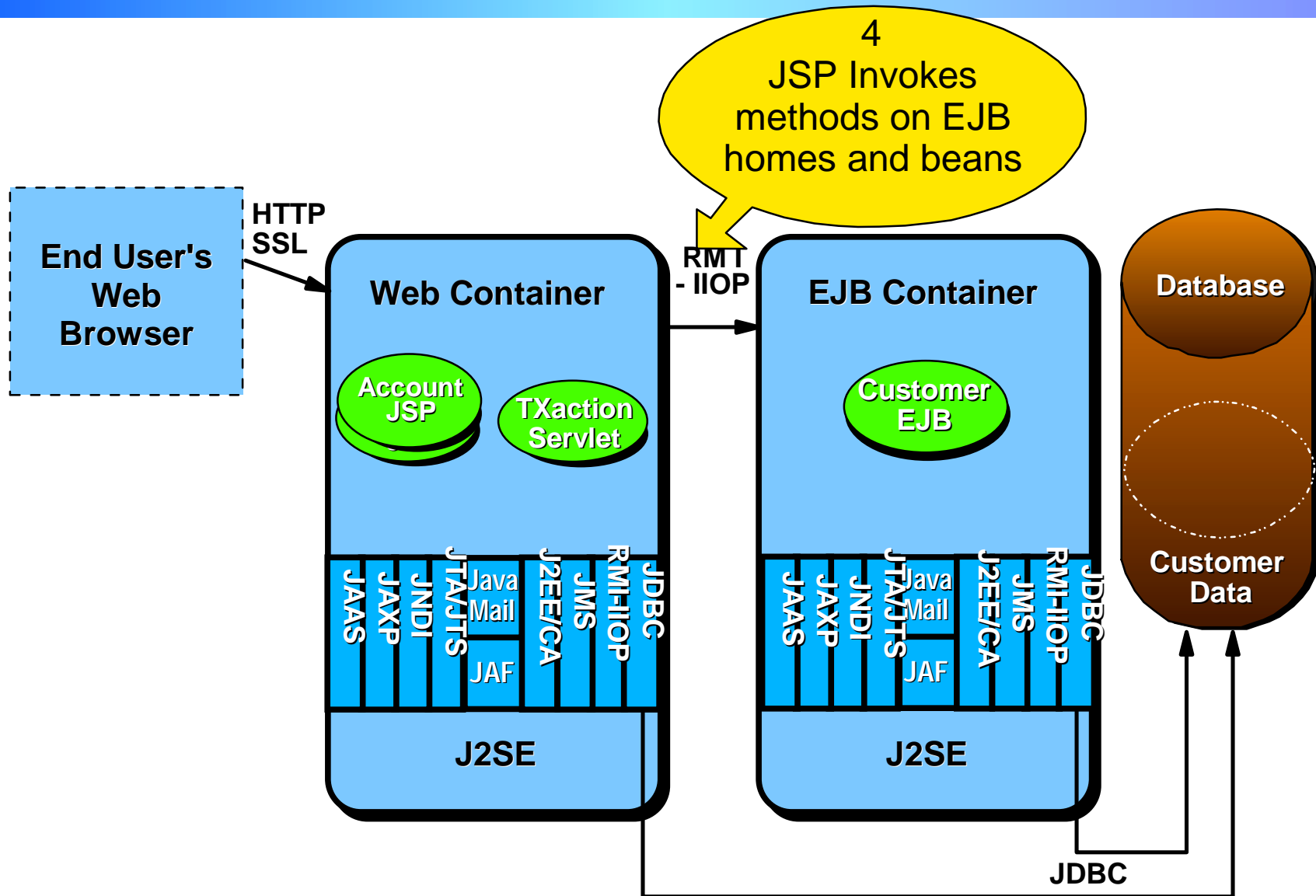
# Example flow of execution



# Example flow of execution

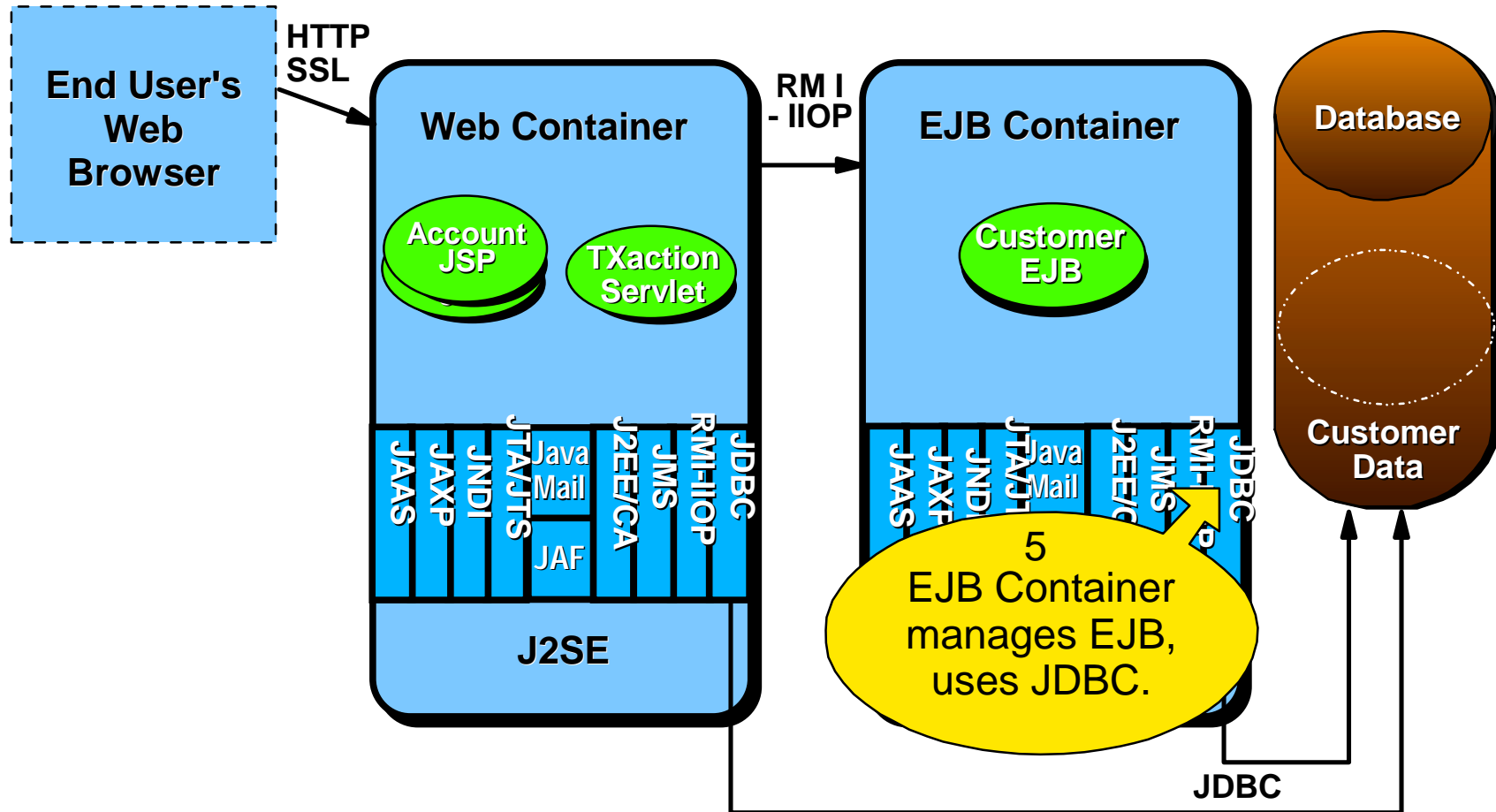


# Example flow of execution

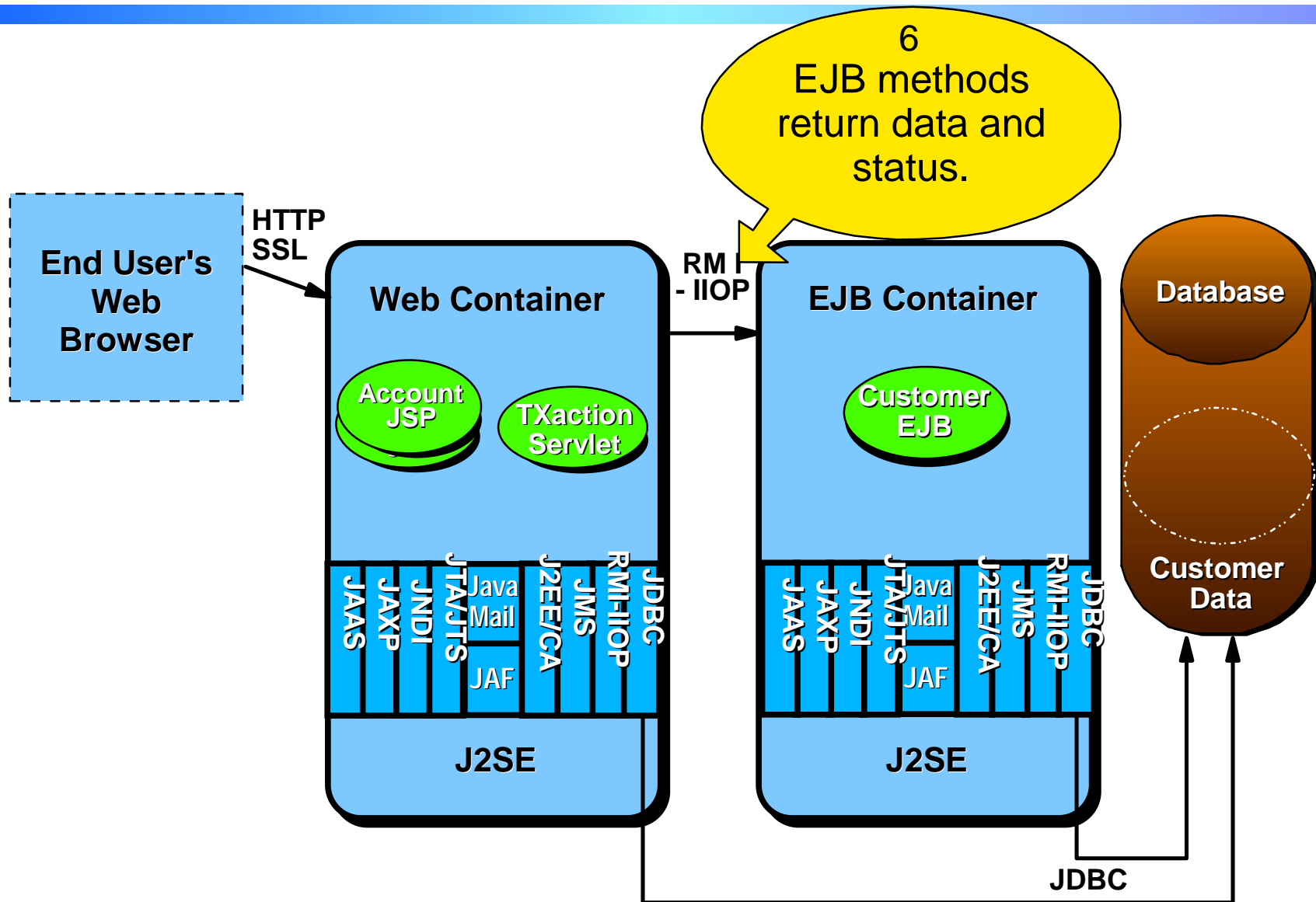




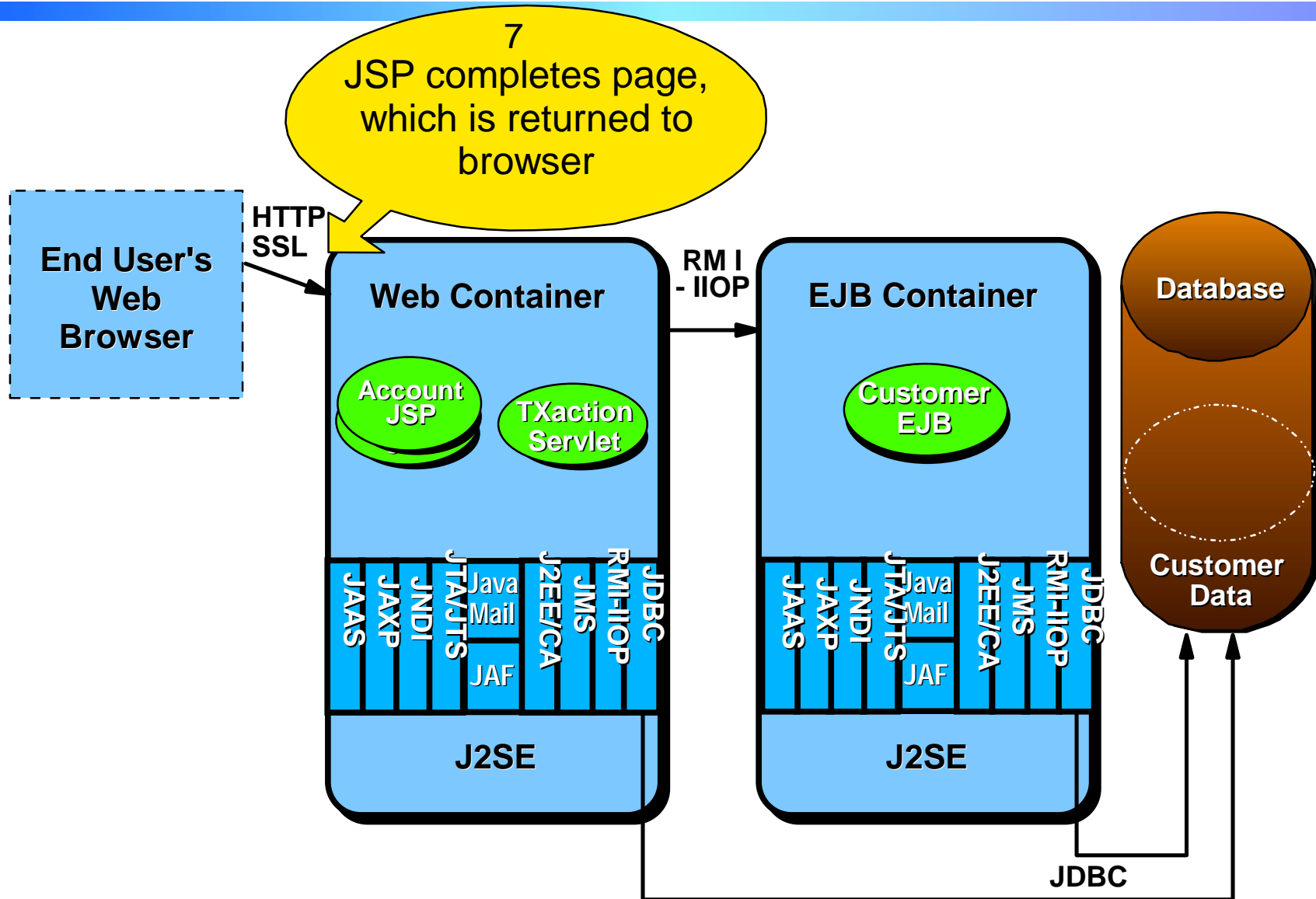
# Example flow of execution



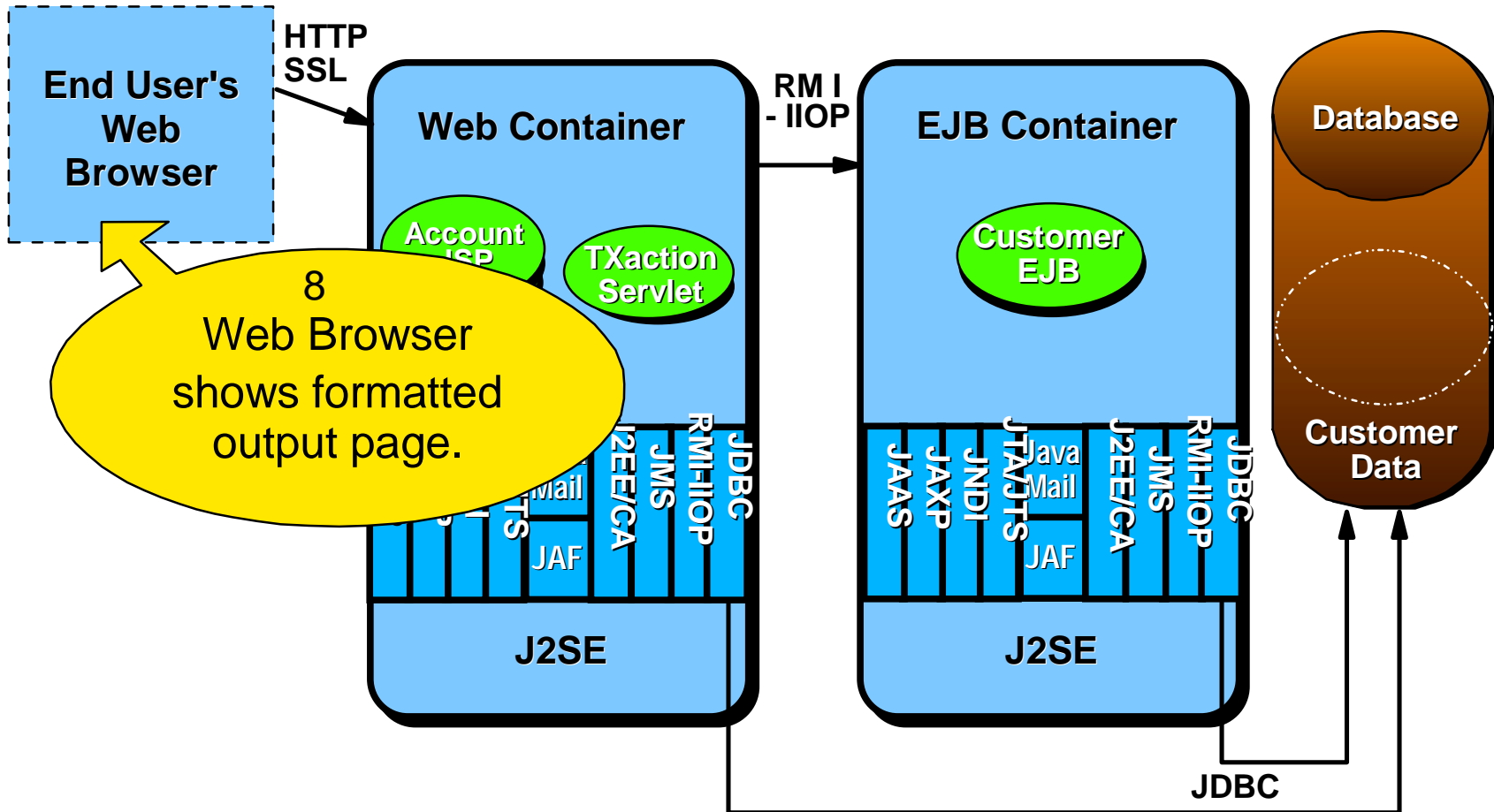
# Example flow of execution



# Example flow of execution



# Example flow of execution



# How does Microsoft compare to J2EE?

---

## ■ .NET

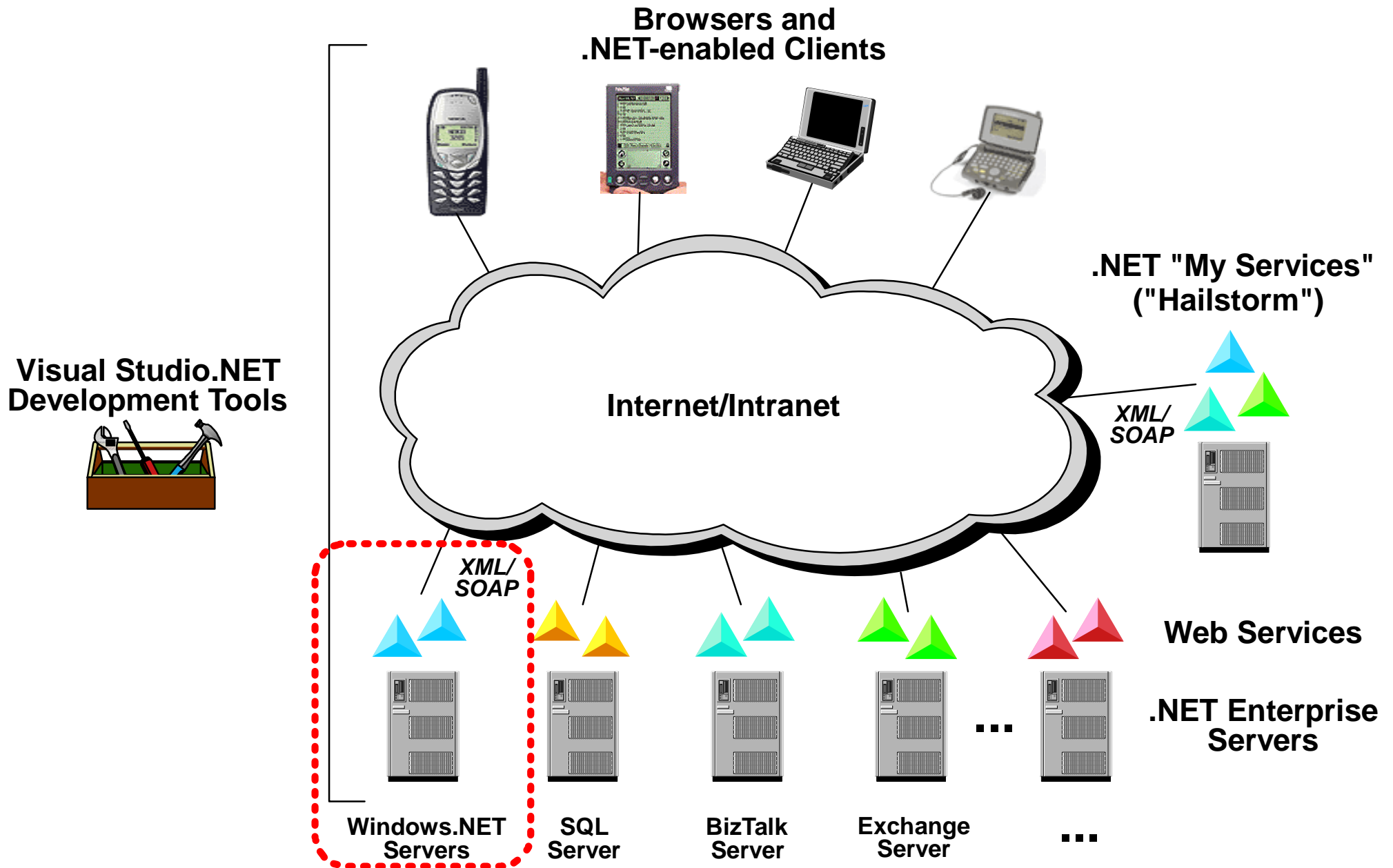
- ▶ Announced in 2000
  - Microsoft messages increase confusion in the industry
- ▶ Early announcement intended to slow down J2EE momentum
- ▶ Few products are currently .NET- enabled
  - Visual Studio.NET development environment just released (mid - 2/2002)

# .NET's Primary Thrusts

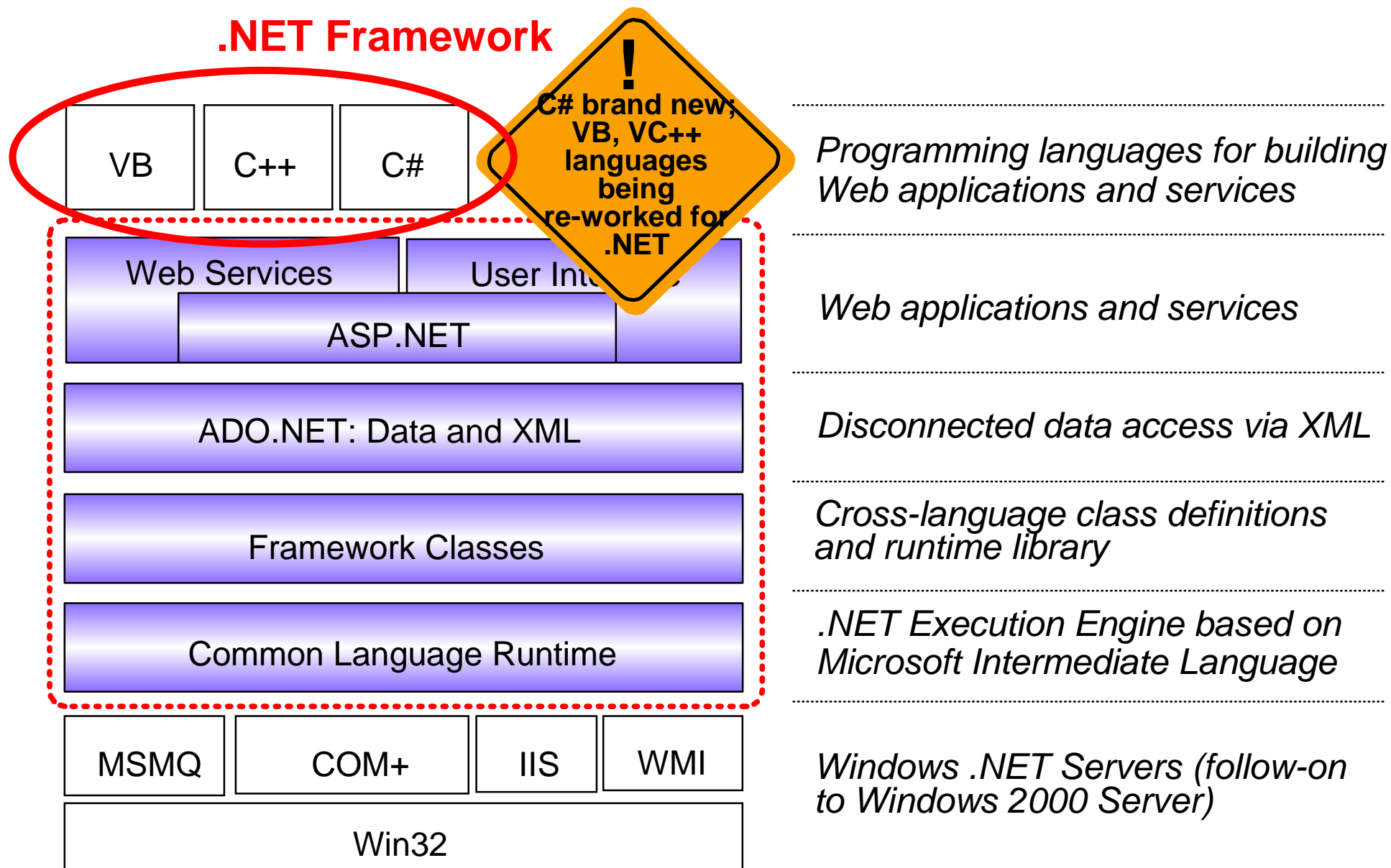
---

- Address runtime deficiencies vis-a-vis Java and J2EE
  - ▶ performance and scalability
  - ▶ defuse cross-platform problem
  - ▶ defuse programming language interoperability problem
- Web Services computing model
  - ▶ new orchestration role for heavy client
  - ▶ counteract "thin" client strategies
  - ▶ subscription opportunities

# Microsoft's .NET Vision



# .NET Requires Brand New Development and Execution Environment





# Visual Basic.NET Requires Extensive Re-Training

## Page from MS Web

Microsoft Visual Basic: Visual Basic 6.0 Upgrade to Visual Basic .NET - Microsoft Internet Explorer

Address <http://msdn.microsoft.com/vbasic/technical/upgrade/transition/language.asp>

### The Transition from Visual Basic 6.0 to Visual Basic .NET

#### Language

In Visual Basic .NET, the Microsoft Visual Basic language has been streamlined and modernized—removing legacy keywords, improving type safety, and exposing low-level constructs that advanced developers require.

To deliver these features, a number of syntax changes have been made to the language. These changes, as well as their upgrade paths, are outlined below:

<a href="#">Arrays</a>	<a href="#">Fixed-Length Strings</a>	<a href="#">Parameter ByVal/ByRef Default</a>
<a href="#">As Any Parameters in Declares</a>	<a href="#">GoSub...Return</a>	<a href="#">Property</a>
<a href="#">Assignment</a>	<a href="#">Implements</a>	<a href="#">ReDim</a>
<a href="#">Calling Procedures</a>	<a href="#">Integer &amp; Long</a>	<a href="#">Resource Files</a>
<a href="#">Currency</a>	<a href="#">IsMissing and Optional Parameters</a>	<a href="#">Static Procedures</a>
<a href="#">Date</a>	<a href="#">Local Variables Inside Blocks</a>	<a href="#">Type</a>
<a href="#">Debug.Print</a>	<a href="#">LSet</a>	<a href="#">User-Defined Type Storage</a>
<a href="#">Def&lt; type &gt;</a>	<a href="#">New Auto-Reinstantiation</a>	<a href="#">Variant</a>
<a href="#">Default Properties</a>	<a href="#">Null and Null Propagation</a>	<a href="#">VarPtr, StrPtr, and ObjPtr</a>
<a href="#">Empty</a>	<a href="#">Object Finalization</a>	<a href="#">While</a>
<a href="#">Enumerations</a>	<a href="#">On...GoTo and On...GoSub</a>	
<a href="#">File I/O</a>	<a href="#">ParamArray Parameters</a>	

*significant changes to syntax and semantics*

Last Updated: 4/4/01

# VB6 to VB .NET : Sweeping language changes

---

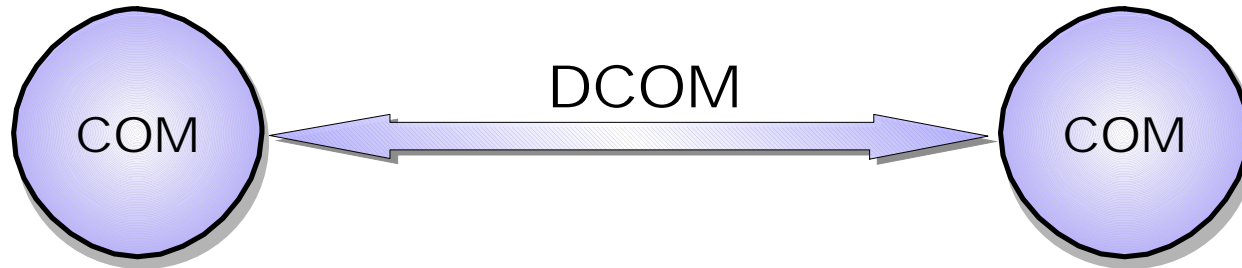
- Arrays
  - ▶ all arrays are now zero-based.
  - ▶ no more provision for explicit array bounds
- Variable scope within blocks
  - ▶ Variable declared inside a block is not accessible outside block now
- Parameter passing mechanism
  - ▶ Default in VB6 was ByRef, now it is ByVal
- ...and **many many** more!!
  
- Migration could require careful scrutiny and manual changes to avoid logical bugs!

# Visual Studio.NET

---

- Visual Studio.NET just released (mid-February 2002)
  - ▶ originally due for general availability in the second half of 2001
  - ▶ Microsoft suffered a serious set back with the first Beta release of Visual Studio.NET when developers were disgruntled with the changes made to Visual Basic.

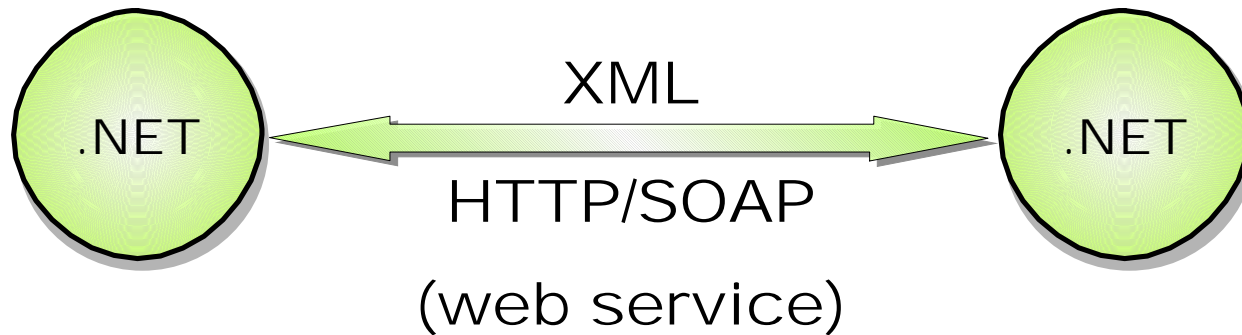
# More Fundamental Problems. . . Two Object Models



**Has difficulty dealing with non-Windows platforms**

**does not go through firewalls**

**Not scalable**



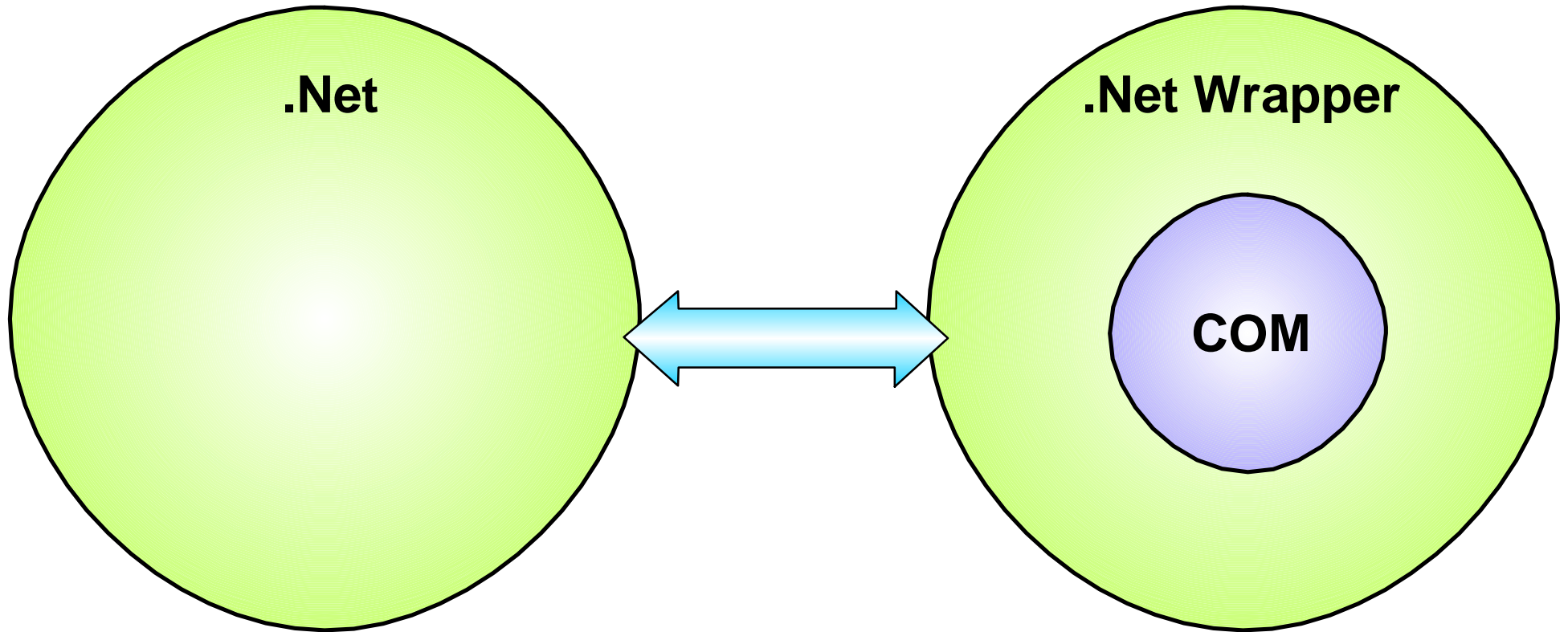
**Better interoperability**

**goes through firewalls**

**Scalable**

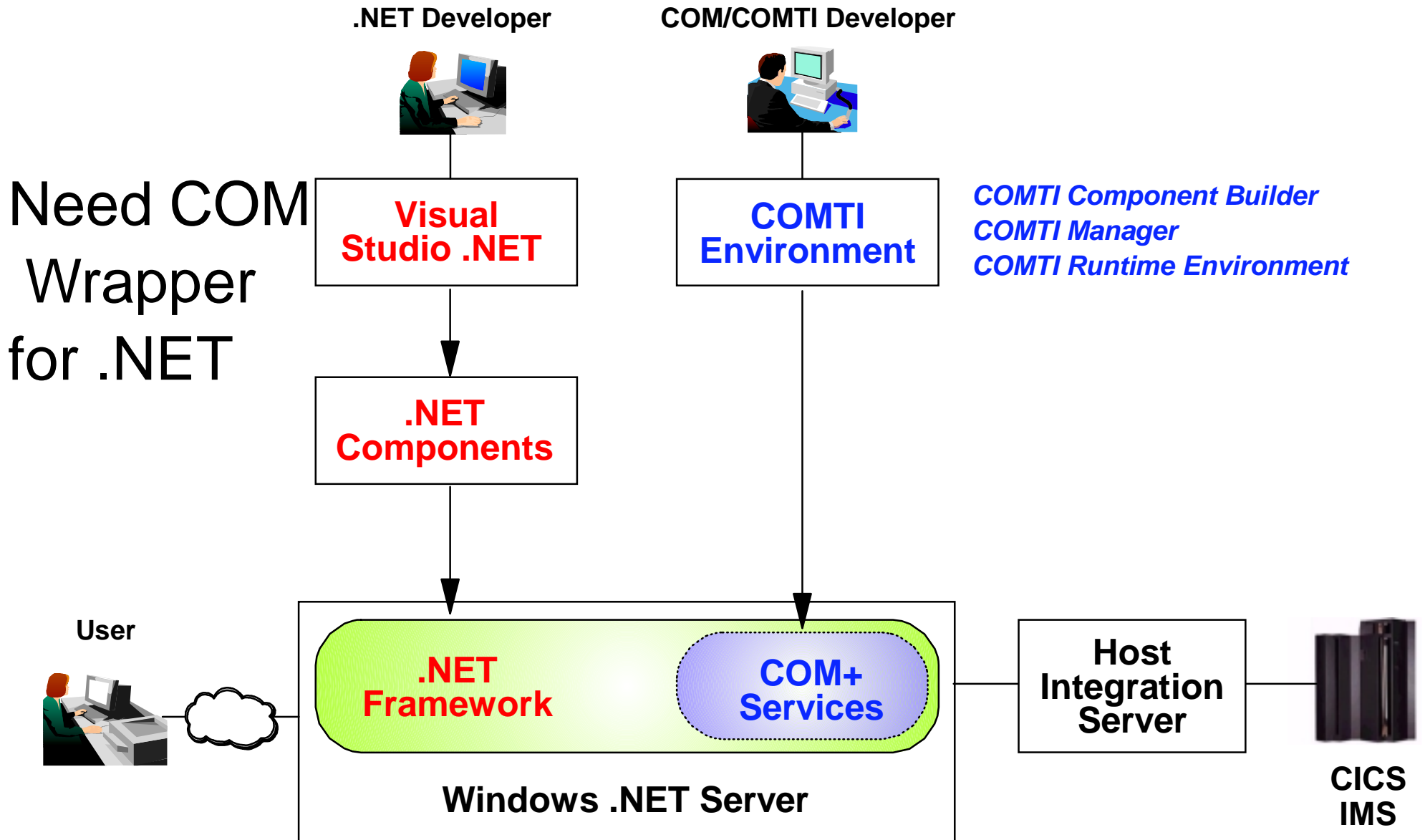
# COM Objects Must be Wrapped to be Used by .NET Objects

---



- Tooling support?
- Performance?
- Debugging?

# Developers Must Deal with Both Object Models When Dealing with .NET and Enterprise Integration



# .NET Programming Model Borrows Heavily from J2EE, But Falls Short of Completeness, Maturity

J2EE is 20-40% more productive<sup>1</sup>



Java Foundation Classes (Swing)

Java WebStart

Java Server Pages (JSP)

Business Transaction Components:

*EJB Session Beans*

Business Object Components:

*EJB Entity Beans*

Message Queue Components:

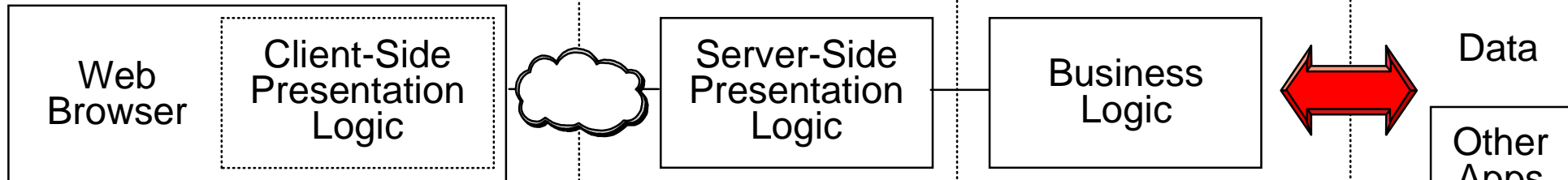
*EJB Message-Driven Beans*

Data Access:

*JDBC*

Application Access:

*J2EE CA JMS*



Microsoft Foundation Classes

Windows Forms

Active Server Pages (ASP.NET)

Business Transaction Components:

*COM+/.NET Framework*

Business Object Components:

*none*

Message Queue Components:

*COM+ Queued Components/.NET Framework*

Data Access:

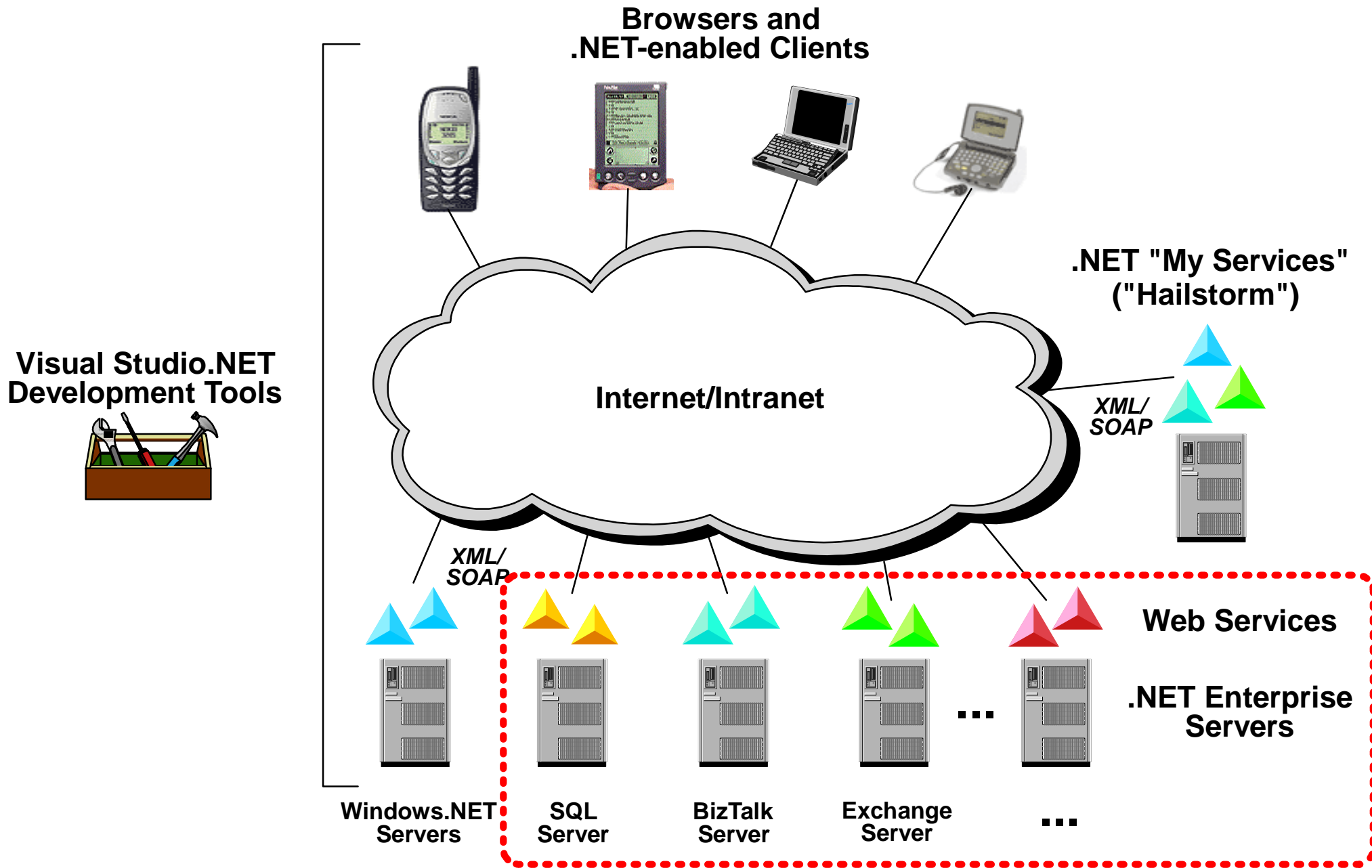
*Active Data Objects (ADO).NET*

Application Access:

*none*

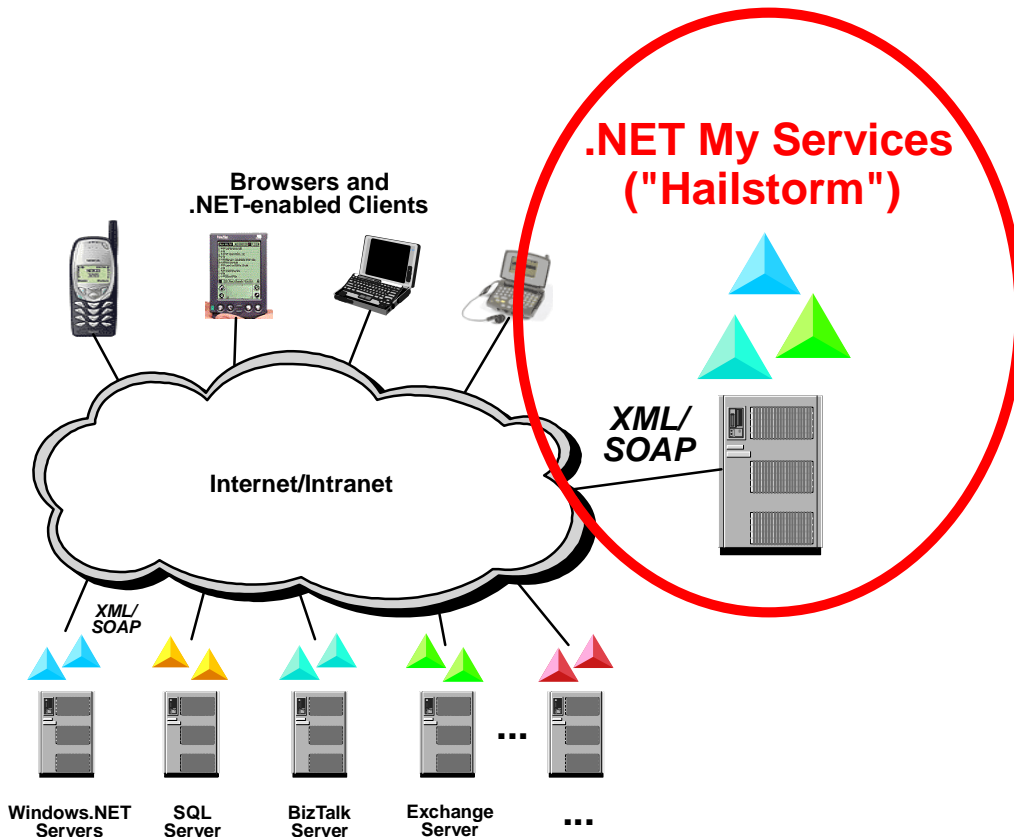
<sup>1</sup> Giga Information Group, "J2EE vs. Microsoft and .NET: The Big Application Platform Battle", 9/18/2001

# A Closer Look at the .NET Enterprise Servers





# .NET "My Services" Introduces Development and Deployment Complexity



- .NET My Services is a set of Microsoft- provided, user-centric XML-based web services
  - ▶ hosted, remote .NET objects
  - ▶ can be used individually or combined
  - ▶ uses Passport for authentication
- Can also be thought of as a set of application building blocks that developers use to build their own XML-based web services offerings
- User subscription fees
- This approach will make all your applications as reliable as the internet

**.NET Profile**

**.NET ApplicationSettings**

**.NET Devices**

**.NET Documents**

**.NET FavoriteWebSites**

**.NET Presence**

**.NET Wallet**

**.NET Alerts**

**.NET Calendar**

**.NET Contacts**

**.NET Inbox**

**.NET Lists**

**.NET Location**

**.NET Categories**

# .NET Vision

---

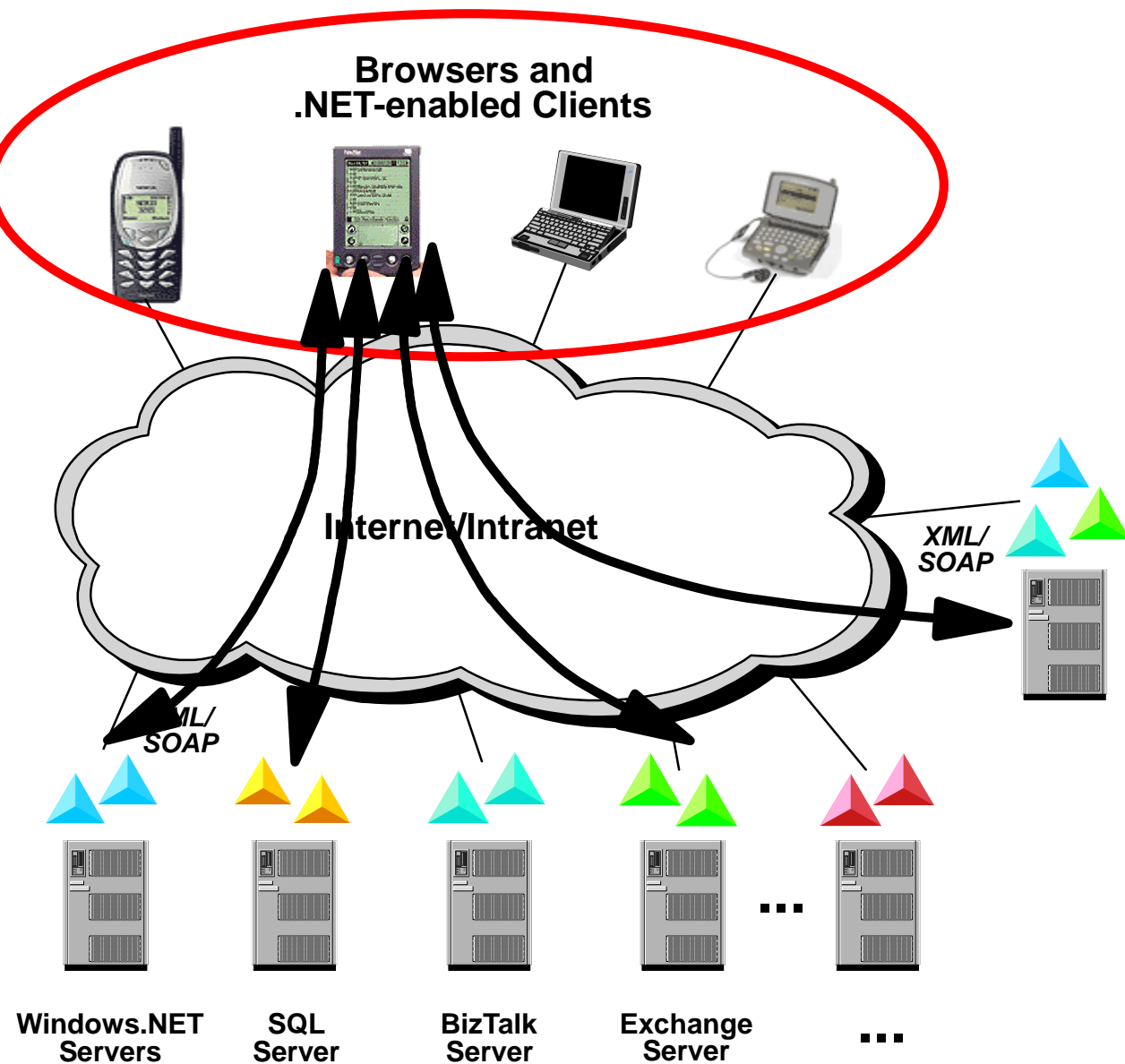
- Future computer applications will be composed of **Web Services** running on different computers within a globally distributed network, and which communicate with one another using XML messages transmitted in accordance with the SOAP protocol.

# .NET Vision

---

- Future computer applications will be composed of **remote objects** running on different computers within a globally distributed network, and which communicate with one another using XML messages transmitted in accordance with the SOAP protocol.

# .NET-Enabled Clients: The New "Heavy" Client



- .Net-enabled clients act as application "orchestrator" by invoking multiple remote web services
- Fine-grained interaction with remote web services will not perform well

# Microsoft's Key Message: Protecting the desktop

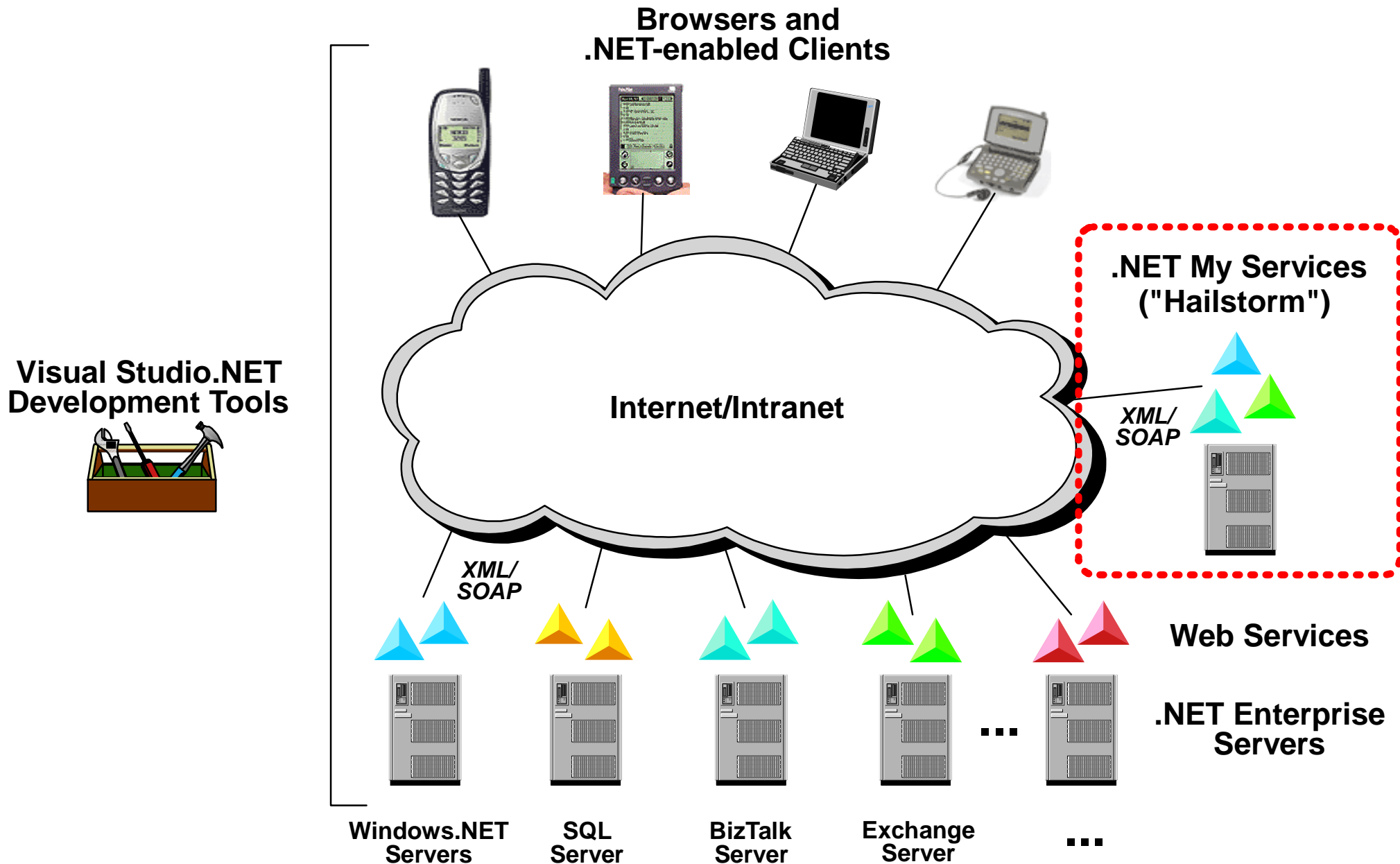
---

*"If you simply have a browser, you can connect up and get to the .NET services. But it's **far richer** if you have a device that actually has the .NET code down on it."--Bill Gates*

# .NET Enterprise Servers: New Name, Same Old Technology

.NET Enterprise Server	Availability	.NET-enabled?
<i>Application Center 2000</i>	now	N/A
<i>BizTalk Server 2002</i>	beta	yes
<i>Commerce Server 2002</i>	beta	yes
<i>Exchange Server 2000</i>	now	no
<i>Content Management Server 2000</i>	now	no
<i>Host Integration Server 2000</i>	now	no
<i>Internet Security and Accelerator Server 2000</i>	now	no
<i>Mobile Information Server 2001</i>	now	no
<i>SharePoint Portal Server 2001</i>	now	no
<i>SQL Server 2000</i>	now	yes

# A Closer Look at the .NET My Services



# Other Analyst Views Favor J2EE over .NET As Well

**American Banker (January 11, 2002):**

**"For A Platform Edge, Java Is Better Choice Than .NET"**

*"...The Gartner Inc. consulting firm estimates that by next year more than 45% of integrated-channel delivery products from top-tier banks will be based on J2EE and its components, Enterprise Java Beans (EJBs)..."*

---

*"...A report from Aberdeen Group says that use of client-side Java results in a 10% to 20% faster time to market, and a 10% to 20% improvement in application quality, both of which result from increased customer choice..."*

---

*"... "With a wider range of application vendors, financial institutions and other companies can select the application or applications that fit their particular needs, rather than settle for inadequate solutions based on a single Microsoft platform. The Microsoft platform -- often faulty and subject to system crashes -- requires a specific middleware application, so applications based on it cannot be ported to a better middleware solution when one comes on the market. Banks can run J2EE on a dozen different operating systems, but .Net is only compatible with Microsoft applications and servers."..."*



# .NET Platform Exposures

---

- Still limited to Microsoft platforms and Intel machines
- Tied to single vendor
  - ▶ C# and .NET Framework are proprietary
- .NET Framework packaged in Windows .NET Servers
  - ▶ OS upgrade may be required to obtain .NET enhancements
- New, untested technology
  - ▶ Only beta-level code available
    - Java code is running on over 7 million Internet sites today
  - ▶ Unix still more stable than Windows 2000
- Language transition
  - ▶ Visual Basic - many syntax and semantic changes
    - Initial problems with Betas 1 & 2 have upset many developers
  - ▶ Visual C++ - must give up much functionality to participate

# .NET Platform Exposures (continued)

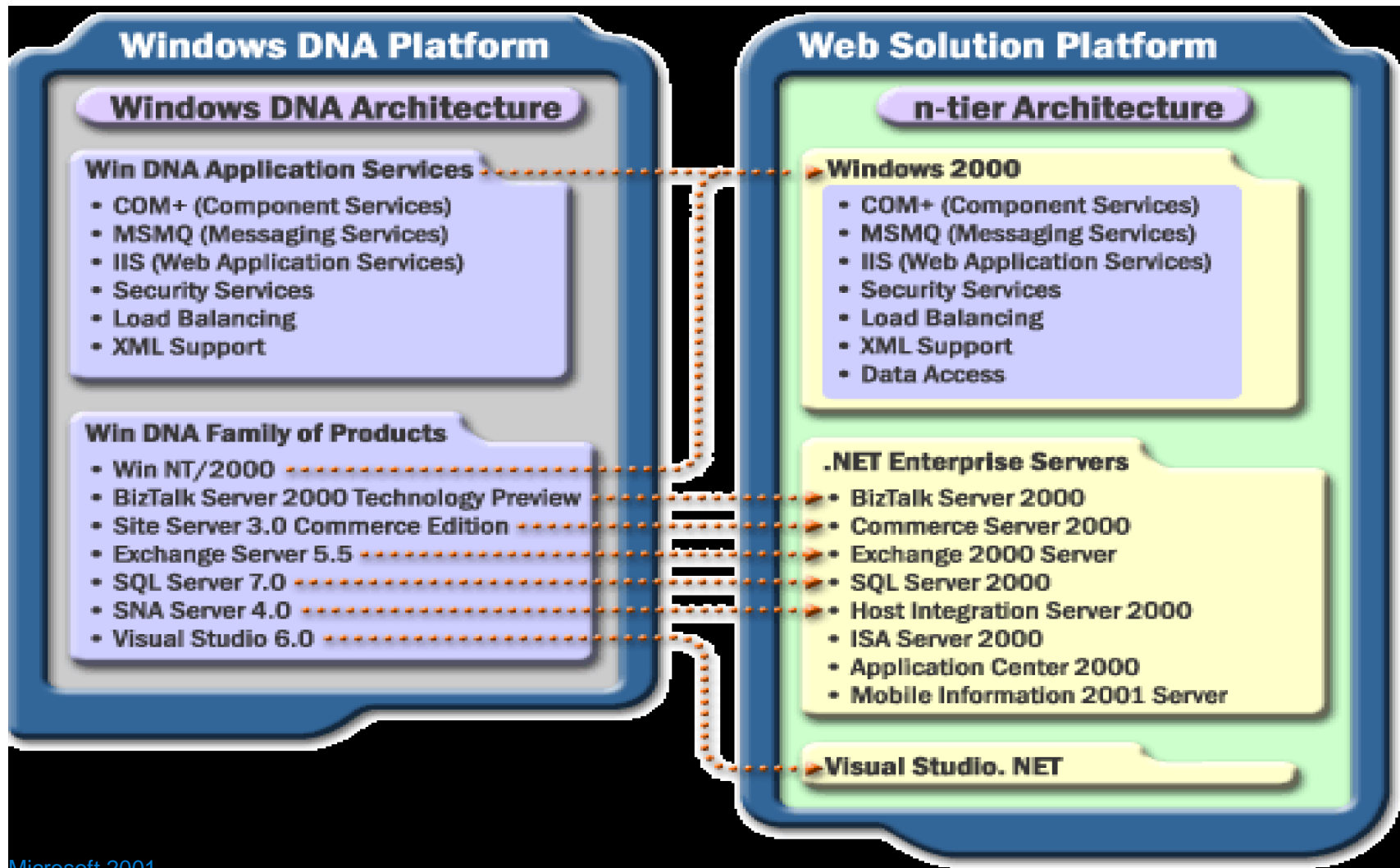
---

- Two different object models
  - ▶ COM and .NET
  - ▶ Development effort required to use COM components within .NET
  - ▶ Inherent performance problems when switching between COM and .NET environments
- Heavy-client orchestration won't perform well across Internet
- Web Services Tooling only addresses simple tasks
  - ▶ Microsoft tooling requires source code to create a web service - IBM tools do not
  - ▶ Microsoft tooling cannot turn COM objects into web services - IBM tools support several artifacts
  - ▶ Wizards are good for the most simple case, not very flexible
  - ▶ No public UDDI support - does not publish to public UDDI registries.
- No test environment in Visual Studio.NET
  - ▶ Everything deployed to local IIS.

---

***Thank you***

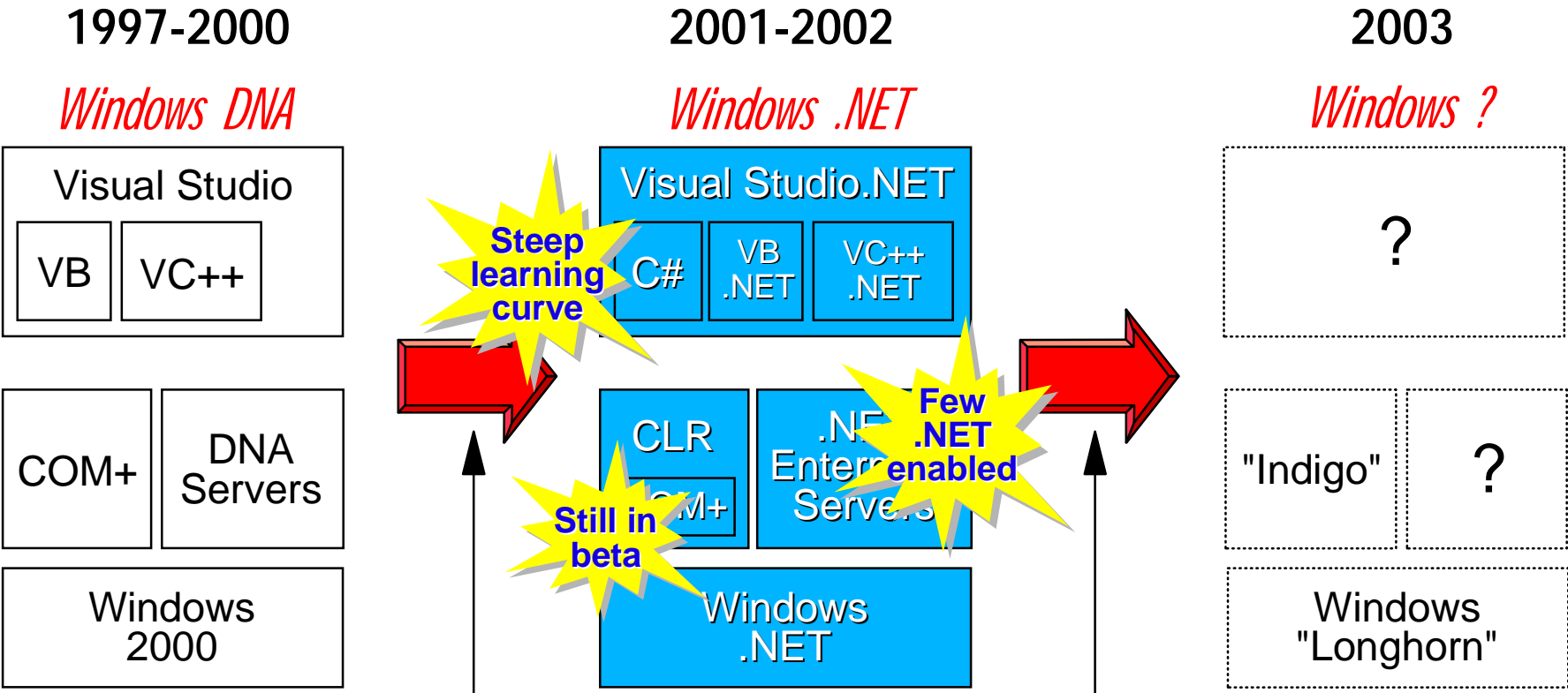
# New name for old technology: Moving away from DNA



# J2EE vs. .NET Summary

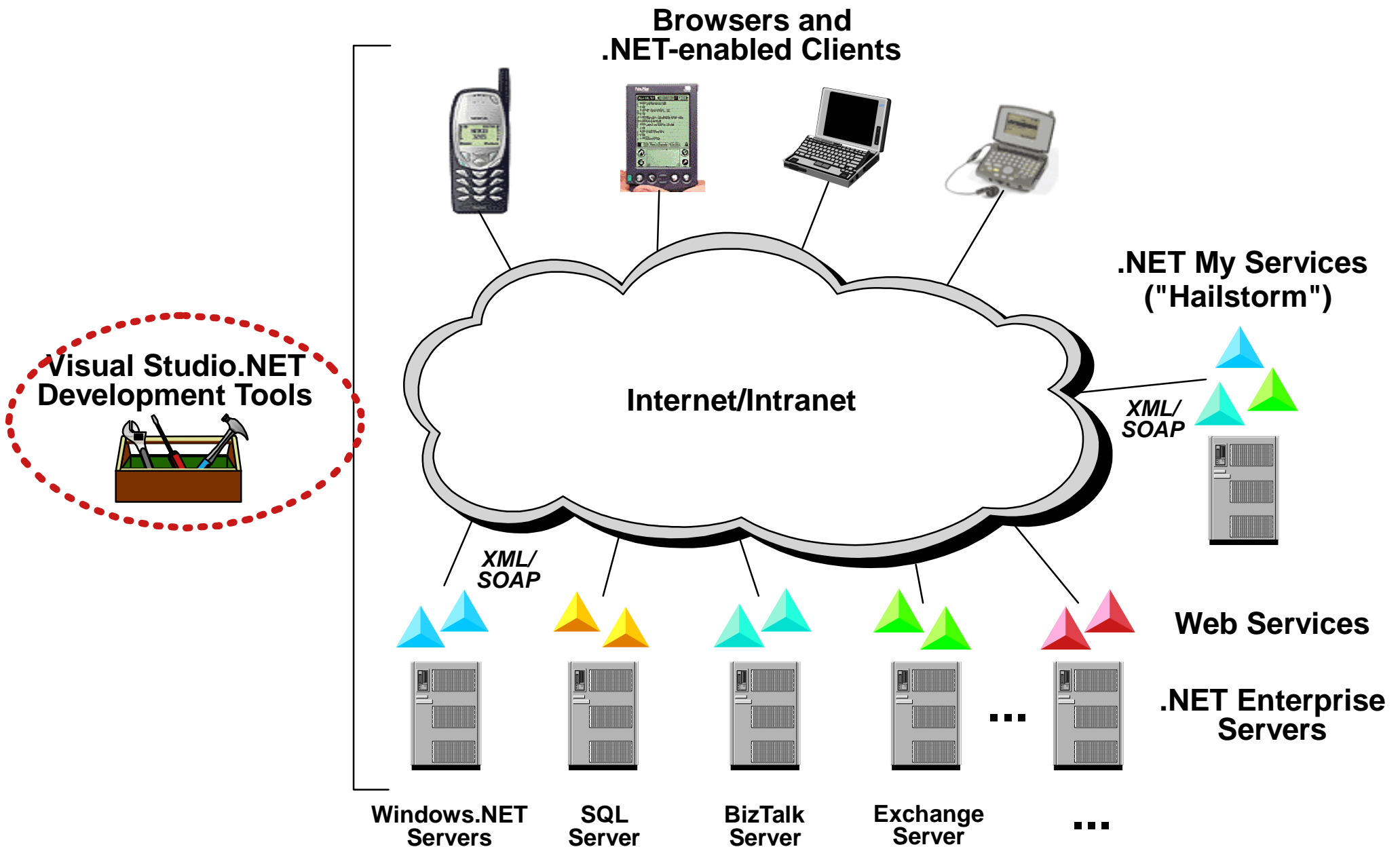
Features	J2EE	.NET
<b><i>Standards-based architecture</i></b>	yes	no (proprietary)
<b><i>Mature architecture</i></b>	yes	no (still under development)
<b><i>Common, consistent object model</i></b>	yes	no (COM and .NET objects)
<b><i>Multi-platform support</i></b>	yes	no (Windows only)
<b><i>Skills availability</i></b>	widespread	few (C#, VB.NET brand new)
<b><i>Proven track record</i></b>	yes	no (still in beta)

# Microsoft .NET is New, Unfinished, Unproven, and Carries Higher Risk



**COM+ not part of CLR**

# A Closer Look at Visual Studio.NET



# .NET Servers

---

***"Windows .NET Server includes all the basic functionalities customers expect from a Windows Server operating system, such as dependability, security, and scalability. In addition, Microsoft has improved and extended the Windows Server product family to enable your business to experience full-blown .NET functionality."***

- Windows .NET Web Server
- Windows .NET Standard Server
- Windows .NET Enterprise Server
- Windows .NET Datacenter Server

currently in Beta 3



# For More Information

---

- Microsoft .NET

- ▶ T & I Competitive Information Database:

- <http://w3-3.ibm.com/software/sales/ti/ticompet.nsf>

- By Competitor/Category > Microsoft > Most Important Competitive Info > .NET White Papers

# Visual Studio.NET Overview

---

- **Visual Studio.NET will provide enhancements to existing Visual Studio 6.0 features and some brand new ones.**
  - ▶ **Key features:**
    - **Web Development Editors - for HTML, XML and style sheets**
    - **Visual Studio Languages - including the new C# (pronounced see-sharp)**
    - **Rapid Application Development (RAD) tools**
    - **Web and Windows Forms - for fast application development**
  - ▶ **Visual Studio will cater to the development of XML-based Web services**