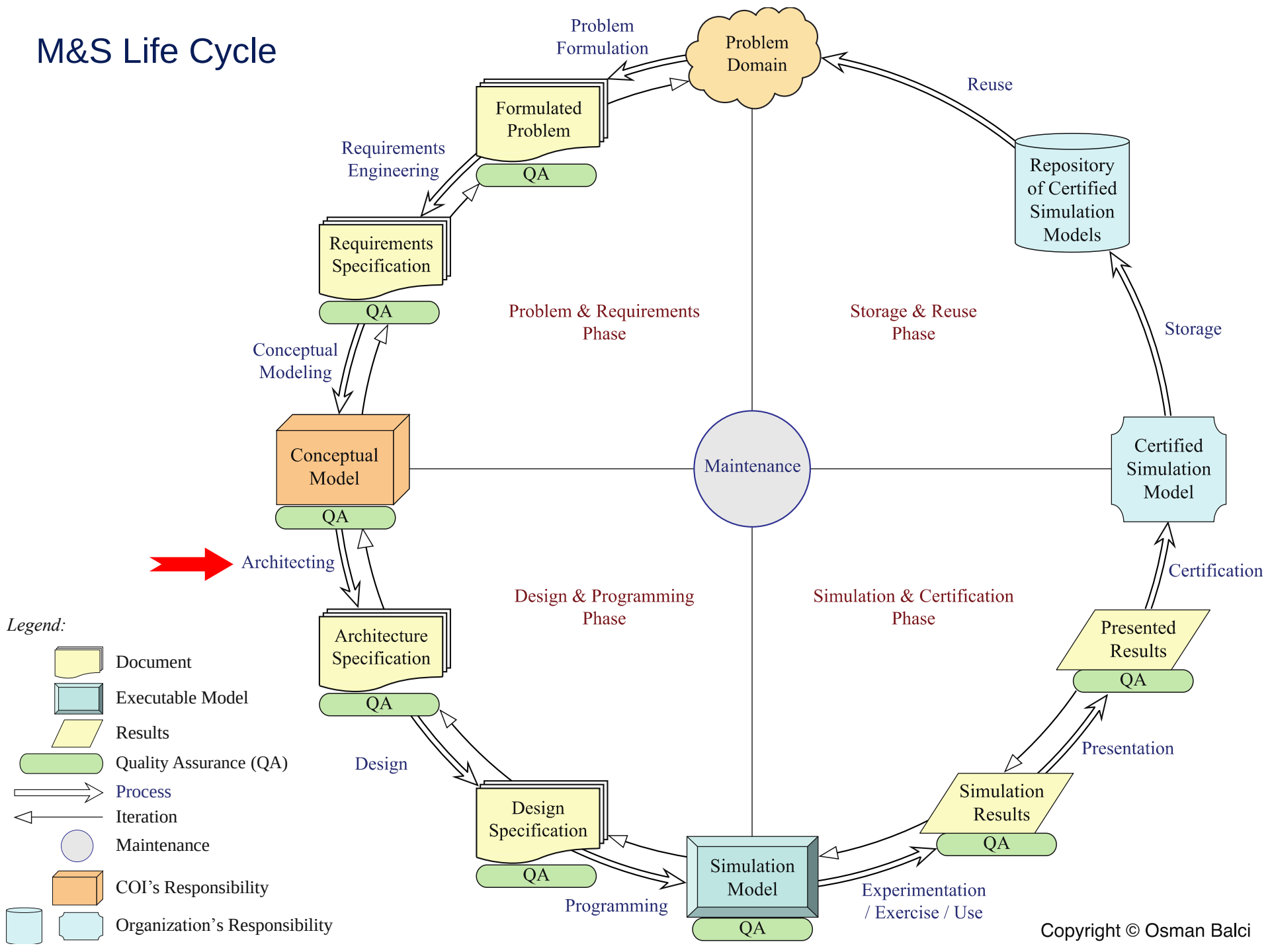
Architecting a Network-Centric M&S Application

OSMAN BALCI
Professor

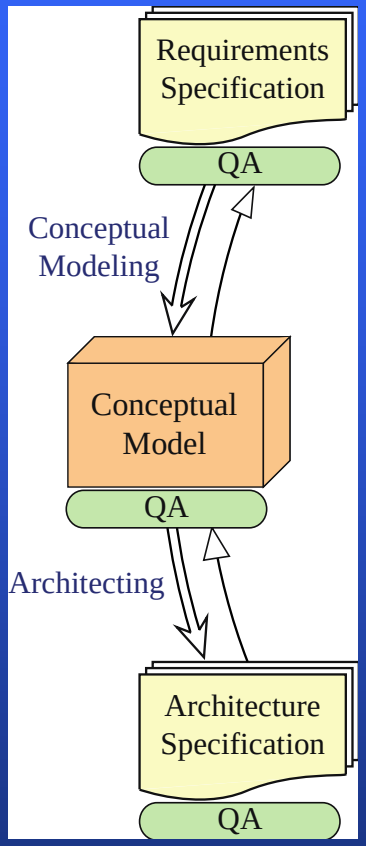
Department of Computer Science
Virginia Polytechnic Institute and State University (Virginia Tech)
Blacksburg, VA 24061, USA

https://manta.cs.vt.edu/balci

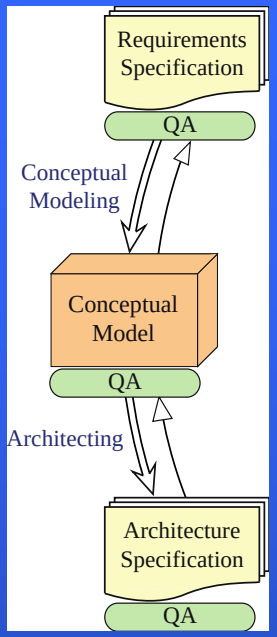
M&S Life Cycle



The Process of Architecting

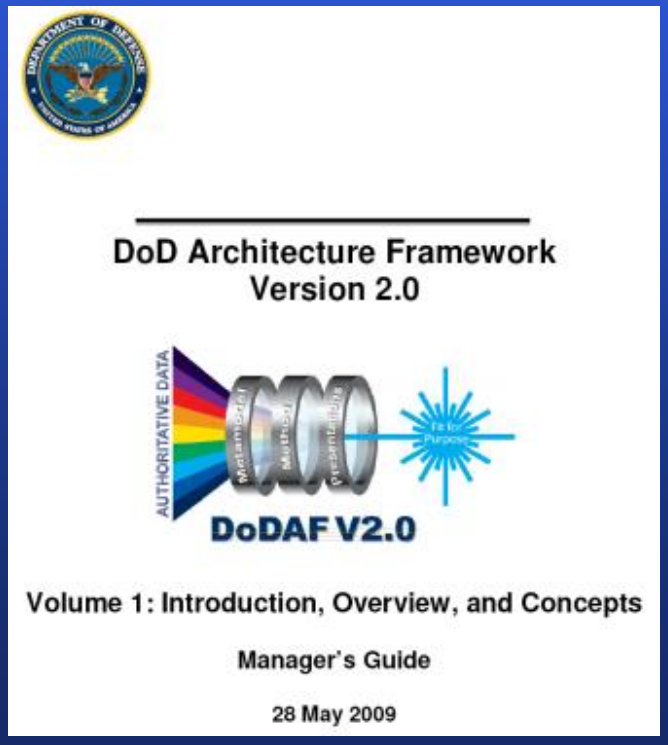


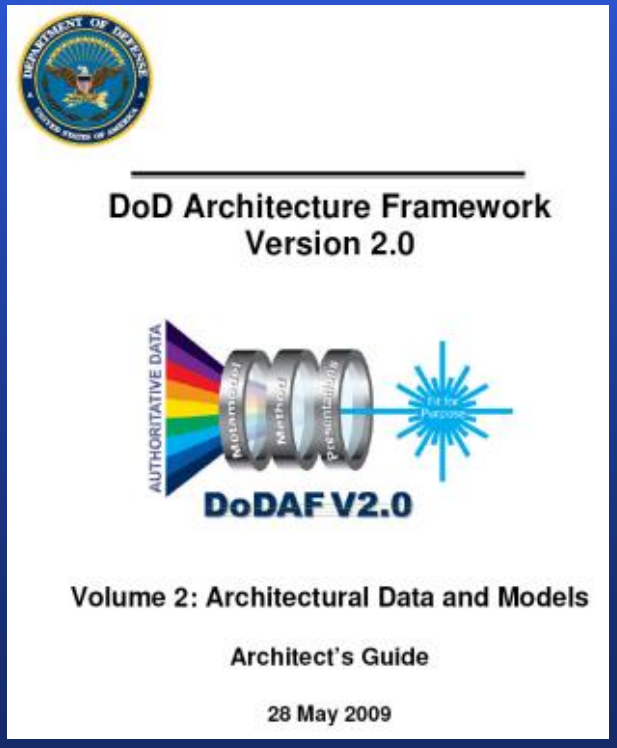
- **Architecting** is the process of creating and specifying an architecture for a **network-centric M&S application** based on the Conceptual Model and the M&S Requirements Specification Document (RSD).
- The process of Architecting takes the **Conceptual Model** and **M&S RSD** as input and produces a network-centric M&S **Architecture Specification** as the output work product.
- Architecting Quality Assurance (QA) integrates the assessments of
  - quality of the Architecture Specification work **product**,
  - quality of the architecting **process**,
  - quality of the **people** employed in architecting, and
  - **project** characteristics related to the life cycle stage for architecting.

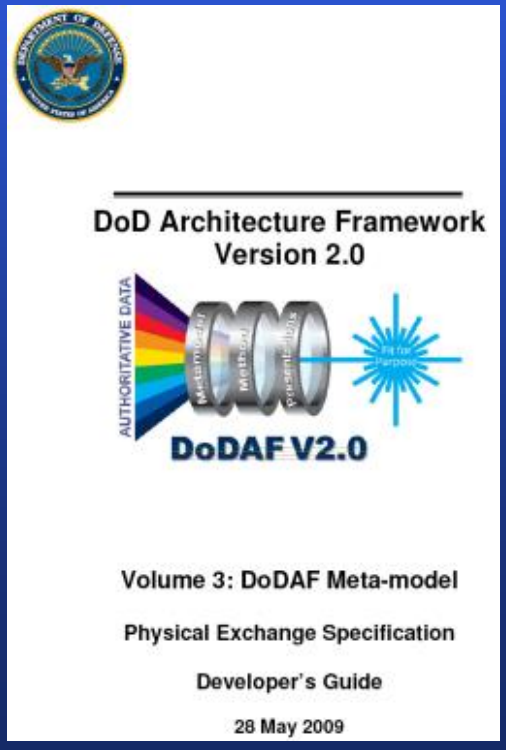


The Process of Architecting

The output, Architecture Specification, is created using a standard description framework such as DoD Architecture Framework (DoDAF).







Is "Architect" a Verb?

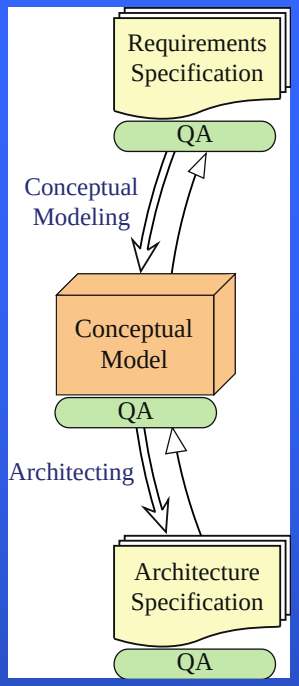
- Both the Oxford English Dictionary and Merriam-Webster's Third International dictionary list "architect" as a verb.
- Architect → (noun) a professional who creates an architecture
- Architect → (verb) act of creating an architecture
- Can you say "an Architecture" ? YES!
- Can you say "Architectures" ? YES!

- Merriam-Webster's 11th Collegiate Dictionary →

Main Entry: **ar-chi-tec-ture**
Pronunciation: 'är-kə-ˌtek-chər
Function: *noun*
Date: 1555

1 : the art or science of building; *specifically* **:** the art or practice of designing and building structures and especially habitable ones
2 a : formation or construction resulting from or as if from a conscious act 〈the *architecture* of the garden〉
b : a unifying or coherent form or structure 〈the novel lacks *architecture*〉
3 : architectural product or work
4 : a method or style of building
5 : the manner in which the components of a computer or computer system are organized and integrated

Example Architectures



The Process of Architecting

Amine Chigani and Osman Balci (2012), "The Process of Architecting for Software / System Engineering," *International Journal of System of Systems Engineering 3*, 1, 1-23.

- **The Architecting process deals with the creation of an architecture for a network-centric M&S application to satisfy the requirements stated in the RSD.**
- **The Architecting process can be conducted by**
  - **Selection of a known architecture,**
  - **Composition of an architecture from a set of known architectures, or**
  - **Creation of a new architecture.**

Architecture Definition

- "The structure of components, their relationships, and the principles and guidelines governing their design and evolution over time."

  [DoD Integrated Architecture Panel, 1995, based on IEEE STD 610.12]

- "An architecture is the fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution."

  [IEEE STD 1471-2000]

Terminology

- Cloud Computing
- Cyberinfrastructure engineering
- Distributed system/software engineering
- E-system (e.g., e-business, e-government) engineering
- Internet-based system/software engineering
- Network-centric system/software engineering
- Online system/software engineering
- Web engineering
- Web-based system/software engineering

All of the terms above imply a system the components of which interoperate with each other over a network (e.g., Internet, virtual private network, wireless network) and the user uses the system over that network.

Network-Centric M&S Architecture

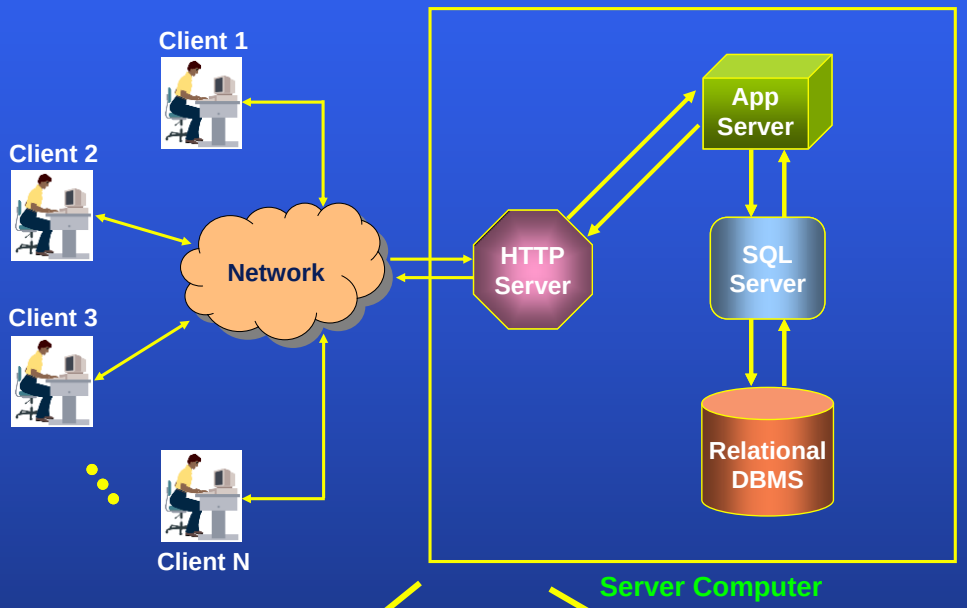
- A network-centric M&S architecture refers to the
  - fundamental organization of M&S components that interoperate with each other over a network,
  - relationships among the M&S components, and
  - the principles and guidelines governing the design and evolution of those M&S components.

- We use the term "network" to refer to one or a combination of many types of communications networks such as
  - Global System for Mobile communication (GSM),
  - Internet,
  - local area network (LAN),
  - mobile ad-hoc network (MANET),
  - virtual private network (VPN), or
  - wireless network.

Architecture versus Design

- A design is an instantiation from an architecture similar to how an object is an instantiation from a class.

Example:



Client-Server Architecture

Instantiation

Instantiation

Design based on the .Net platform

Design based on the Java EE platform

Major Network-Centric Architectures

Major Network-Centric System/Software Architectures that can be used for a Network-Centric M&S application:

1. Client-Server Architecture (CSA)
2. Distributed Objects Architecture (DOA)
3. Service-Oriented Architecture (SOA)

DoD / IEEE / NATO Standard Network-Centric (i.e., Distributed) M&S Architecture:

- High Level Architecture (HLA)

DoD Standard Description of Network-Centric Architectures

- DoD Architecture Framework (DoDAF)

Client-Server Architecture (CSA)

Two industry software standards for CSA:

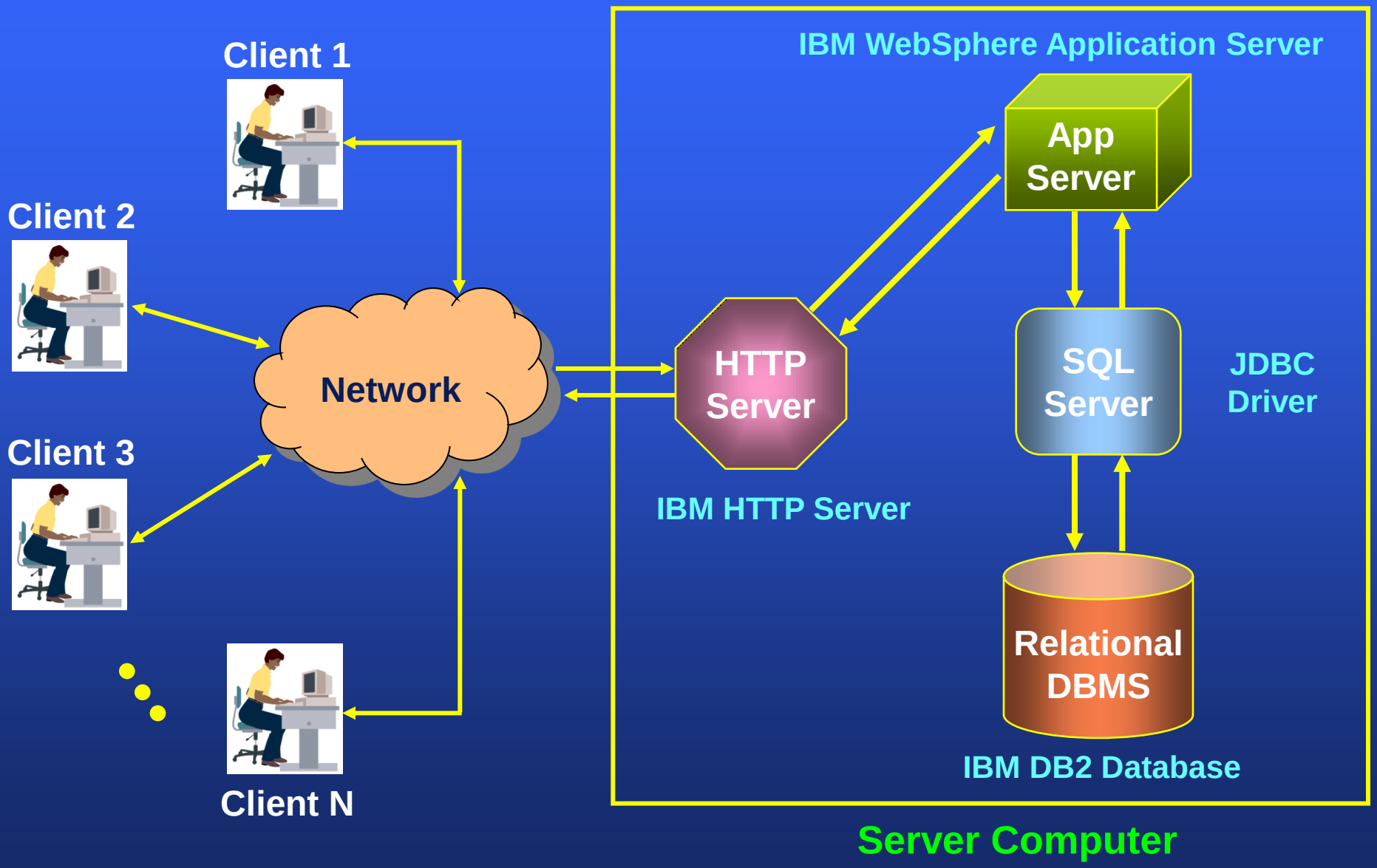
- **Java EE-based Client-Server Architecture**

  Oracle (2013), "Java platform Enterprise Edition (EE)," Oracle Corporation, Redwood Shores, CA.

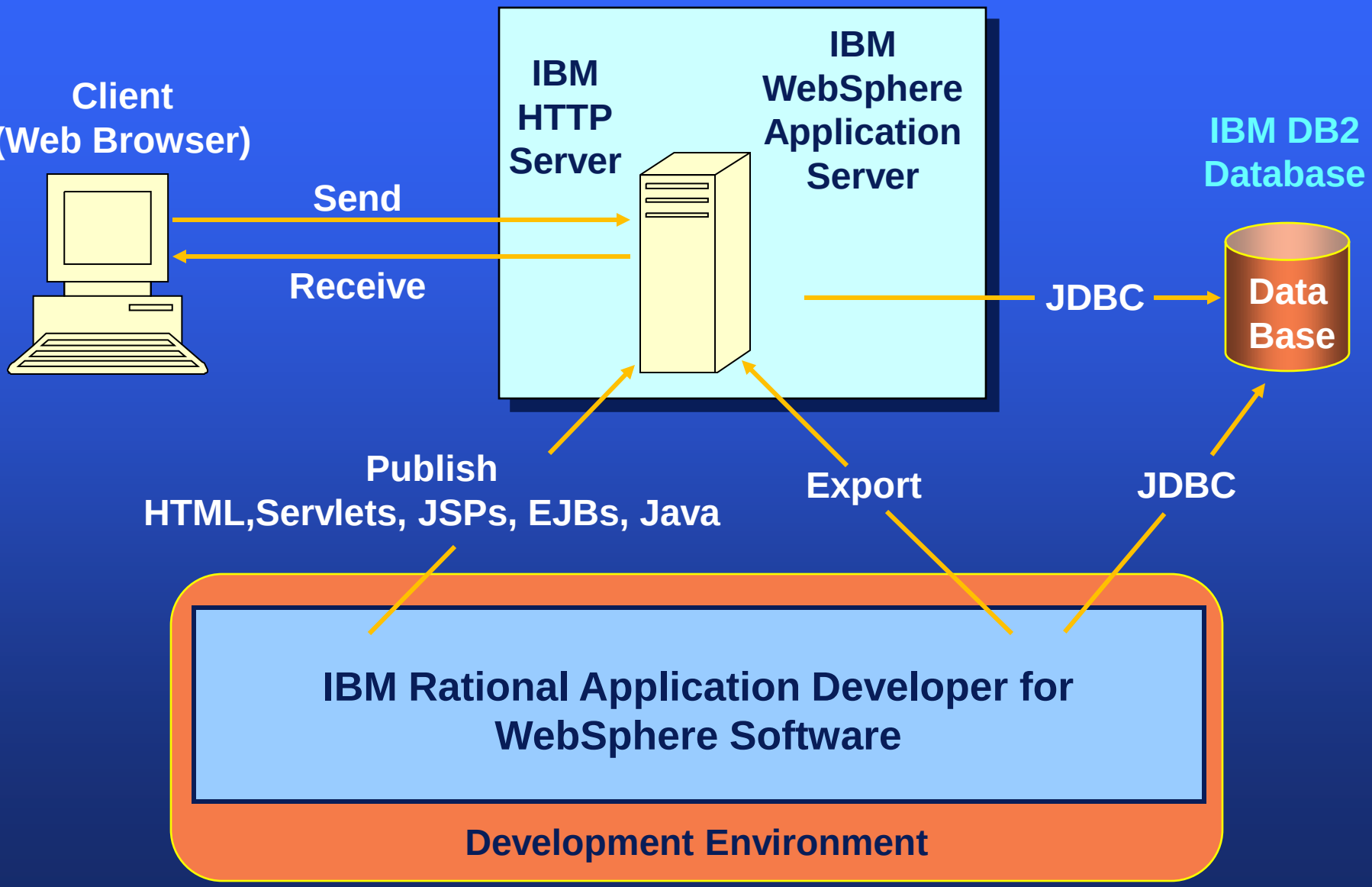
- **Microsoft .NET-based Client-Server Architecture**

  Microsoft (2013), "Microsoft platform .NET Framework," Microsoft Corporation, Redmond, WA.

A Client-Server Architecture with Example Components



Example IBM Development and Deployment Environment



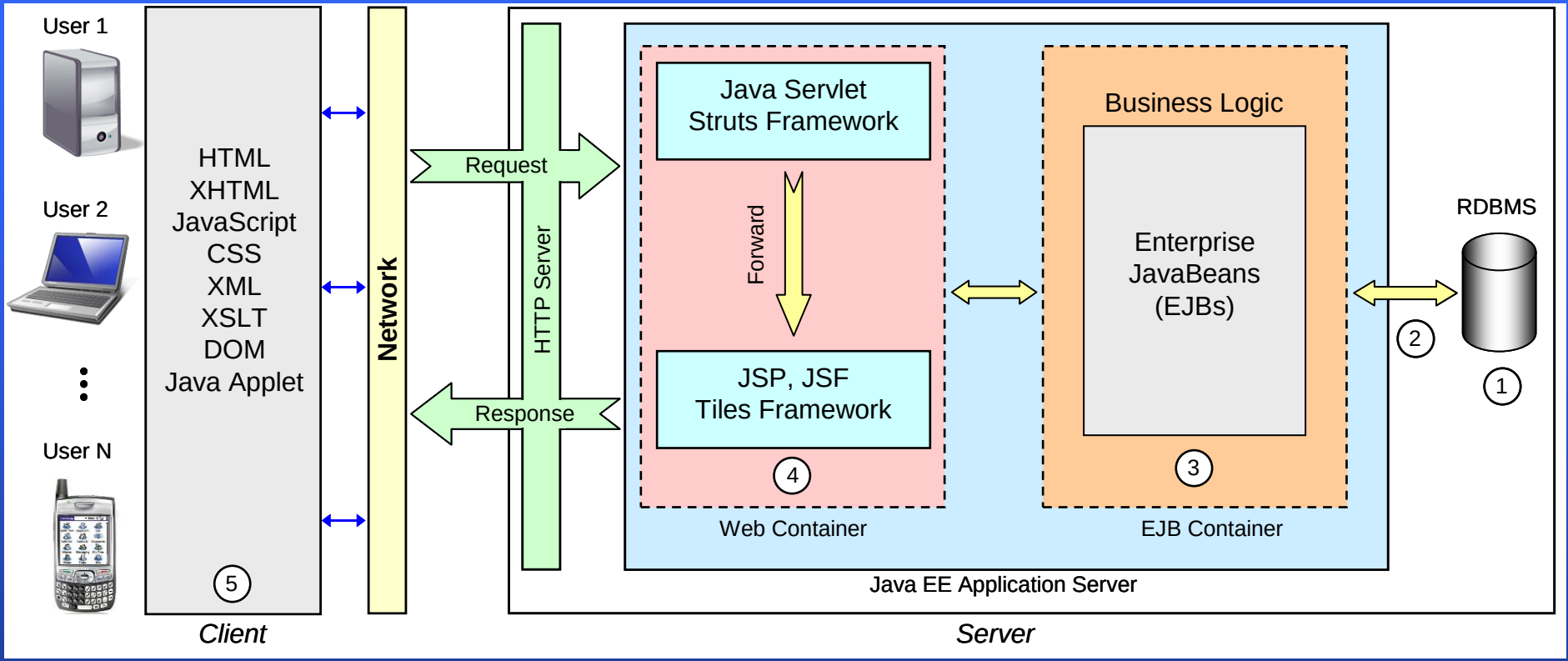
Thin and Thick Clients

- **Thin-Client Model**
  - In a thin-client model, all of the application processing and data management is carried out on the server. The client is simply responsible for running the presentation software.
  - A thin-client computer accesses applications and data from a server.
- **Thick-Client Model**
  - In this model, the server is only responsible for data management. The software on the client implements the application logic and the interactions with the system user.
  - A thick-client computer includes an operating system and installed applications and can run either as a standalone or in a server environment.

Thin and Thick Clients

- **Thin-Client Model**
  - A major disadvantage is that it places a heavy processing load on both the server and the network
- **Thick-Client Model**
  - More processing is delegated to the client as the application processing is locally executed
  - Most suitable for new client-server systems where the capabilities of the client system are known in advance
  - More complex than a thin client model especially for management. New versions of the application have to be installed on all clients

Java EE-based Client-Server Architecture

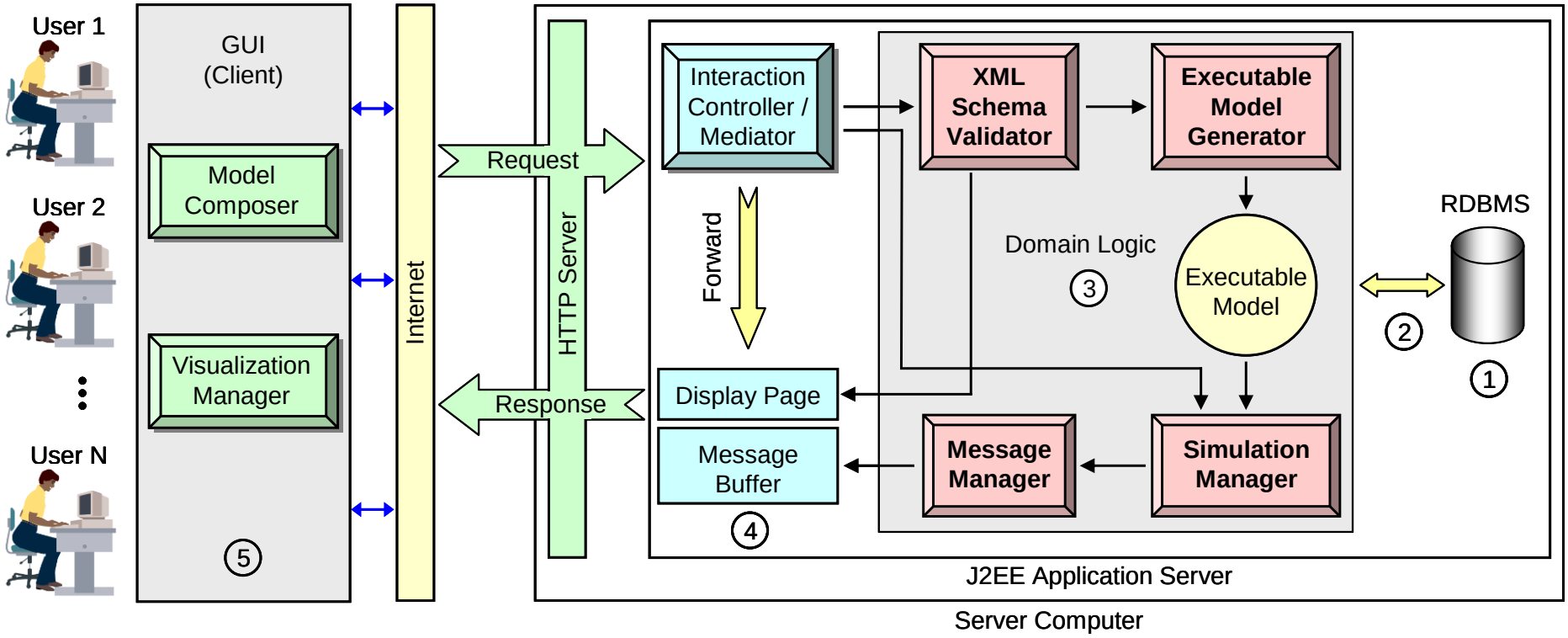


1. **Data Source Layer** (Relational DBMS such as DB2, Oracle, SQL Server)
2. **Data Mapping Layer** (JDBC, Entity EJBs)
3. **Business Logic Layer / Domain Logic Layer / EJB Container Layer** (EJBs)
4. **Web Container Layer** (Servlets, JSP, JSF)
5. **Client Presentation Layer** (JavaScript, XML, AJAX, XHTML, CSS, XSLT, DOM)

Definitions of Acronyms

AJAX	Asynchronous JavaScript and XML
CSS	Cascading Style Sheets
DOM	Document Object Model
EJB	Enterprise Java Bean
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
Java EE	Java Platform, Enterprise Edition
JSF	Java Server Face
JSP	Java Server Page
RDBMS	Relational Data Base Management System
SQL	Structured Query Language
XHTML	Extensible HTML
XML	Extensible Markup Language
XSLT	Extensible Stylesheet Language Transformation

A Web-based Client-Server Visual Simulation Architecture

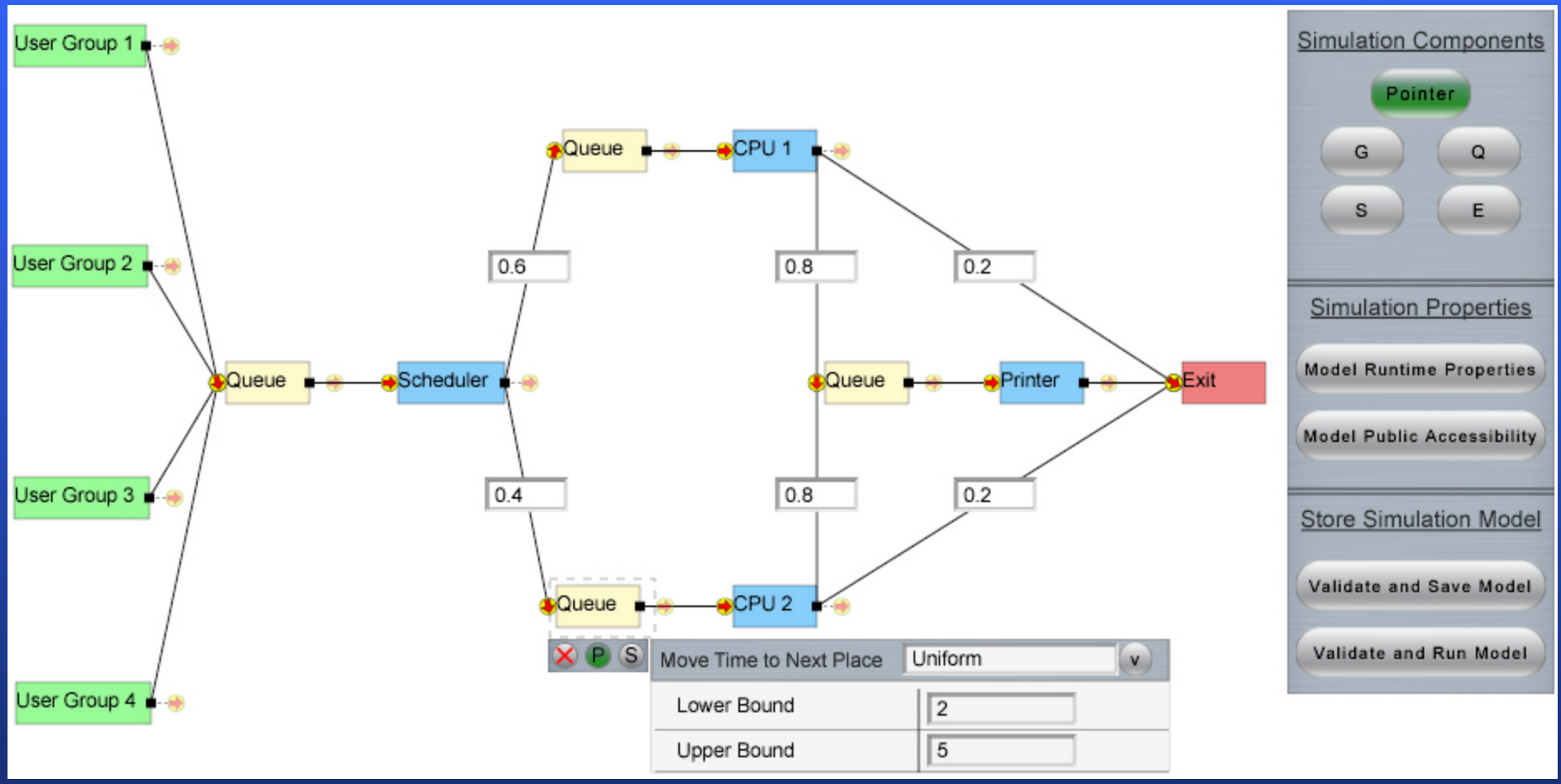


David S. Myers and Osman Balci (2009), "A Web-Based Visual Simulation Architecture," *International Journal of Modelling and Simulation 29*, 2, 137-148.

Web-based Queuing System Simulation System (WebQS3)

WebQS3 Editor

Play the WebQS3 Editor Video

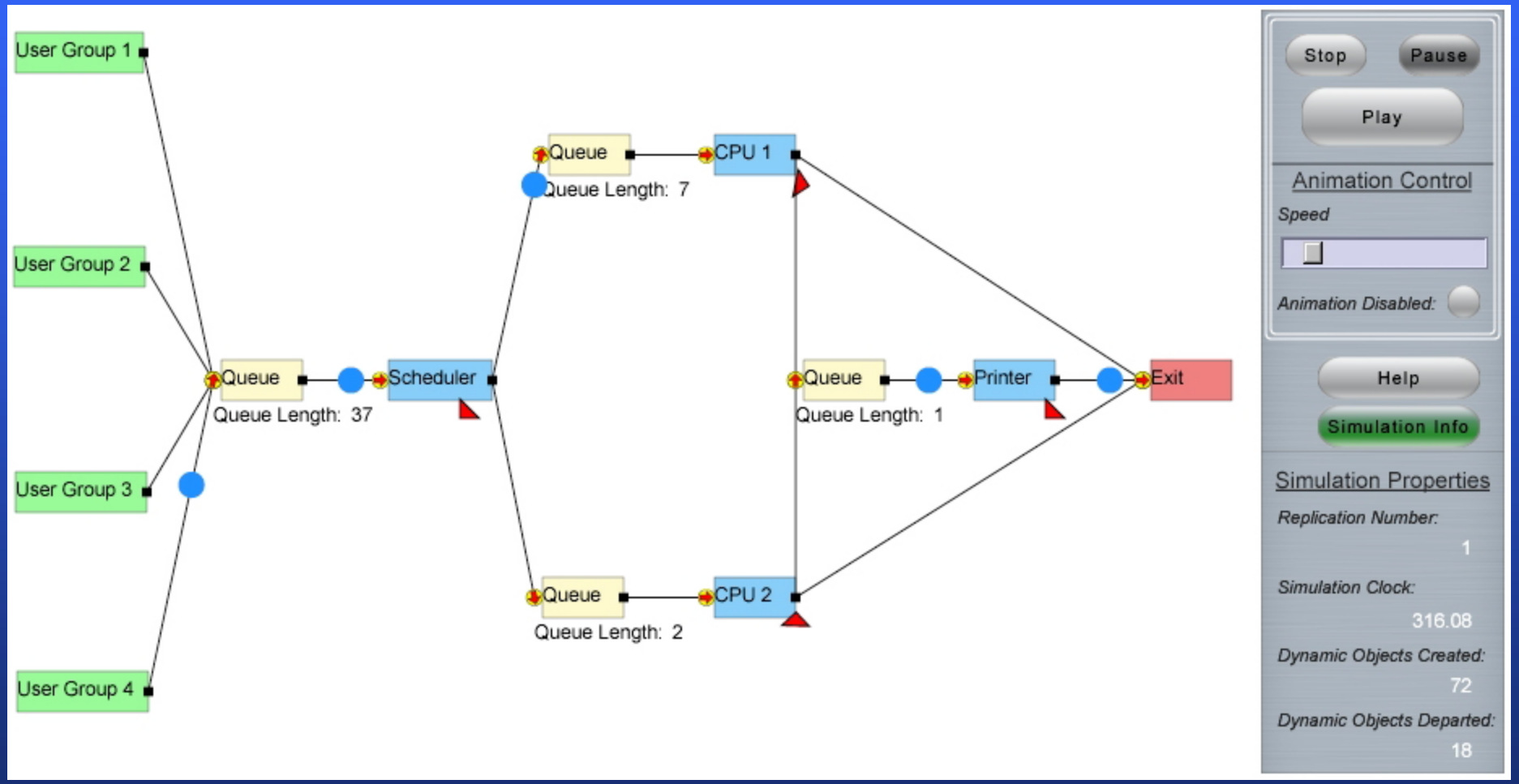


David S. Myers and Osman Balci (2009), "A Web-Based Visual Simulation Architecture," *International Journal of Modelling and Simulation 29*, 2, 137-148.

Web-based Queuing System Simulation System (WebQS3)

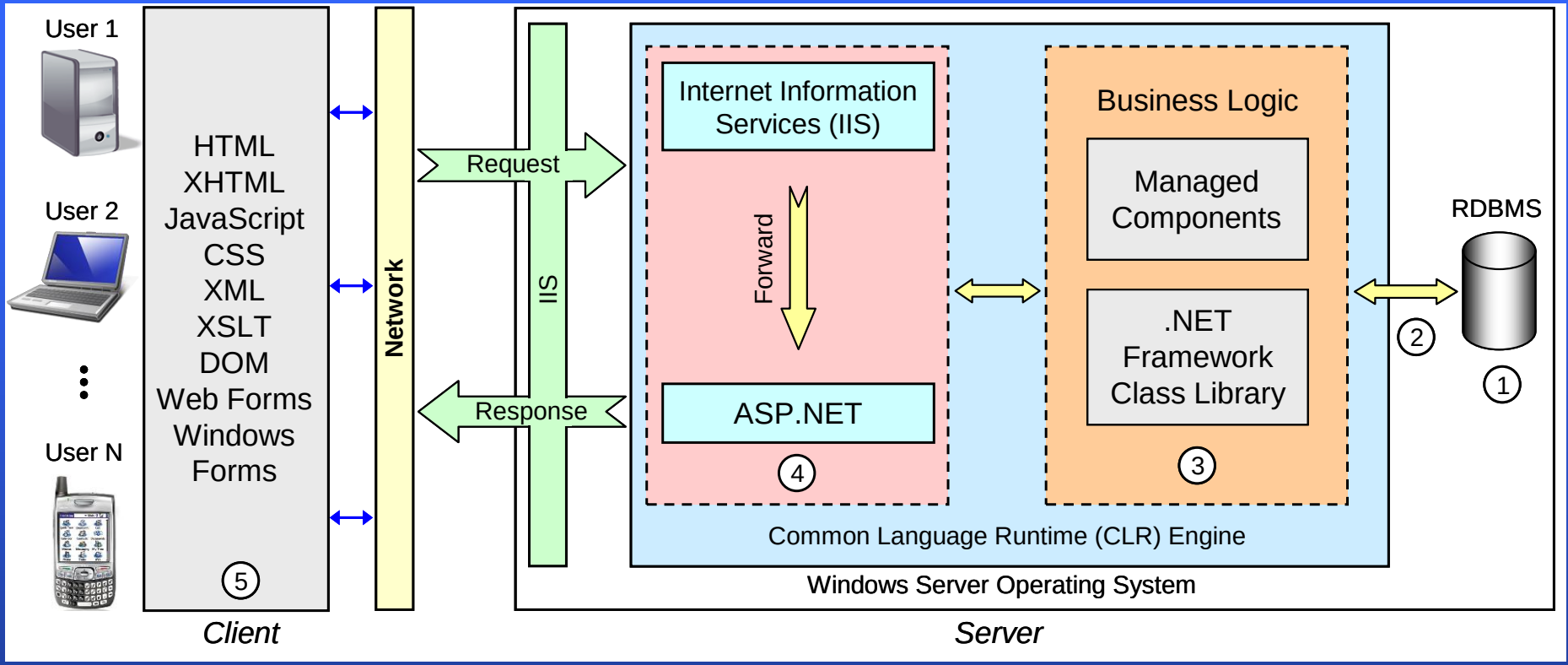
WebQS3 Simulator

Play the WebQS3 Simulator Video



David S. Myers and Osman Balci (2009), "A Web-Based Visual Simulation Architecture," *International Journal of Modelling and Simulation 29*, 2, 137-148.

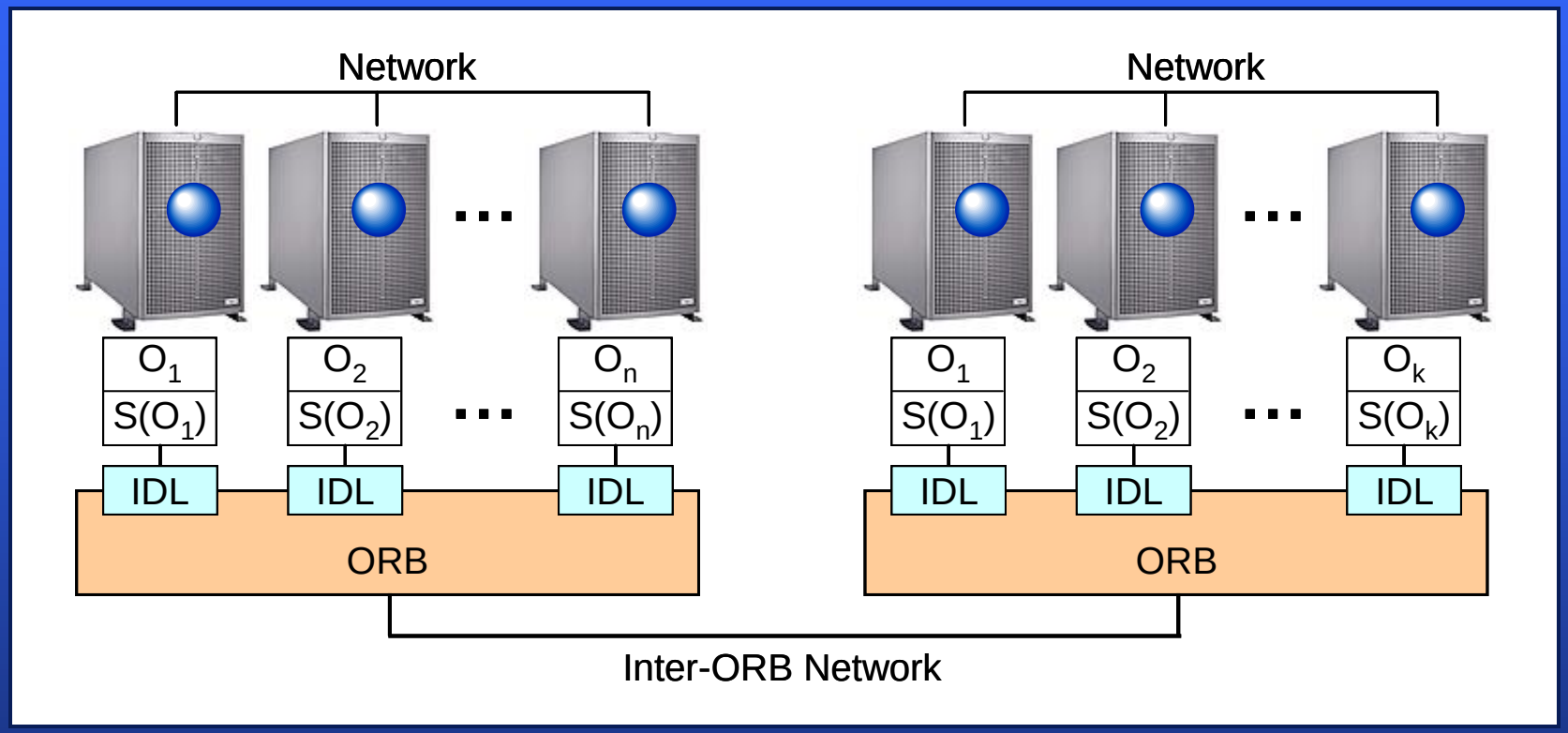
Microsoft .NET-based Client-Server Architecture



1. **Data Source Layer** (Relational DBMS such as DB2, Oracle, SQL Server)
2. **Data Mapping Layer** (ActiveX Data Objects - ADO.NET)
3. **Business Logic Layer / Domain Logic Layer** (Managed Components, .NET Framework Class Library)
4. **Web Container Layer** (IIS, Active Server Pages - ASP.NET)
5. **Client Presentation Layer** (JavaScript, XML, AJAX, XHTML, CSS, XSLT, DOM, Web Forms, Windows Forms)

Distributed Objects Architecture (DOA)

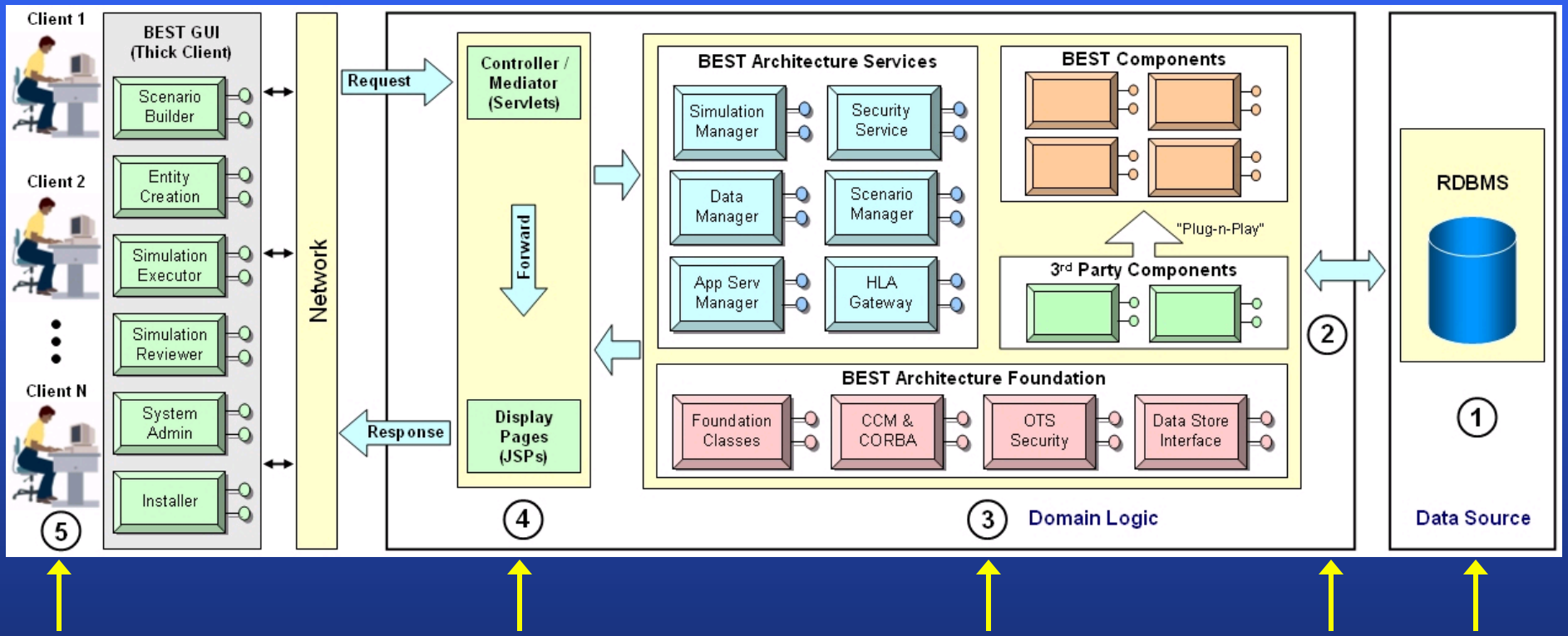
1. Common Object Request Broker Architecture (CORBA)



- Interface Definition Language (IDL)
- Object Request Broker (ORB)

2. Microsoft's Distributed Component Object Model (DCOM)

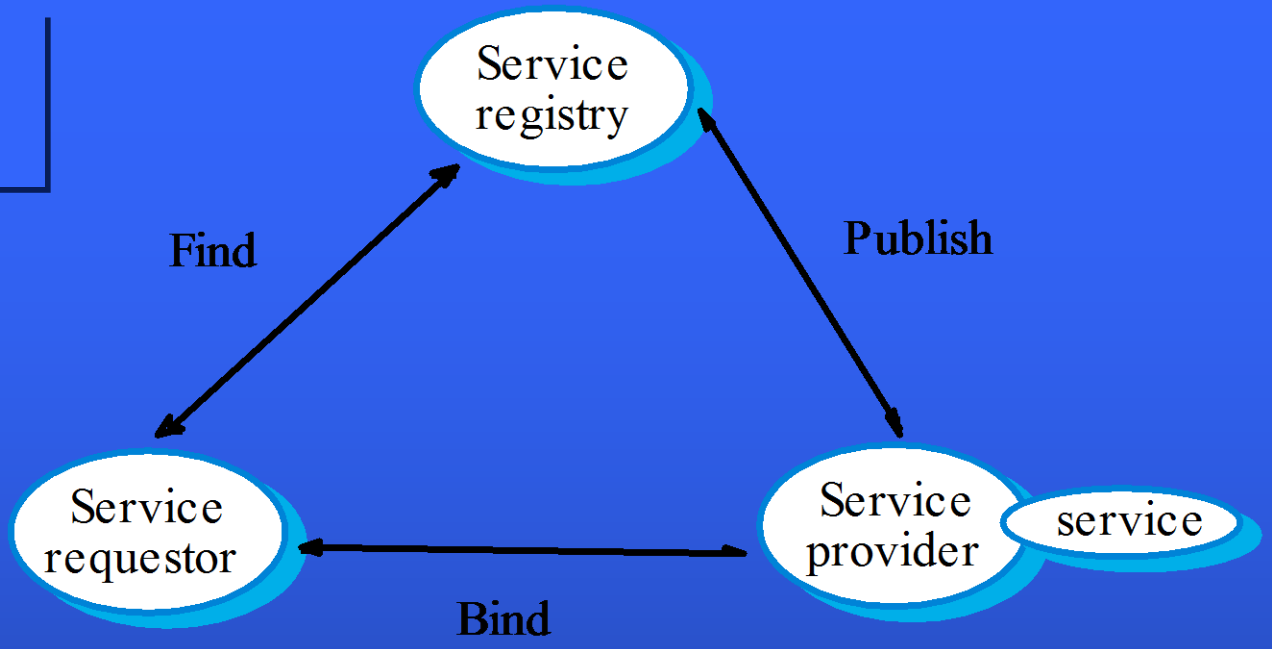
An Example Composition of CSA and DOA



Client-Server Architecture

BEST = Battlespace Environment and Signatures Toolkit
A simulation integrated development environment (IDE)

Service-Oriented Architecture

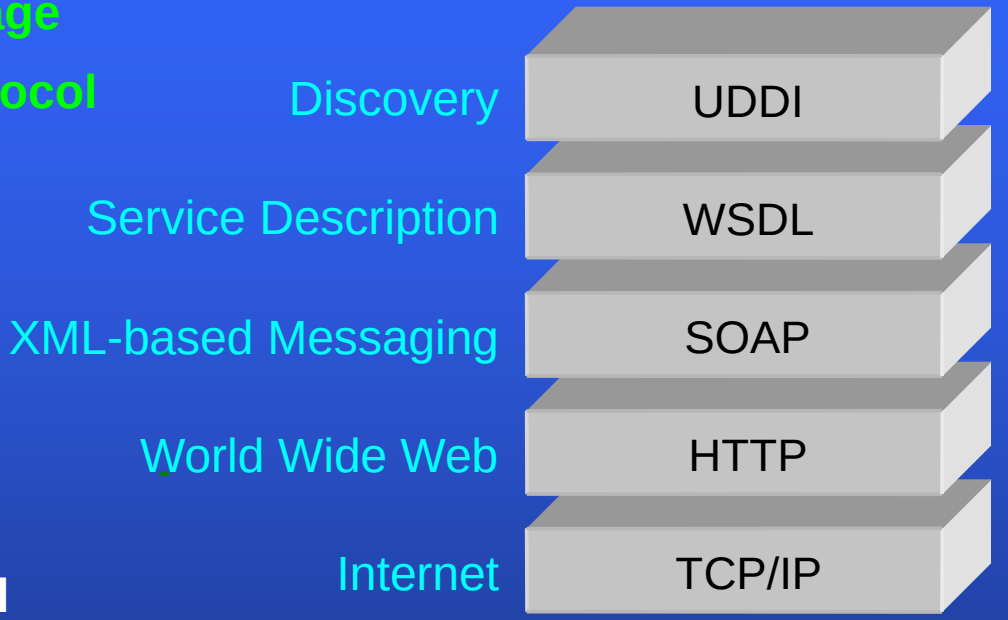


- Provider independence.
- Public advertising of service availability.
- Potentially, run-time service binding.
- Opportunistic construction of new services through composition.
- Pay for use of services.
- Smaller, more compact applications.
- Reactive and adaptive applications.

Service-Oriented Architecture (SOA)

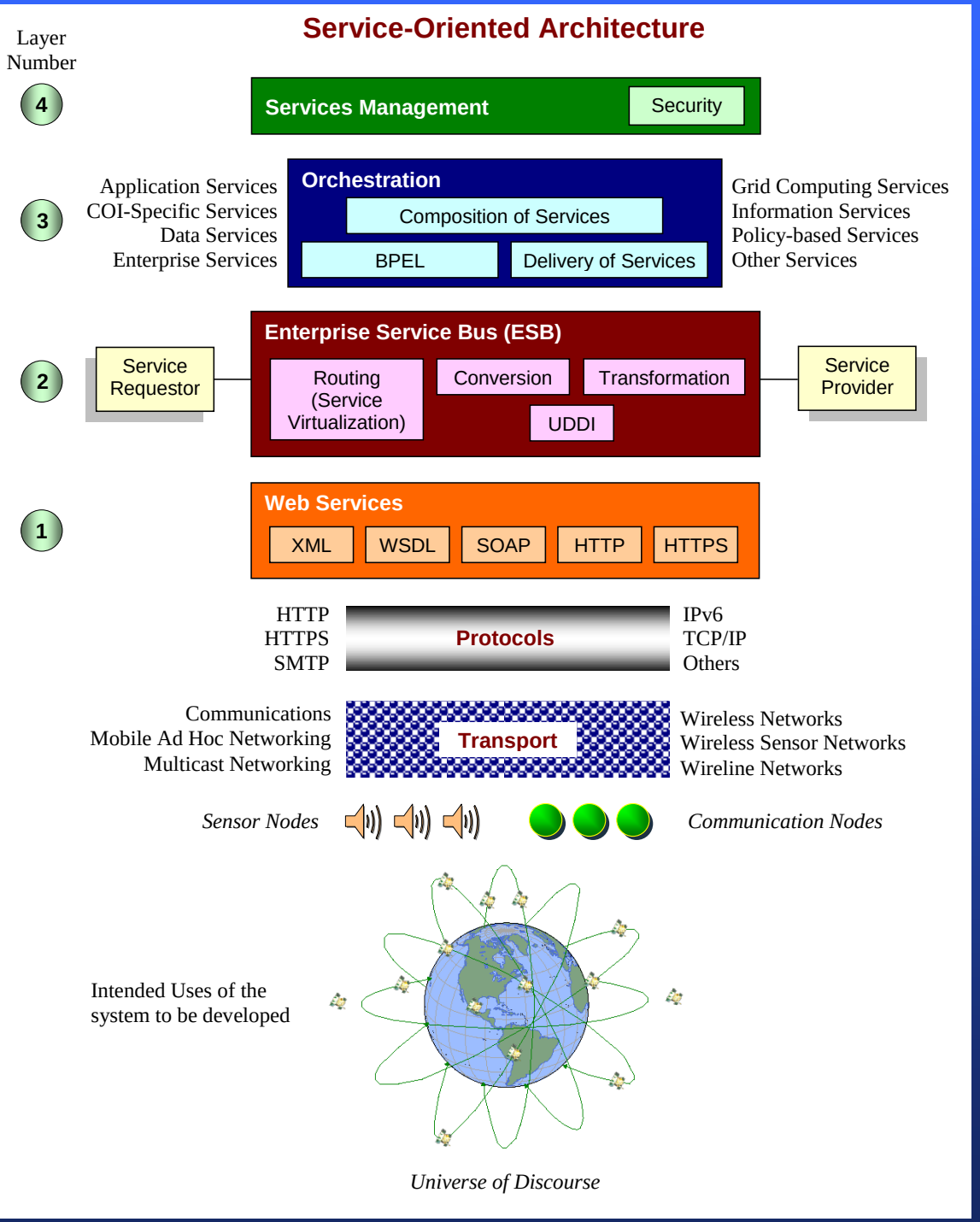
Web Services-based SOA

- **XML:** eXtensible Markup Language
- **TCP/IP:** Transmission Control Protocol / Internet Protocol
- **HTTP:** HyperText Transfer Protocol
- **SOAP:** Simple Object Access Protocol, an XML-based messaging protocol used to encode the information in web service request and response messages before sending them over a network.

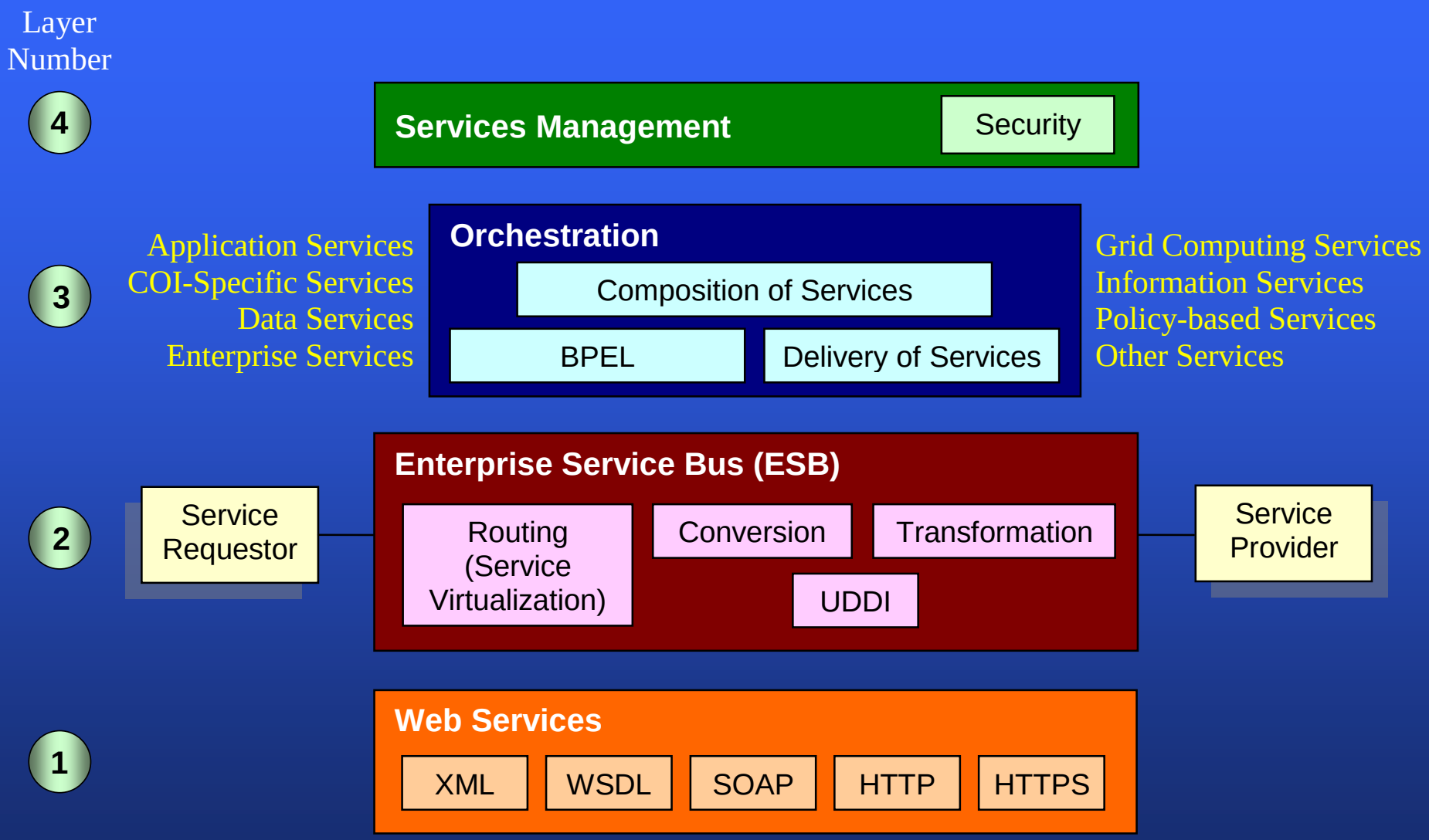


- **WSDL:** Web Services Description Language, an XML-formatted language used to describe a Web service's capabilities.
- **UDDI:** Universal Description, Discovery and Integration, a web-based distributed directory that enables listing of web services and discovering each other, similar to a traditional phone book's yellow and white pages.

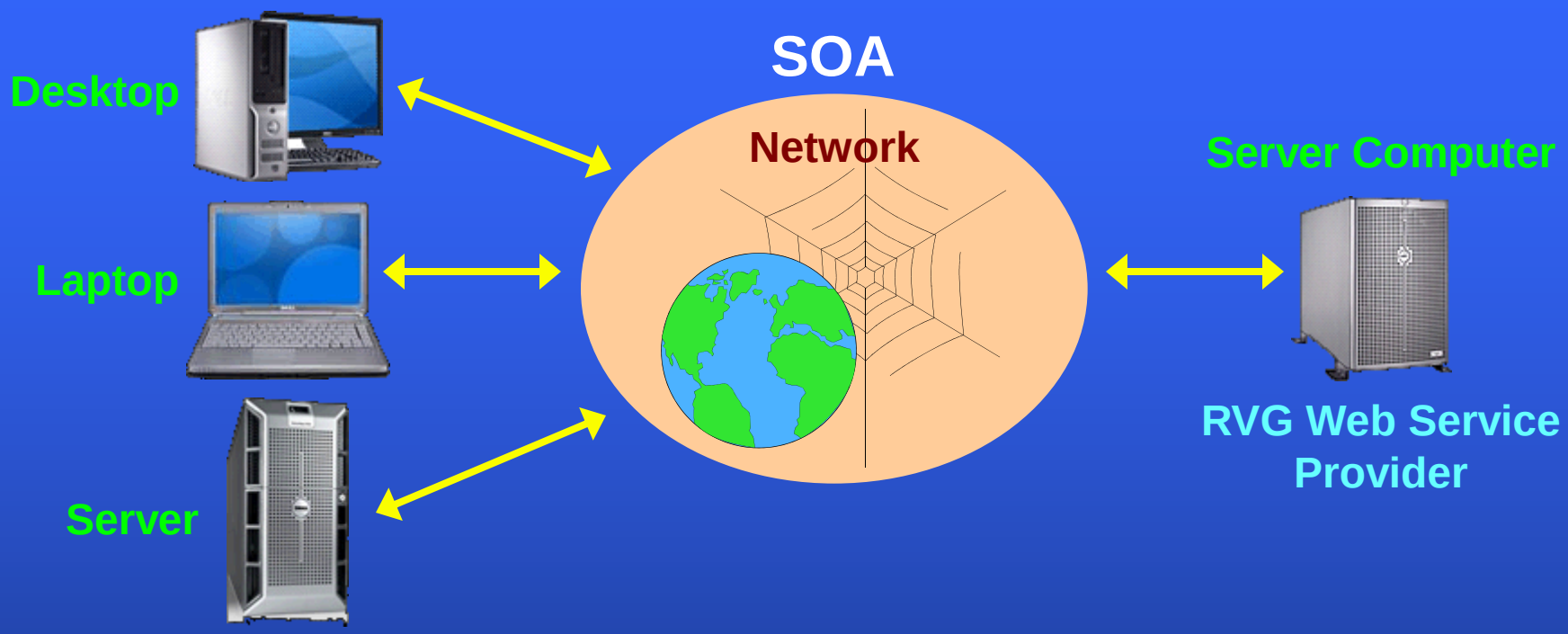
Service-Oriented Architecture



Service-Oriented Architecture (SOA)



Random Variate Generation (RVG) Web Service for Stochastic Simulations

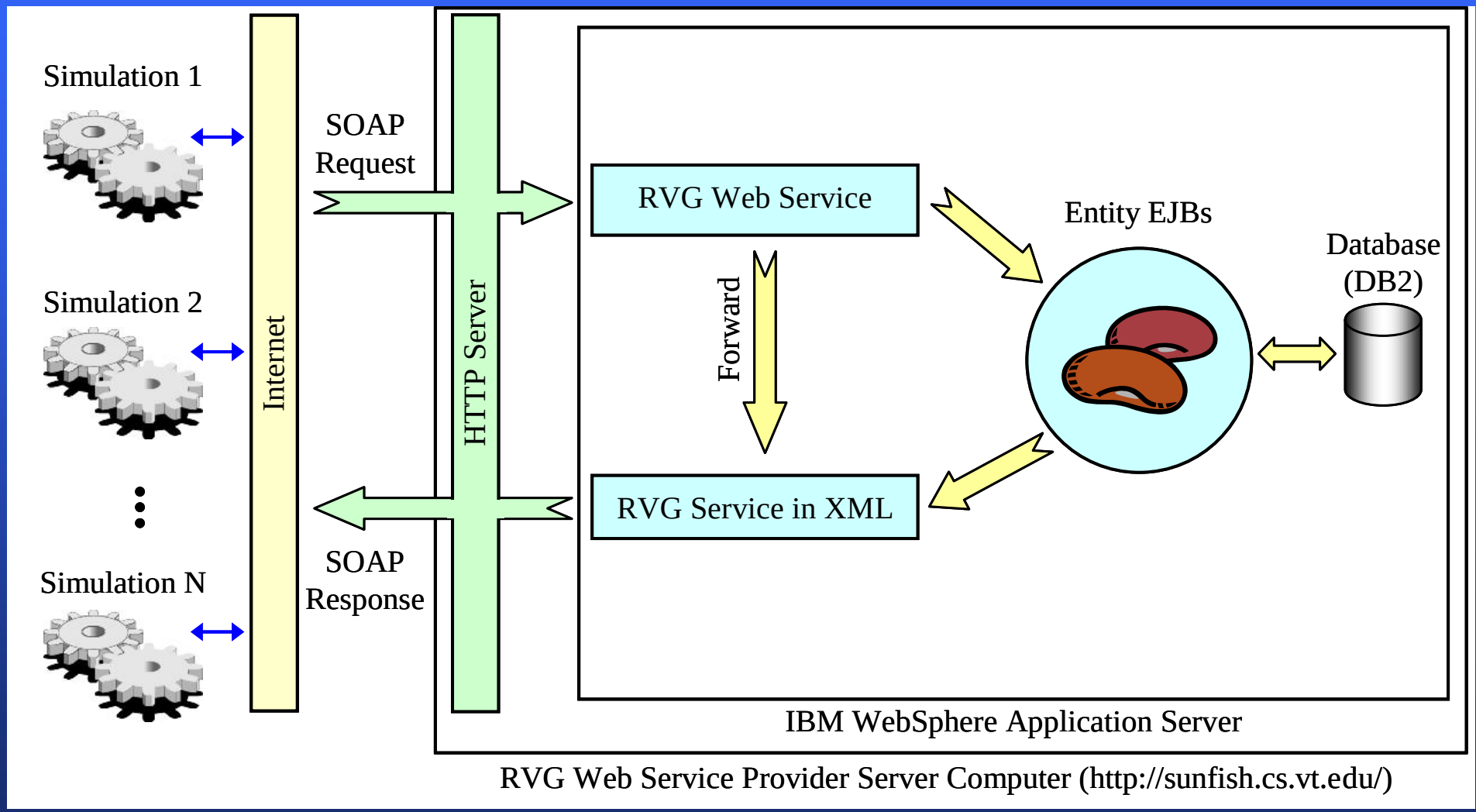


A simulation application running on a client or server computer sends an RVG service request as an XML file over the network to the RVG web service provider software running on a server computer.

Mohammad Sabah and Osman Balci (2005), "Web-based Random Variate Generation for Stochastic Simulations," *International Journal of Simulation and Process Modelling 1*, 1-2, 16-25.

Random Variate Generation (RVG) Web Service for Stochastic Simulations

Play the Video



Mohammad Sabah and Osman Balci (2005), "Web-based Random Variate Generation for Stochastic Simulations," *International Journal of Simulation and Process Modelling 1*, 1-2, 16-25.

High Level Architecture (HLA)

- **HLA is a general purpose architecture that facilitates interoperability among network-centric M&S applications and enables reuse of M&S applications and their components.**
- **HLA-compliant M&S applications running on different hardware platforms can interoperate with each other over a network.**
- **Interoperability among the network-centric M&S applications is enabled by a RunTime Infrastructure (RTI).**
- **HLA is an IEEE Standard, a DoD Standard, and a NATO Standard.**

- IEEE (2000), *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules*. IEEE Standard No. 1516-2000.
- IEEE (2000), *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) – Federate Interface Specification*. IEEE Standard No. 1516.1-2000.
- IEEE (2000), *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) – Object Model Template (OMT) Specification*. IEEE Standard No. 1516.2-2000.
- IEEE (2003), *IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP).* IEEE Standard No. 1516.3-2003.

HLA Components

HLA is composed of three major components:

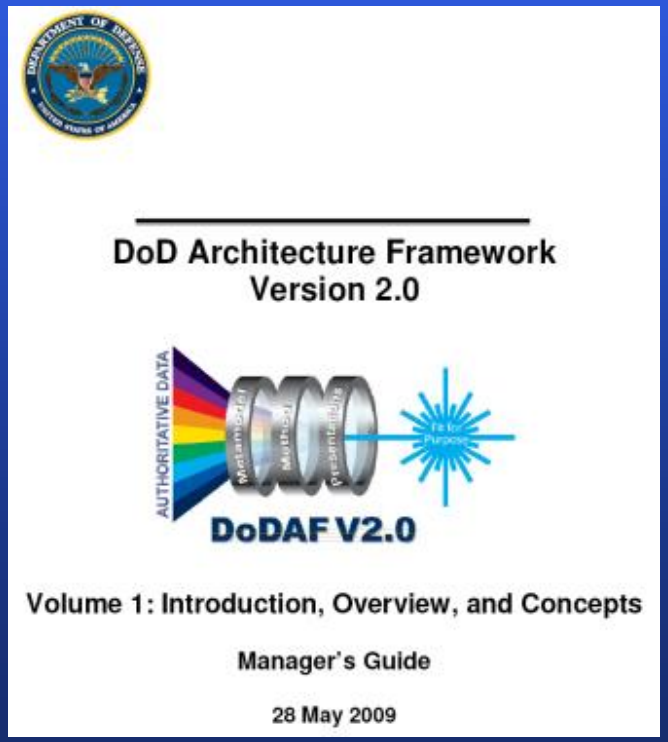
- **HLA Rules**
  - A set of ten basic rules that together describe the general principles defining the HLA.
- **HLA Interface Specification**
  - A description of the functional interface between simulations (federates) and the HLA runtime infrastructure (RTI).
- **HLA Object Model Template (OMT)**
  - A specification of the common format and structure for documenting HLA object models.

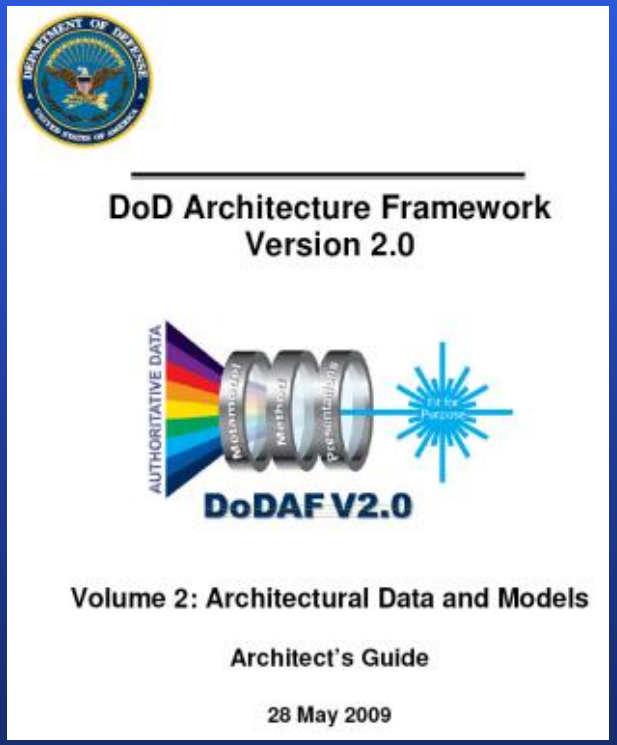
What is DoDAF?

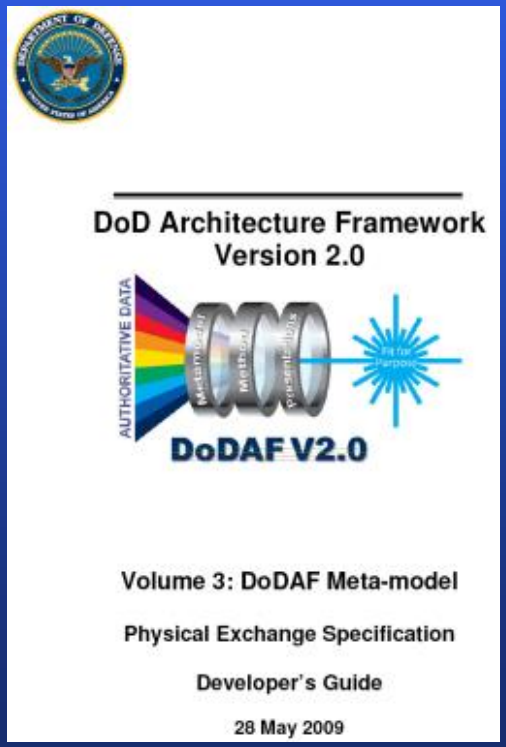
- The U.S. Department of Defense (DoD) Architecture Framework (DoDAF):
  - "Provides a foundational framework for developing and representing architecture descriptions that ensure a common denominator for understanding, comparing, and integrating architectures across organizational, Joint, and multinational boundaries.
  - It establishes data element definitions, rules, and relationships and a baseline set of products for consistent development of systems, integrated, or federated architectures.
  - These architecture descriptions may include Families of Systems (FoSs), Systems of Systems (SoSs), and net-centric capabilities for interoperating and interacting in the Network-Centric Environment."

DoDAF Version 2.0

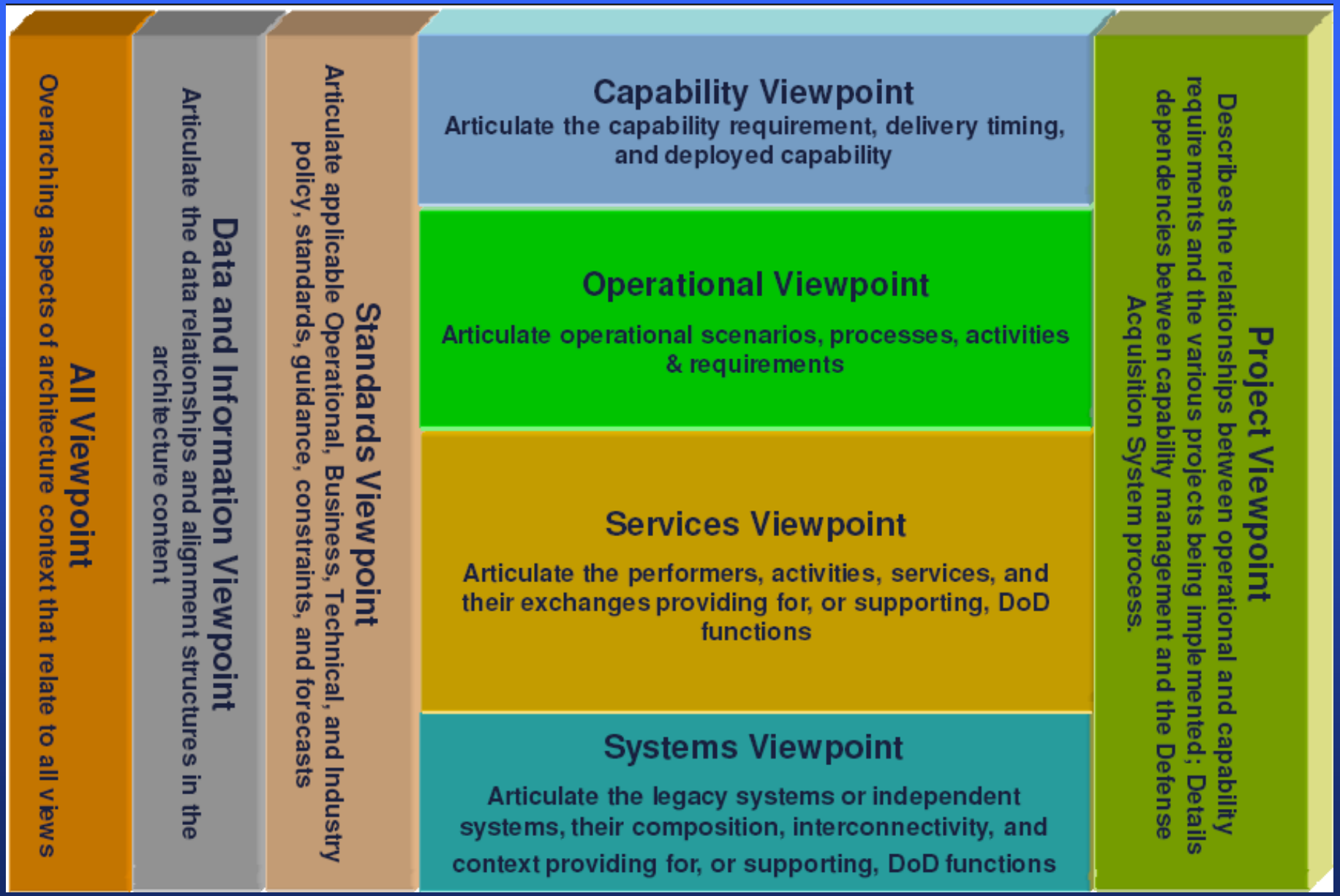
- Volume 1: Introduction, Overview, and Concepts - Manager's Guide
- Volume 2: Architectural Data and Models - Architect's Guide
- Volume 3: DoDAF Meta-model, Physical Exchange Specification - Developer's Guide







Architecture Viewpoints in DoDAF V2.0



DoDAF V2.0 Models

Models	Descriptions
AV-1: Overview and Summary Information	Describes a Project's Visions, Goals, Objectives, Plans, Activities, Events, Conditions, Measures, Effects (Outcomes), and produced objects.
AV-2: Integrated Dictionary	An architectural data repository with definitions of all terms used throughout the architectural data and presentations.
CV-1: Vision	The overall vision for transformational endeavors, which provides a strategic context for the capabilities described and a high-level scope.
CV-2: Capability Taxonomy	A hierarchy of capabilities which specifies all the capabilities that are referenced throughout one or more Architectural Descriptions.
CV-3: Capability Phasing	The planned achievement of capability at different points in time or during specific periods of time. The CV-3 shows the capability phasing in terms of the activities, conditions, desired effects, rules complied with, resource consumption and production, and measures, without regard to the performer and location solutions.
CV-4: Capability Dependencies	The dependencies between planned capabilities and the definition of logical groupings of capabilities.
CV-5: Capability to Organizational Development Mapping	The fulfillment of capability requirements shows the planned capability deployment and interconnection for a particular capability phase. The CV-5 shows the planned solution for the phase in terms of performers and locations and their associated concepts.
CV-6: Capability to Operational Activities Mapping	A mapping between the capabilities required and the operational activities that those capabilities support.
CV-7: Capability to Services Mapping	A mapping between the capabilities and the services that these capabilities enable.

DoDAF V2.0 Models

DIV-1:Conceptual Data Model	The required high level data concepts and their relationships.
DIV-2: Logical Data Model	The documentation of the data requirements and structural business process (activity) rules. In DoDAF V1.5, this was the OV-7.
DIV-3: Physical Data Model	The physical implementation format of the Logical Data Model entities, e.g., message formats, file structures, physical schema. In DoDAF V1.5, this was the SV-11.
OV-1: High Level Operational Concept Graphic	The high-level graphical/textual description of the operational concept.
OV-2: Operational Resource Flow Description	A description of the resource flows exchanged between operational activities.
OV-3: Operational Resource Flow Matrix	A description of the resources exchanged and the relevant attributes of the exchanges.
OV-4: Organizational Relationships Chart	The organizational context, role or other relationships among organizations.

DoDAF V2.0 Models

Models	Descriptions
OV-5a: Operational Activity Decomposition Tree	The capabilities and activities (operational activities) organized in an hierarchal structure.
OV-5b: Operational Activity Model	The context of capabilities and activities (operational activities) and their relationships among activities, inputs, and outputs; Additional data can show cost, performers or other pertinent information.
OV-6a: Operational Rules Model	One of three models used to describe activity (operational activity). It identifies business rules that constrain operations.
OV-6b: State Transition Description	One of three models used to describe operational activity (activity). It identifies business process (activity) responses to events (usually, very short activities).
OV-6c: Event-Trace Description	One of three models used to describe operational activity (activity). It traces actions in a scenario or sequence of events.
PV-1: Project Portfolio Relationships	Describes the dependency relationships between the organizations and projects and the organizational structures needed to manage a portfolio of projects.
PV-2: Project Timelines	A timeline perspective on programs or projects, with the key milestones and interdependencies.
PV-3: Project to Capability Mapping	A mapping of programs and projects to capabilities to show how the specific projects and program elements help to achieve a capability.

DoDAF V2.0 Models

SvcV-1 Services Context Description	The identification of services, service items, and their interconnections.
SvcV-2 Services Resource Flow Description	A description of resource flows exchanged between services.
SvcV-3a Systems-Services Matrix	The relationships among or between systems and services in a given Architectural Description.
SvcV-3b Services-Services Matrix	The relationships among services in a given Architectural Description. It can be designed to show relationships of interest, (e.g., service-type interfaces, planned vs. existing interfaces).
SvcV-4 Services Functionality Description	The functions performed by services and the service data flows among service functions (activities)
SvcV-5 Operational Activity to Services Traceability Matrix	A mapping of services (activities) back to operational activities (activities).
SvcV-6 Services Resource Flow Matrix	It provides details of service resource flow elements being exchanged between services and the attributes of that exchange.
SvcV-7 Services Measures Matrix	The measures (metrics) of Services Model elements for the appropriate time frame(s).
SvcV-8 Services Evolution Description	The planned incremental steps toward migrating a suite of services to a more efficient suite or toward evolving current services to a future implementation.

DoDAF V2.0 Models

Models	Descriptions
SvcV-9 Services Technology & Skills Forecast	The emerging technologies, software/hardware products, and skills that are expected to be available in a given set of time frames and that will affect future service development.
SvcV-10a Services Rules Model	One of three models used to describe service functionality. It identifies constraints that are imposed on systems functionality due to some aspect of system design or implementation.
SvcV-10b Services State Transition Description	One of three models used to describe service functionality. It identifies responses of services to events.
SvcV-10c Services Event-Trace Description	One of three models used to describe service functionality. It identifies service-specific refinements of critical sequences of events described in the Operational Viewpoint.
StdV-1 Standards Profile	The listing of standards that apply to solution elements.
StdV-2 Standards Forecast	The description of emerging standards and potential impact on current solution elements, within a set of time frames.
SV-1 Systems Interface Description	The identification of systems, system items, and their interconnections.
SV-2 Systems Resource Flow Description	A description of resource flows exchanged between systems.
SV-3 Systems-Systems Matrix	The relationships among systems in a given Architectural Description. It can be designed to show relationships of interest, (e.g., system-type interfaces, planned vs. existing interfaces).
SV-4 Systems Functionality Description	The functions (activities) performed by systems and the system data flows among system functions (activities).

DoDAF V2.0 Models

SV-5a Operational Activity to Systems Function Traceability Matrix	A mapping of system functions (activities) back to operational activities (activities).
SV-5b Operational Activity to Systems Traceability Matrix	A mapping of systems back to capabilities or operational activities (activities).
SV-6 Systems Resource Flow Matrix	Provides details of system resource flow elements being exchanged between systems and the attributes of that exchange.
SV-7 Systems Measures Matrix	The measures (metrics) of Systems Model elements for the appropriate timeframe(s).
SV-8 Systems Evolution Description	The planned incremental steps toward migrating a suite of systems to a more efficient suite, or toward evolving a current system to a future implementation.
SV-9 Systems Technology & Skills Forecast	The emerging technologies, software/hardware products, and skills that are expected to be available in a given set of time frames and that will affect future system development.
SV-10a Systems Rules Model	One of three models used to describe system functionality. It identifies constraints that are imposed on systems functionality due to some aspect of system design or implementation.
SV-10b Systems State Transition Description	One of three models used to describe system functionality. It identifies responses of systems to events.
SV-10c Systems Event-Trace Description	One of three models used to describe system functionality. It identifies system-specific refinements of critical sequences of events described in the Operational Viewpoint.