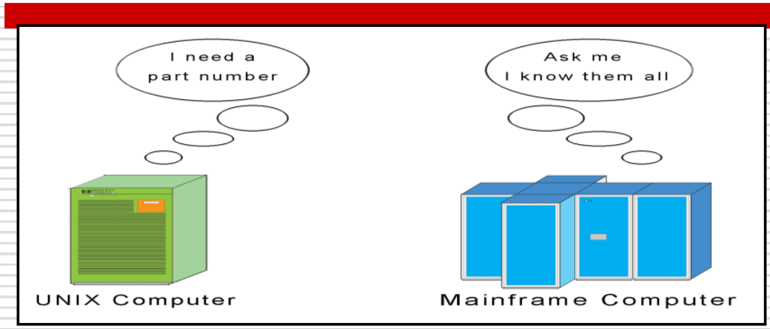
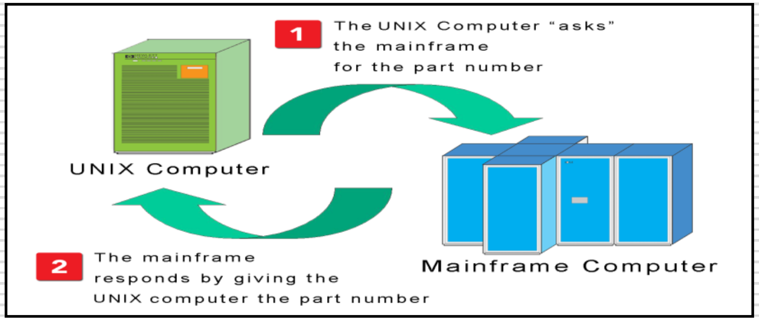
SECTION 1

MOTIVATION FOR SOA

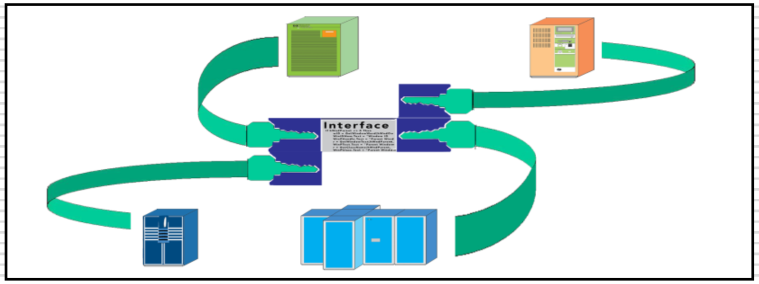
What is SOA? First, Understand “Tight Coupling”



- □ Data and functionality typically reside on more than one system (and application)


1 The UNIX Computer "asks" the mainframe for the part number
UNIX Computer
Mainframe Computer
2 The mainframe responds by giving the UNIX computer the part number

- □ Applications need to be able to “talk to each other”



- □ Status quo: Proprietary or custom communication interfaces between applications

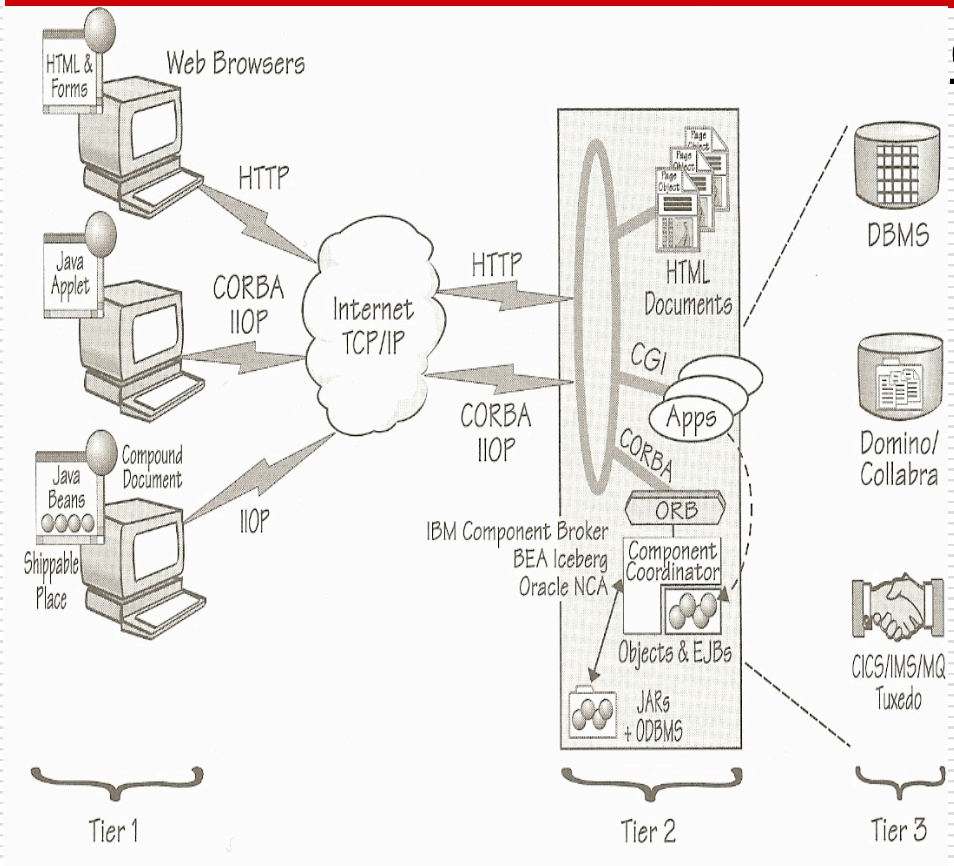
Source: H. Taylor, “*Service-Oriented Architecture (SOA) 101 ‘What’s Hype, What’s Real?’*”, *Juniper Networks, Inc.,2007.*

Challenges with Tight Coupling

- While tight coupling is inherently sound, the following challenges are encountered in its implementation:
  - It's costly to maintain
  - Slow and costly to change
  - Cost and complexity compounded by multi-party scenarios such as B2B or integration with the public sector
  - Cost and complexity of managing and changing a tightly coupled architecture makes business agility difficult (IT can't keep up with business needs, but it's not their fault)
  - **Does not support reuse!**
- Recognized for many years as challenge industry wanted to solve
- Evolution of reuse solutions reflects industry's concerns
  - Header files, inheritance and polymorphism at the object level, frameworks
  - CORBA (Common Object Request Broker Architecture)
  - Microsoft COM (Component Object Model)
  - EAI (Enterprise Application Integration)
  - Web Services

Source: H. Taylor, "*Service-Oriented Architecture (SOA) 101 'What's Hype, What's Real?'*", *Juniper Networks, Inc.,2007.*

Challenges with Tight Coupling



The 3-Tier CORBA/Java Object Web.
Source: Client/Server Programming with JAVA and CORBA Second Edition by R. Orfali and D. Harkey, p. 45.

Overview of CORBA

□ Tier1 belongs to traditional Web browsers and Web-centric applications

□ Tier 2 runs on any server that can support HTTP and CORBA clients
  - CORBA objects, like EJBs, encapsulate business logic

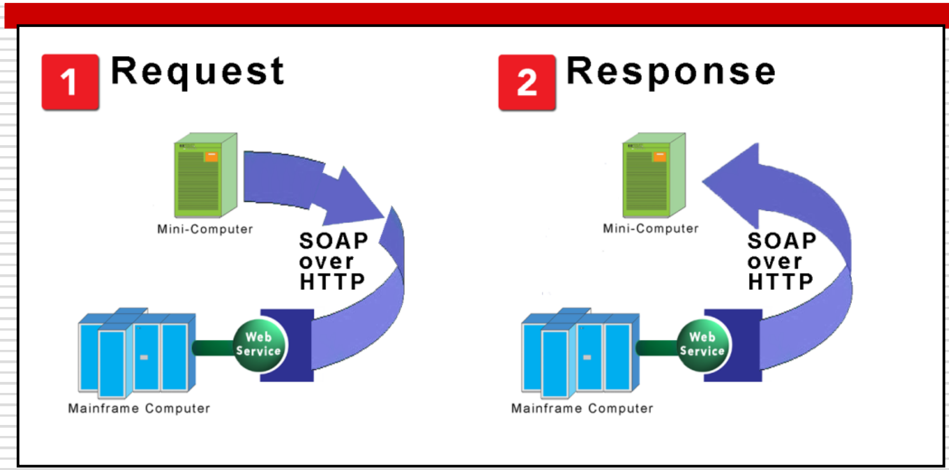
□ Tier 3 consists of almost anything a CORBA object can access

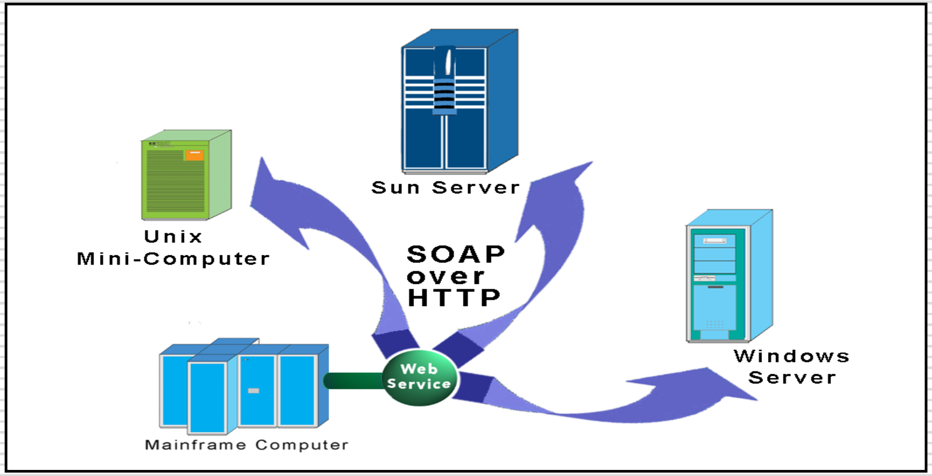
Challenges with Tight Coupling

- Overview of COM
  - Microsoft COM (Component Object Model) enables software components to communicate
  - COM is used by developers to create re-usable software components, link components together to build applications, and take advantage of Windows services
  - COM objects can be created with a variety of programming languages including object-oriented languages such as C++
  - The family of COM technologies includes COM+, Distributed COM (DCOM) and ActiveX® Controls.
  - The .NET Framework provides bi-directional interoperability with COM, which enables COM-based applications to use .NET components and .NET applications to use COM components
- Reasons CORBA, COM, EAI and others did not work
  - Lack of open standards
  - Proprietary components

Source: http://www.microsoft.com/com

SOA: The Ideal of Open Interoperability (Loose Coupling)





SOA – A Definition

- □ An IT architecture composed of software that has been exposed as "Services" – i.e. invoked on demand using a standard communication protocol.
- □ "Web Services" – software available as a "service" using Internet protocols.
- □ One software application talking to another using a standards-based (i.e. non-proprietary) language over a standards-based communication protocol.
- □ Universal "Dial Tone" between software applications
- □ An IT architecture that enables "loose coupling" of applications

Source: H. Taylor, "*Service-Oriented Architecture (SOA) 101 'What's Hype, What's Real?'*", *Juniper Networks, Inc.,2007.*

Core SOA Definitions

- **XML** – Extensible Markup Language
- **SOAP** – Simple Object Access Protocol
- **WSDL** – Web Services Description Language
- **UDDI** - Universal Description, Discovery and Integration
- **ESB –** Enterprise Service Bus
- Key Concepts
  - Network Transparency
  - Virtualized endpoint
  - Self-describing software
  - Universally discoverable software
  - Universally understood software
  - Machine to machine interaction

Source: H. Taylor, "*Service-Oriented Architecture (SOA) 101 'What's Hype, What's Real?'*", *Juniper Networks, Inc.,2007.*

SOA Usage & Supporting Platforms

- SOA Usage
  - B2B
  - Enterprise Application Integration (EAI)
  - Application to Application
  - Government
- Major Players in SOA Space
  - IBM: WebSphere SOA Product Suite
  - BEA: Aqualogic (WebLogic)
  - Oracle: Fusion Middleware
  - Microsoft: .NET
  - SAP: NetWeaver

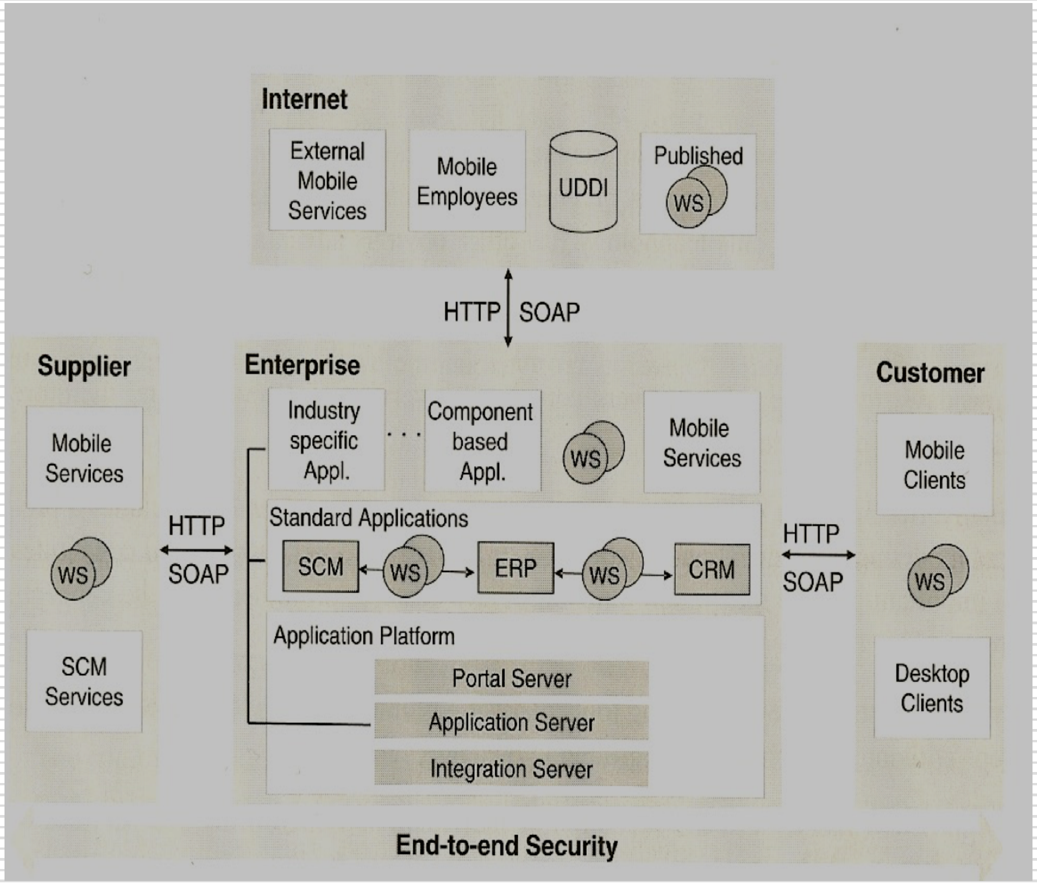
Source: H. Taylor, "*Service-Oriented Architecture (SOA) 101 'What's Hype, What's Real?'*", *Juniper Networks, Inc.,2007.*

What Makes Web Services Appealing?

	CORBA	JAVA RMI	ONC(SUN) RPC	WEB SERVICES
Data Encoding	Common Data Representation (CDR)	Serialized Java/CDR	Extended Data Representation (XDR)	XML (WS-I doc-literal, SOAP Encoding)
Message Format	IIOP (GIOP)	RMI Protocol/IIOP	RPC RMS	SOAP
Transport Protocol	TCP	TCP	UDP TCP	HTTP
Description Language	CORBA IDL	Java Interface/Class	RPC IDL	WSDL
Discovery Mechanism	COS Naming	RMI Registry	Undefined	UDDI
Invocation Method	CORBA RMI	Java RMI (method call)	RPC	Undefined

Source: See [56], Page 4.

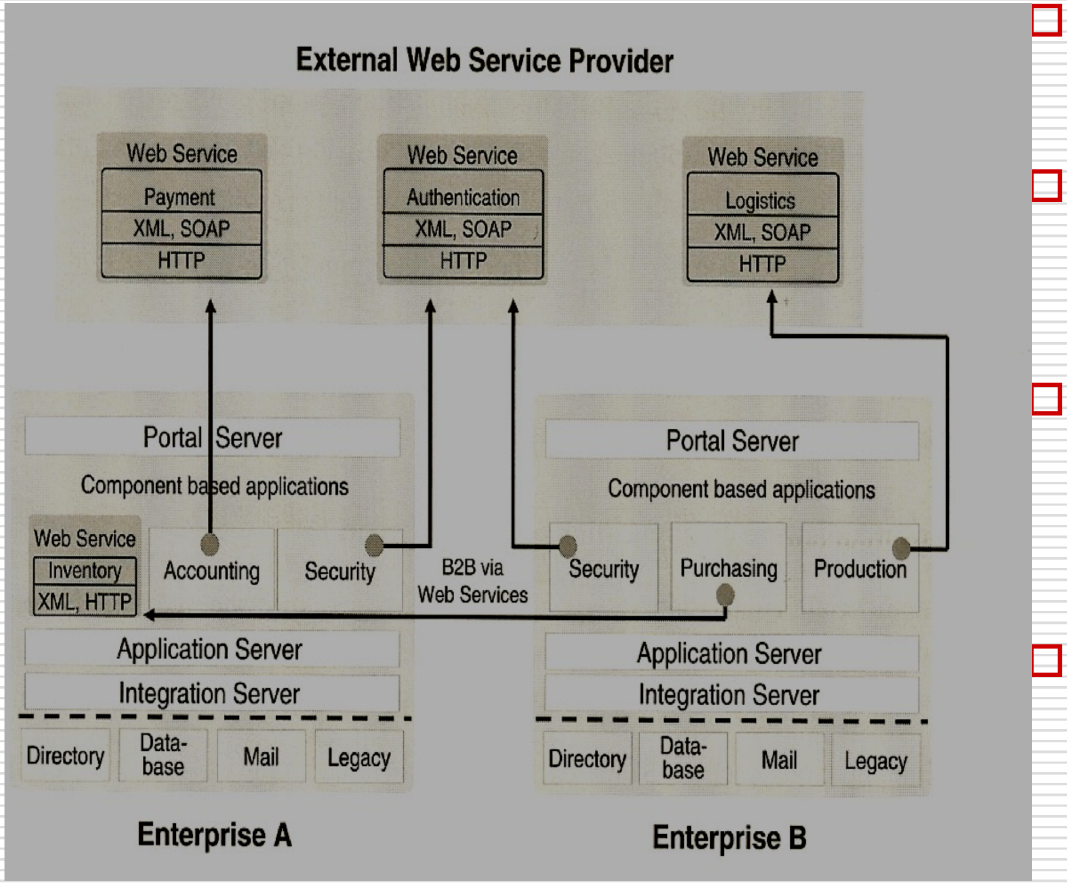
Cross-Enterprise Solution Architecture



- □ Figure illustrates tomorrow's e-business solution architecture
- □ 4 stakeholders communicate via Web services
  - Suppliers
  - Customers
  - Enterprise
  - Employees

Source: Mobility, Security and Web Services: Technologies and Service-Oriented Architectures for a new Era of IT Solutions by Gerhard Wiehler, p. 46.

e-Business Architecture for B2B



- ❑ 1 External Provider, 2 Enterprises: A & B
- ❑ Enterprise B **Purchasing** accesses Enterprise A’s **Inventory** Web Service
- ❑ Both enterprises access an authentication Web service provided by an external provider
- ❑ Enterprises A & B separately access Web services that provide **Payment** and **Logistics** Web Services

Source: Mobility, Security and Web Services: Technologies and Service-Oriented Architectures for a new Era of IT Solutions by Gerhard Wiehler, p. 101.