

# Server-side Components and the Inter/tra/Extranet

Components go Global



Graham McLeod  
August 1998



# Key Trends

- **Staggering Internet Growth**
- As big or greater demand to use web technologies inhouse (**Intra/Extranet**)
- **e-Commerce** taking off in a big way
- Web is inherently object with **multimedia** and hyperlinks
- **Fat clients are too expensive** and are not manageable
- Huge need to "serve" objects
- Objects go distributed
- Applications span technology environments & enterprises
  - ▶ Requires portability, interoperability
- Objects must persist across sessions, locations; Transactions must be secure, available and reliable
- Major need for services to support complex distributed object transactional environment



# Large-scale object web computing requires

## ■ Client transparency

- ▶ It has to run on all the popular browsers

## ■ Thin Client

- ▶ Must be quick to download
- ▶ Must not strain local resources
- ▶ Easy to administer centrally

## ■ Interoperability with

- ▶ Development tools
- ▶ Execution environments
- ▶ Protocols
- ▶ Legacy stuff

## ■ Industrial Strength

- ▶ Security => Encryption, Authentication, Non-repudiation  
Tamper-proofing
- ▶ Reliability => Transaction Management
- ▶ Portability => Write once, run anywhere
- ▶ Scalability => Hundreds of connections, Thousands of transactions, Millions of objects



# Large-scale object web computing requires (cont'd)

---

## ■ Multi-tier architecture

- ▶ Start small and grow
- ▶ Failsafe

## ■ Investment Protection

- ▶ Strategic security
- ▶ Multi-vendor support
- ▶ Portability

## ■ Highly productive development tools

- ▶ Visual building
- ▶ Reusable Components
  - Technical
  - Business



# Java is hype, but here to stay

---

- Its **sexier** than C++
- Its close to real objects
- Its almost efficient
- Its almost portable
- It can leverage a huge
  - ▶ Skills base
  - ▶ Code base
  - ▶ Range of development tools
  - ▶ Range of libraries
- Supported by all major industry players including:
  - ▶ Sun
  - ▶ IBM
  - ▶ Inprise (Borland), Sybase, Oracle, Symantec etc.
  - ▶ Microsoft (sort of)

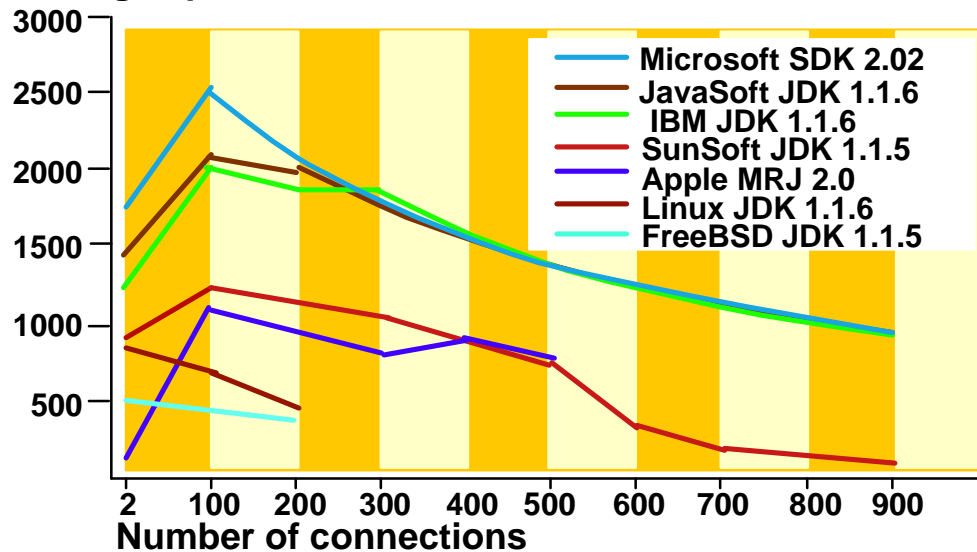


# But can it scale?

- Source: Neffenger, Aug 98
- Individual Java VMs are getting better quickly
- But multi-server, load balanced environments required for really high volumes

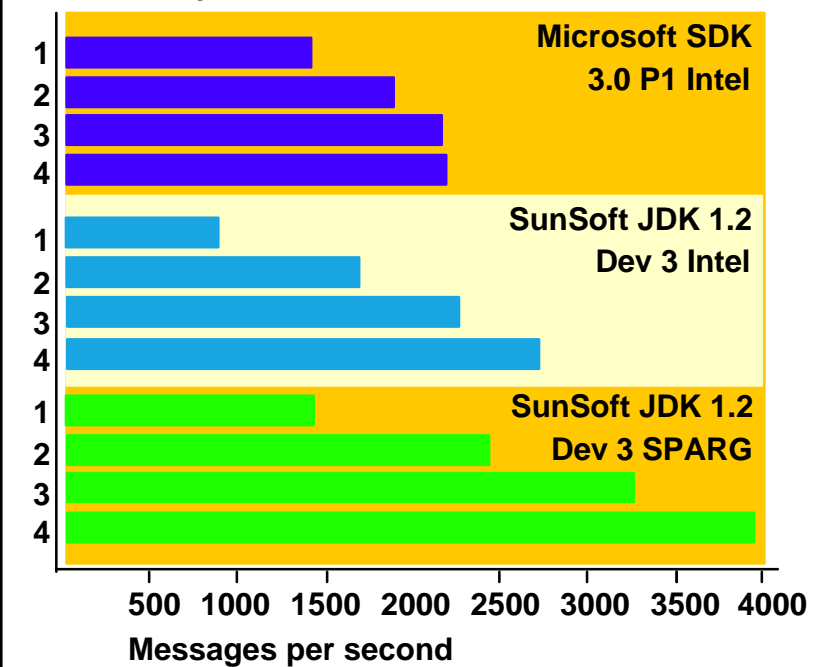
VolanoMark 2.0.0 Build 137 Network Test:

Current Java VM scalability  
Messages per second



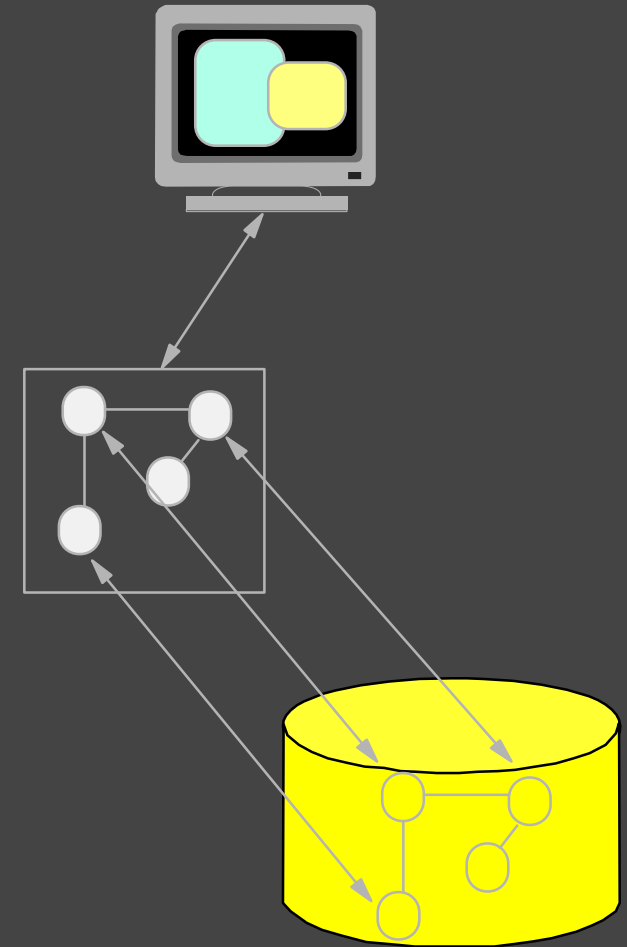
VolanoMark 2.0.0 Build 137 Local Test:

Processor scalability  
Number of processors

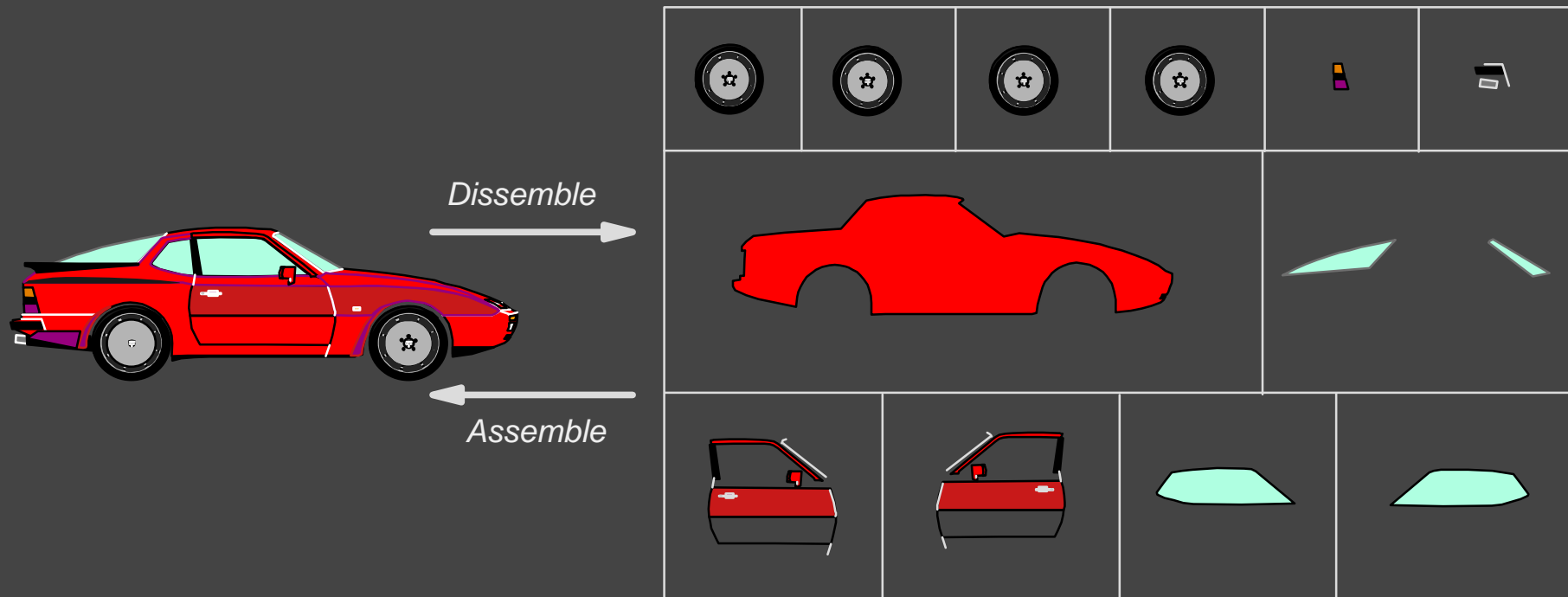


# Object Databases - Characteristics

- ✓ Provide all standard database facilities
- ✓ Support Classes/Types and inheritance in schema's
- ✓ Provide transparent persistence to programmer
- ✓ Store objects (logically: data with related procedures)
- ✓ Store millions or trillions of objects
- ✓ Support multiple OO languages (C++, Smalltalk, Java)
- ✓ Provide intelligent objects which manage their own integrity
- ✓ Provide easy, safe maintenance and extensibility
- ✓ Very high performance for complex data
- ✗ Don't have "weight" of relational vendors
- ✗ Interoperability and strategic investment were concerns



# Parking the *Car* Object in a Relational Garage

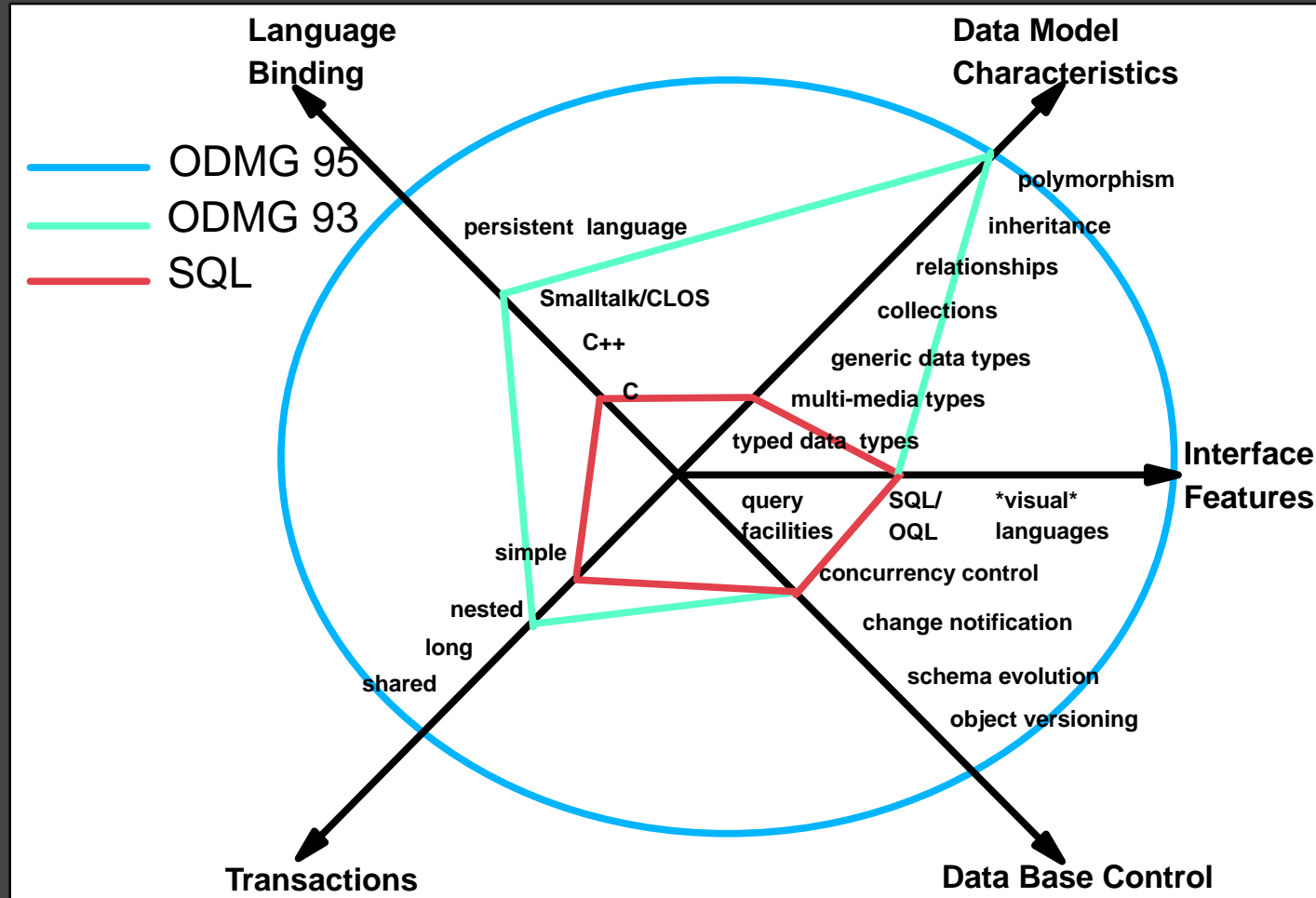


- Requires either:
  - ▶ Specialists to assist the driver
  - ▶ An extremely technically competent driver
- And lots of time...!



# Database Capabilities Compared

- Diagram compares the ODMG'95, ODMG'93 and SQL 2 standards

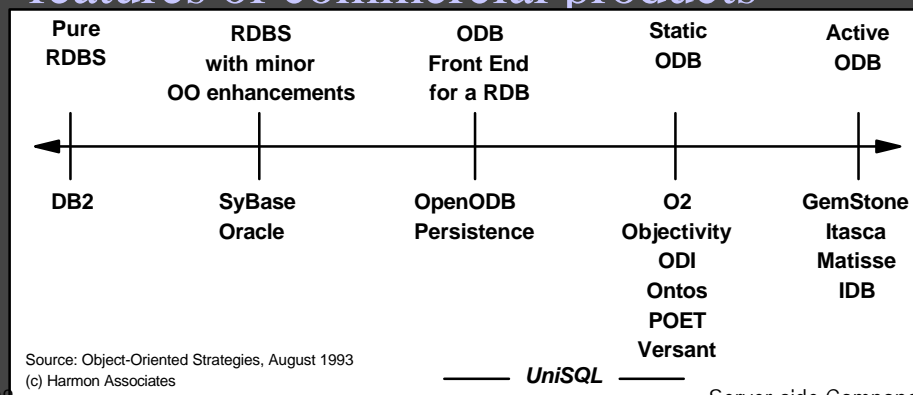


# Object Databases - Players

- Object Design Inc.                      ObjectStore
- Versant                                      Versant
- Ardent                                        O2
- Gemstone                                    Gemstone
- Computer Associates                    Jasmine
- Ontos                                        Ontos
- Poet                                         Poet
- Objectivity                                Objectivity
- ADB                                         Matisse



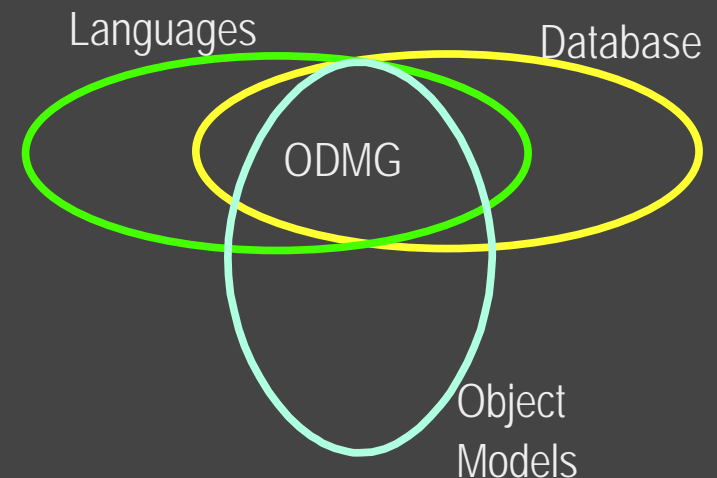
- Continuum below shows progression in features of commercial products



# ODMG 2.0

---

- To ODBMS what SQL 2 spec was for relational
- Transparent persistence for C++, Smalltalk, Java
  - ▶ Persistence by reachability
- Multiple implementations
  - ▶ Preprocessor, post processor, smart interpreter
- Java native implementation (ODMG classes added to language)
  - ▶ Includes collections and relationships
- Applications portable across object databases
- MetaModel language independent
  - ▶ Standard data interchange
- Widely supported by DBMS vendors



# Object Relational

---

- Relational camp under threat
  - ▶ Not for traditional TP and batch work
  - ▶ But for
    - Multimedia (Voice, Video, Music)
    - Time series
    - Spacial data
    - Complex data (e.g. CAD, CASE)
    - Web applications (esp Java, XML)
    - Imaging, Document management
  - ▶ Most capable product probably Informix/Illustra
  - ▶ Challengers:
    - IBM Universal Server
    - Oracle
    - Sybase



# Component Models

---

## ■ DCOM/Activex

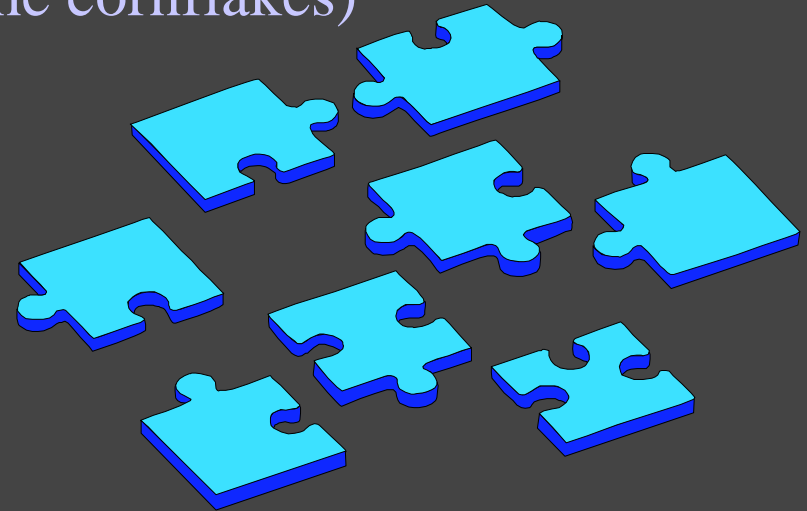
- ▶ On virtually every desktop (look in the cornflakes)
- ▶ But questionable architecture

## ■ CORBA

- ▶ Mature, multivendor, capable
- ▶ Not always fully implemented
- ▶ Used to cost money
- ▶ Verification Service - Open Group
- ▶ Tools were lacking - beaten by Ms "X" appeal
- ▶ Vsn 3.0 today!

## ■ JAVA Beans

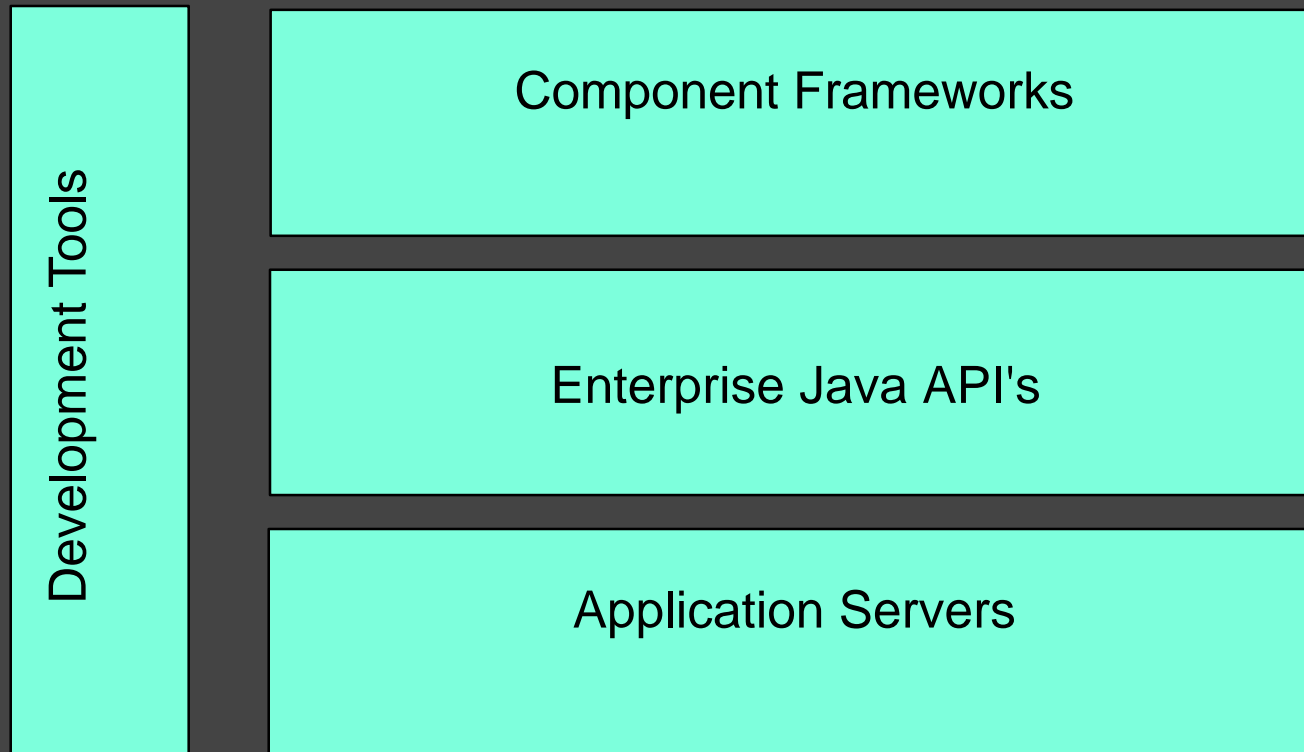
- ▶ Attractive to developers
- ▶ Portability promises, promises
- ▶ Only part of the picture



# Enterprise Java Platform

---

- Sun's vision for portable server code



# Enterprise Java Beans

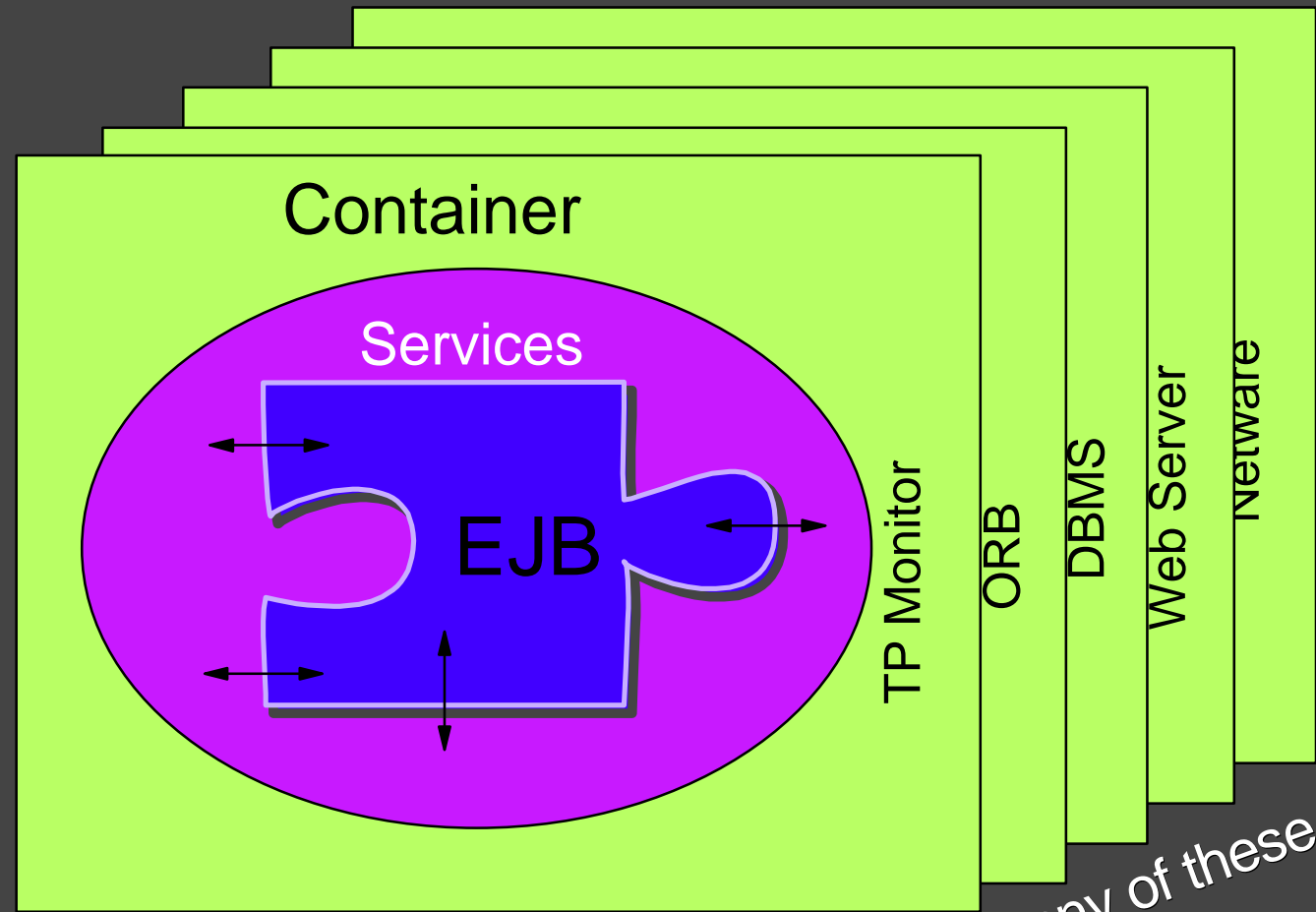
---

- Derived from Java Beans
  - ▶ Server Component Model for Java
  - ▶ With additions for server based processing
  - ▶ Non-visual
  - ▶ Supported by common services required
- Aim to be (via multi-tier, distributed object, thin client architecture)
  - ▶ Scalable (global web serving)
  - ▶ Reliable (redundant, no single point of failure)
  - ▶ Efficient (sharing of resources, threading etc)
  - ▶ Transactional
  - ▶ Secure
  - ▶ Flexible
  - ▶ Easy to maintain, manage



# Component Execution Environment

- Threading
- Creation, Destruction
- State Mgmt
- Resource Sharing
- Identity Resolution
- Transaction Management
- Persistence



EJB Server can be any of these



# EJB Server

---

- Bean container
- Transaction Services (implicit, declarative)
- Lifecycle management, naming
- Persistence, DB access
- Transparent Distribution, Remote Invocation
- Security services

**Kind of like  
CORBA?**

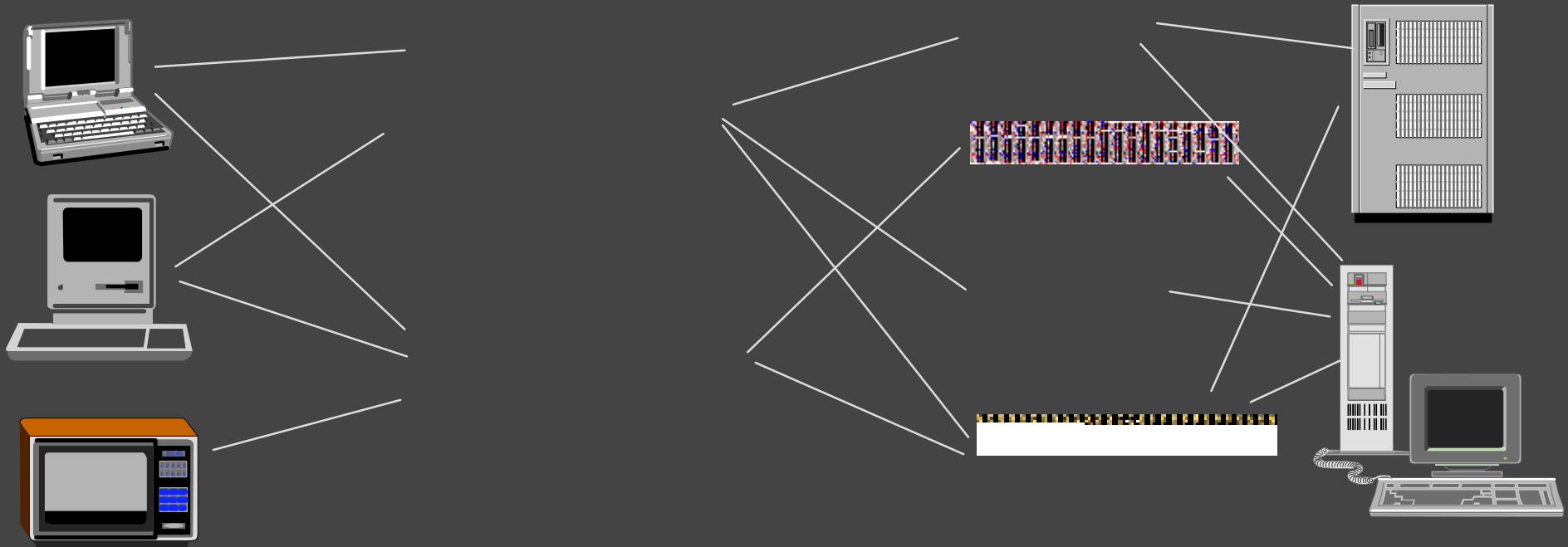
## Server Candidates:

- |                     |  |
|---------------------|--|
| ■ TP Monitors       | CICS, Encina, Tuxedo (M3)                  |
| ■ Component Servers | Jaguar, MTS                                |
| ■ ORBs              | VisiBroker, Orbix                          |
| ■ DBMS              | Oracle, Sybase, DB2, Gemstone...           |
| ■ Web Servers       | Java Web Server, Netscape Enterprise Serv. |



# Serious Flexibility

- Leverage
  - ▶ CORBA
    - Interoperability (Languages, Protocols, Platforms)
    - Services (Naming, Security, Scalability, Messaging, Transactions...)
  - ▶ Java Portability (Development tools, execution environment, instant deployment)



Clients

Development Tools

EJB Servers

Platforms



# Objects meet Transactions meet the Web

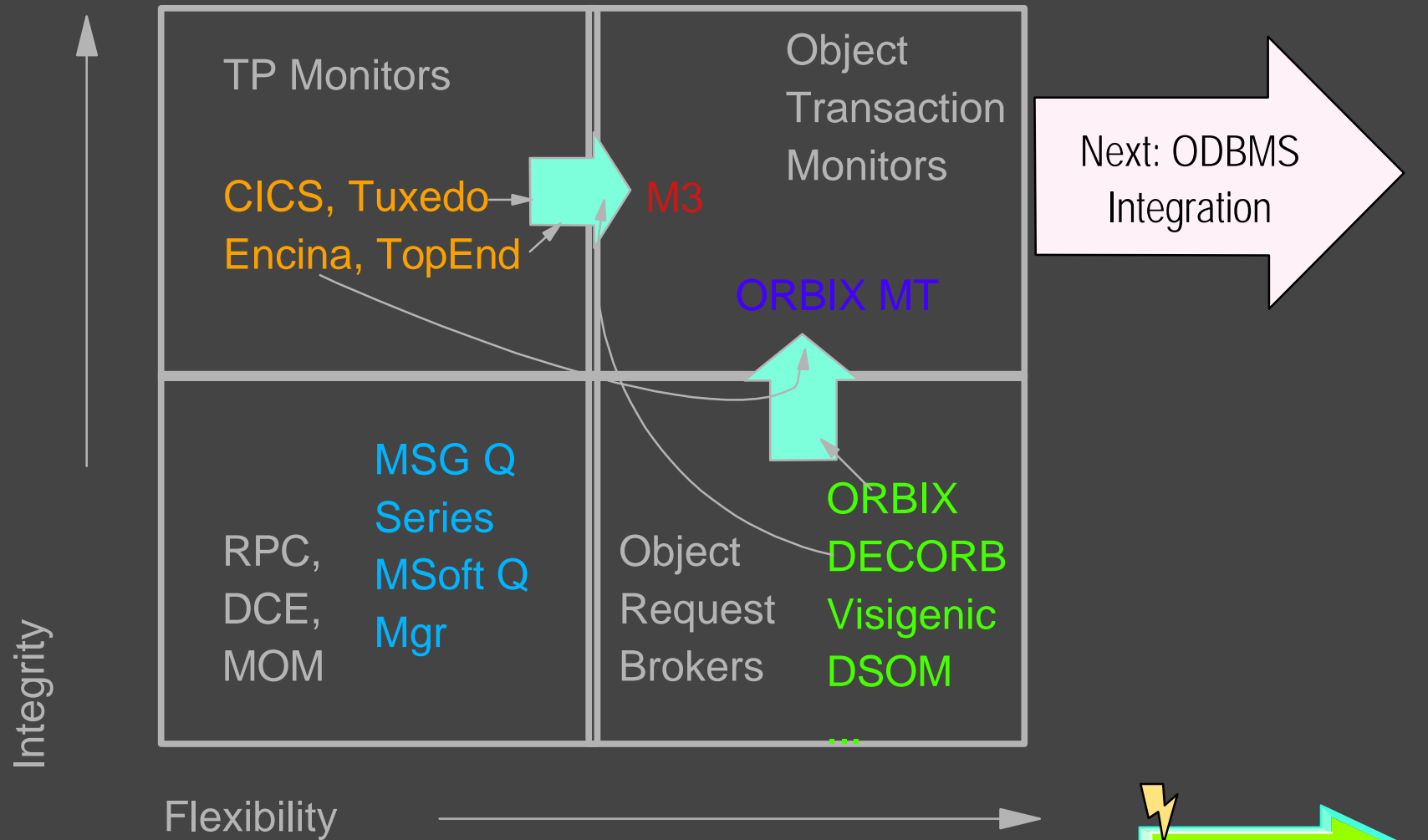
---

- **Applets**
  - ▶ Java (or other downloadable programs) that run in a browser client
- **Servlets**
  - ▶ Java (or other components) that run in the server
- **Orblets**
  - ▶ Lightweight ORB written in Java, runs inside browser
- **Aglets**
  - ▶ Downloadable, portable agents
- **Iona releases OTM incorporating Encina with Orbix**
- **BEA releases M3/Iceberg, incorporating DEC Orb**



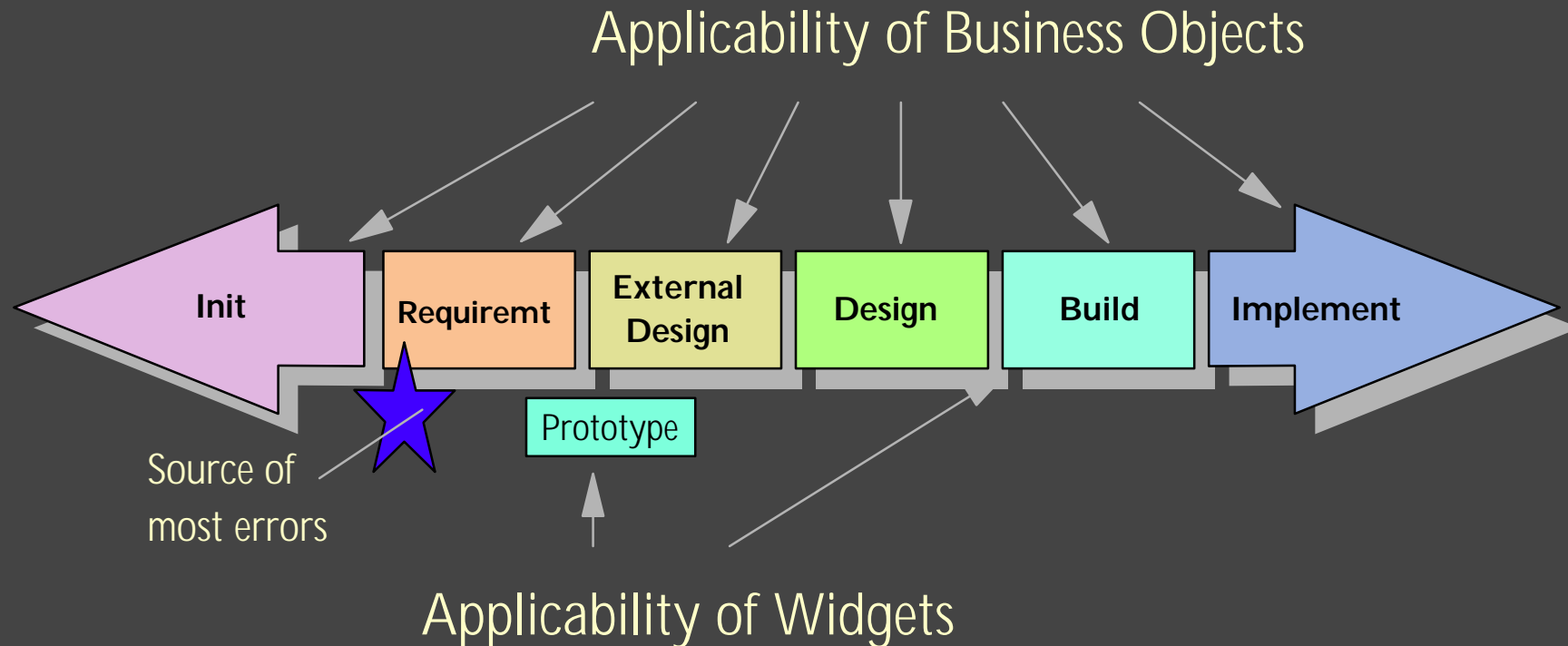
# Marriage of TP and ORBs

Matrix styled after Gartner Group model



# Leverage of Components

- Why all the fuss?
  - ▶ Components can dramatically shorten development
  - ▶ Ease maintenance
  - ▶ Increase target space



# Religious Wars

---

- The battles
  - ▶ Microsoft DCOM vs CORBA
  - ▶ Java RMI vs CORBA
  - ▶ Java Beans vs ActiveX
  - ▶ RPC vs MOM
- Truce!
  - ▶ MS agrees to CORBA beyond the desktop
  - ▶ JAVA agrees to CORBA
  - ▶ OMG supports DCOM
  - ▶ CORBA supports MOM as alternative model



# CORBA 3.0 - CORBA Beans

---

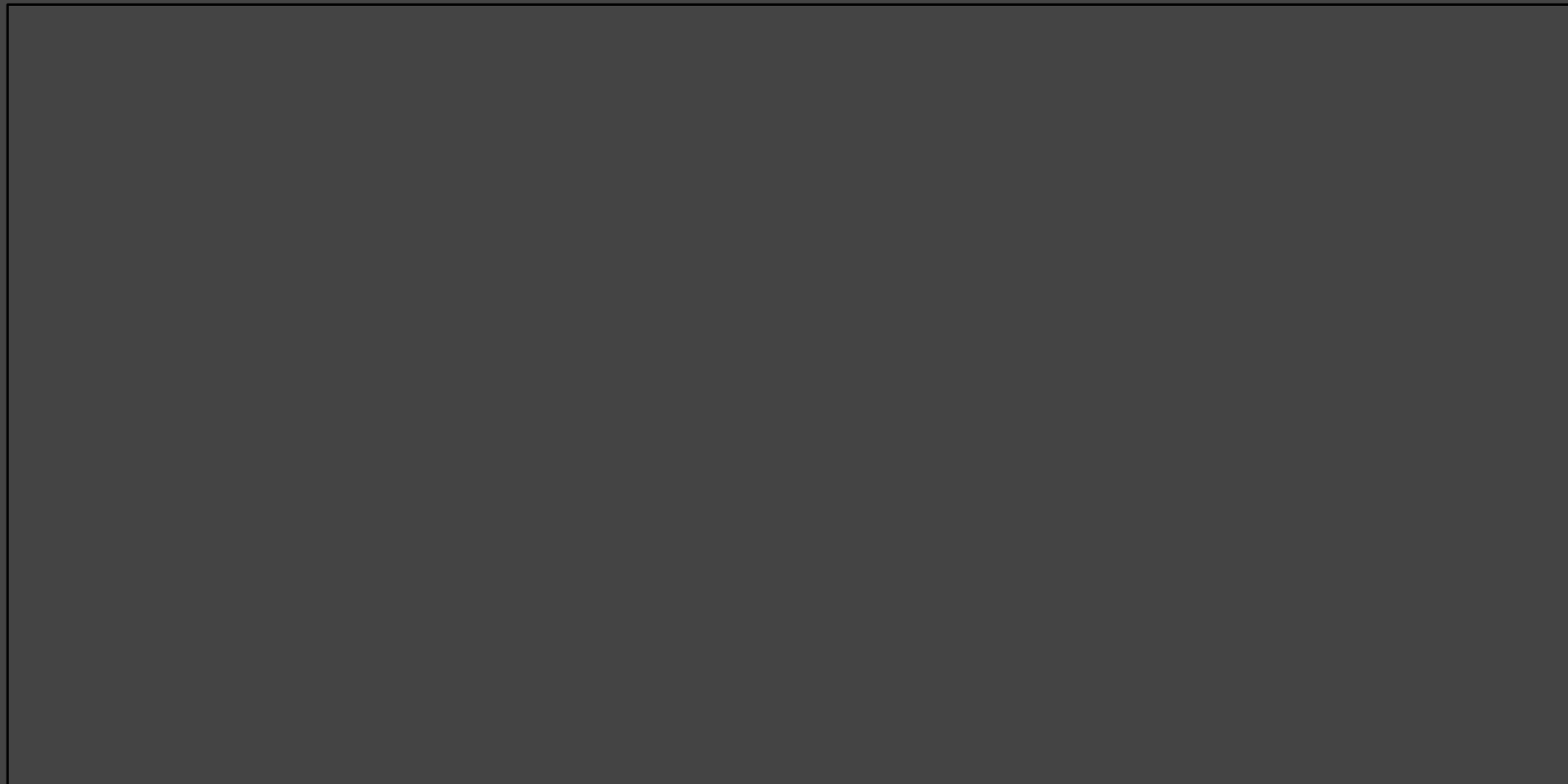
- Announced Sept 9 1998
- Includes full CORBA to EJB interoperability
- Standards for Message Oriented Middleware asynchronous messaging
- DCOM/ActiveX interoperability
- CORBA Beans - CORBA component model based on EJB's, but language neutral
- Fully backward compatible to CORBA 2
- Visual Development and Application Scripting



# Business Components Ahoy!

---

- IBM San Francisco
- OMG
- Many other vendors with EJB



- COTS vendors e.g. BAAN



# Role of Agents

---

