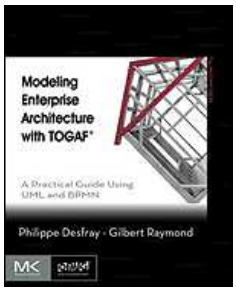


Modeling Enterprise Architecture with TOGAF: A Practical Guide Using UML and BPMN

by Philippe Desfray and Gilbert Raymond

Chapter 10: Models for Phase D—Technology Architecture

Technology architecture associates application components from application architecture with technology components representing software and hardware components. Its components are generally acquired in the marketplace and can be assembled and configured to constitute the enterprise's technological infrastructure. Technology architecture provides a more concrete view of the way in which application components will be realized and deployed. It enables the migration problems that can arise between the different steps of the IS evolution path to be studied earlier. It provides a more precise means of evaluating responses to constraints (nonfunctional requirements) concerning the IS, notably by estimating hardware and network sizing needs or by setting up server or storage redundancy. Technology architecture concentrates on logistical and location problems related to hardware location, IS management capabilities, and the sites where the different parts of the IS are used. Technology architecture also ensures the delivered application components work together, confirming that the required business integration is supported.

10.1 PHASE D ARTIFACTS

10.1.1 Nature of Phase D Artifacts: Technology Architecture

Among the diagrams presented, network computing hardware diagrams play a central role (Table 10.1). The other diagrams are derived from this type of diagram and provide views that focus on particular aspects. The need to implement a type of diagram also depends on which existing elements are retained (application architecture/technology architecture). If technology architecture is seen as a receptacle for as yet undeveloped application architecture, then the "Platform decomposition diagram" (not presented in this book) is particularly adapted. However, if application architecture is defined to study the best technical platform to host it, then the network computing hardware diagram will quickly be implemented to schematize the best adapted overall configuration (Figure 10.3).

Table 10.1: TOGAF Artifacts and Artifacts Presented in This Chapter

TOGAF Artifacts	Models Presented	Comments
Environment and location diagram	Environment and location diagram	
Processing diagram	Processing diagram	
Networked computing diagram	Network computing hardware diagram	
Platform decomposition diagram		Network computing hardware diagram view focused on technology platforms that support the IS
Communication engineering diagram		Network computing hardware diagram view focused on communication technologies (network, protocols, etc.)
Technology standards catalog		Lists technologies recommended across the enterprise
Technology portfolio catalog		Lists all the technologies used across the enterprise
Application technology matrix		Documents the mapping of applications to technology platforms; can be produced from network computing hardware diagrams

10.1.2 Essential Concepts Used in Technology Architecture

- 

Headquarters (location): Geographically determines where enterprise elements are deployed (organization units, IT hardware, actors, etc.).

- 

Site (location): Geographically defines where enterprise elements are deployed (organization units, IT hardware, actors, etc.). An enterprise frequently has headquarters and several sites.

- 

Server: Represents a hardware platform that can be connected to other servers and on which application components will be deployed.

- 

Workstation: Linked to an information system via network connections. Application components can also be deployed on them.

- 

Utility component: In the case of technology architecture, physical-level technology components will frequently be utility components. For example, this is how application servers, database servers, or business process servers, which themselves host application components, will be represented.

10.2 THE "ENVIRONMENT AND LOCATION DIAGRAM" ARTIFACT

Name	Environment and location diagram
Experts	Technical architects, business experts
Designers	Technical architects
Recipients	Business experts, technical architects, operations managers
Aim	To define the deployment of hardware and applications on different enterprise sites
Useful preliminary information	Application architecture, existing technical architecture, geographical organization of the enterprise, nonfunctional requirements

- 

Headquarter (location)

- 

Site (location)

- 

Interaction component

- 

IT hardware: Server

- 

Application component

- 

Workstation

- 

  Association: Here, this describes the connection between a site and its headquarters.

Figure 10.3 shows the definition of several hardware servers (for example, MVS server) and their network connections. In Figure 10.1, we can see that these servers are concentrated in the headquarters (Paris). Three different servers host the accounting ERP, the "TripPortfolioManager" application, and the trip reservation site.

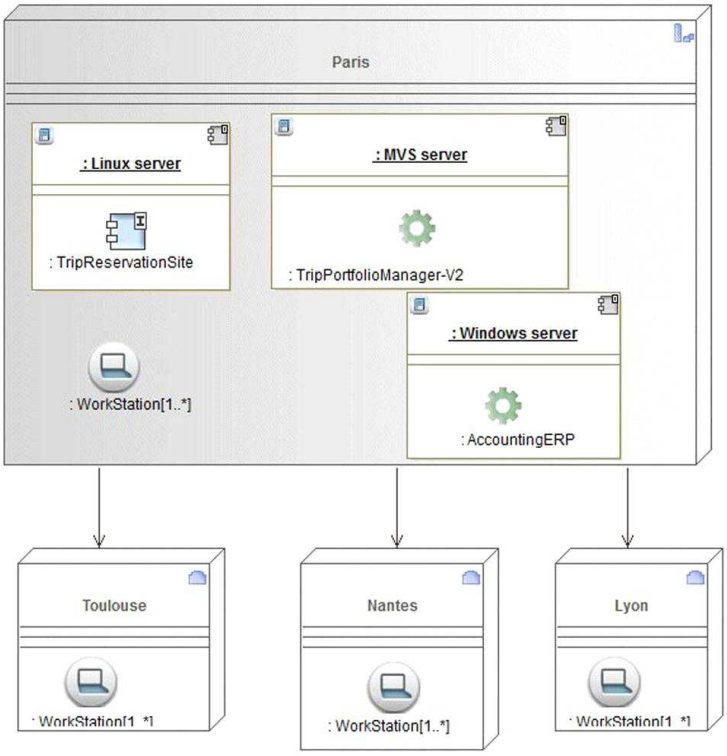


Figure 10.1: The main applications and devices are located in Paris

Environment and location diagrams define which locations host which applications. They identify which technologies and/or applications are used in which locations, and can determine which actors interact with these applications in which locations. This type of diagram also shows the existence and position of different deployment environments, including environments not destined for production, such as those destined for development and preproduction.

Because they focus on the representation of deployment on the enterprise's hardware and sites, environment and location diagrams use element embedding. Hardware shown as being embedded inside a location is geographically situated in this location, while an application shown as being embedded in hardware indicates where it is physically deployed. Shortcuts can be made simply by presenting applications in geographical locations, if the focus is not on the hardware. For example, in the case of cloud computing applications, we can simply express the locations where certain applications are predominantly used. Typically, this type of diagram represents the location where each server is located, and shows the server on which each application is run.

10.3 THE "PROCESSING DIAGRAM" ARTIFACT

Name	Processing diagram
Experts	Technical architects
Designers	Technical architects
Recipients	Operations engineers, developers
Aim	To provide details on the technical components necessary to run application architecture. To define the application architecture deployment mode
Useful preliminary information	Application architecture, existing technical architecture, technological choices

- 

Utility component: Here, this represents a physical technical component, such as an application server.

- 

Process component

- 

Interaction component

- 

Entity component

- 

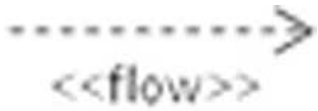
Application component

- 

Workstation

- 

Association

- 

Information flow: Defines the flows exchanged between different servers via their networks.

Figure 10.2 presents three technical components, which act as servers hosting different application components defined by the application architecture: a web server is required for the site and a business process server hosts process components, while the other components are hosted by an application server.

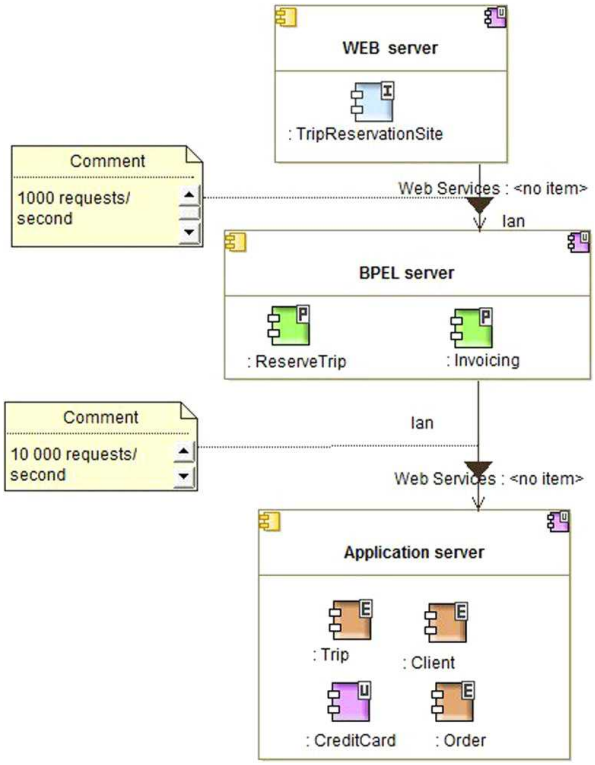


Figure 10.2: A processing diagram. Deployment of application components in different types of application servers

Processing diagrams focus on deployable units of code or application configuration, and on the way in which these are deployed on technical platforms. A deployment unit constitutes a group of business functions, services, or application components. The following questions are dealt with by processing diagrams:

- Which physical groups are implemented to constitute a deployment unit?
- Which deployment units are interconnected (network) using which protocols?
- How do application configuration and usage modes translate into capability or load increase requirements for the different technical components?

Processing diagrams will use deployment in a more generic way than network computing hardware diagrams by focusing on deployment units. Deployment units can be presented in the form of instances of components where application components are deployed, or by utility components, which host the application components deployed. For example, this is the case for an application server, which is a technical utility component in which the application components run by the server are deployed.

In SOA architecture, application components will frequently represent their own deployment units, and this facilitates management and traceability. Sometimes assemblies of application components into a deployment unit may also be defined.

Dedicated links between these deployment units will represent connections, while information flows will be used to indicate the nature of the information exchanged.

Information on hardware and technical component capability requirements will be provided in these diagrams.

However, the deployment configuration presented in the example in Figure 10.2 remains independent of future deployment on physical servers.

10.4 THE "NETWORK COMPUTING HARDWARE DIAGRAM" ARTIFACT

Name	Network computing hardware diagram
Experts	Technical architects, system and network engineers
Designers	Technical architects
Recipients	System and network engineers, developers
Aim	To define the entire network, hardware, and technology architecture to meet functional and nonfunctional requirements
Useful preliminary information	Application architecture, technological choices, existing technical architecture

- 

Interaction component

- 

IT material: Server

- 

Entity component

- 

Application component

- 

Workstation

- 

Database component

- 

Process component

- 

Internal actor

- 

Internet connection

- 

External actor

- 

Utility component: Here, this represents a physical technical component, such as an application server.

- 

Association: Here, this describes the network connection between two IT hardware components.

IT systems previously built around "mainframe" architecture evolved into client/server systems before adapting to "eBusiness" and J2EE-type architectures, and then finally migrating to architectures that rely heavily on distributed systems on networks, with secure zones. Most current applications have a "front end" and are based on a multilayer architecture that separates web presentation, business logic, and data management aspects. It is common practice for applications to be based on common technical or software infrastructures. Therefore, it is critically important to document correspondence between logical applications and the technical components (such as servers) that support applications both in development and production environments.

Thus, the aim of network computing hardware diagrams is to present the logical view deployed in a distributed environment on the network of physical and software servers.

Physical servers are represented. In some cases, these will be instances of real servers, and in other cases server typologies. Sometimes these correspond to known, located hardware, while in "cloud computing"-type configurations, the physical servers are not known and can be increased according to load increase needs. In all cases, their typology must be known.

On these "physical" servers, the software used in the infrastructure is presented, for example, web servers or servers for business processes. Finally, application components identified during the definition of application architecture are deployed on physical servers or software infrastructures.

Connections between these hardware or software elements are also modeled, either as physical network connections or software bus connections.

It is useful to link roles and actors identified in business architecture in order to position them with regard to the system and to ensure that the hardware available will suit users.

In the diagram shown in Figure 10.3, deployments (software servers in physical servers, application components in software servers, etc.) are represented through element embedding. Instances of application components are therefore presented in the context of a particular server, and one component can have several instances in different contexts (servers).

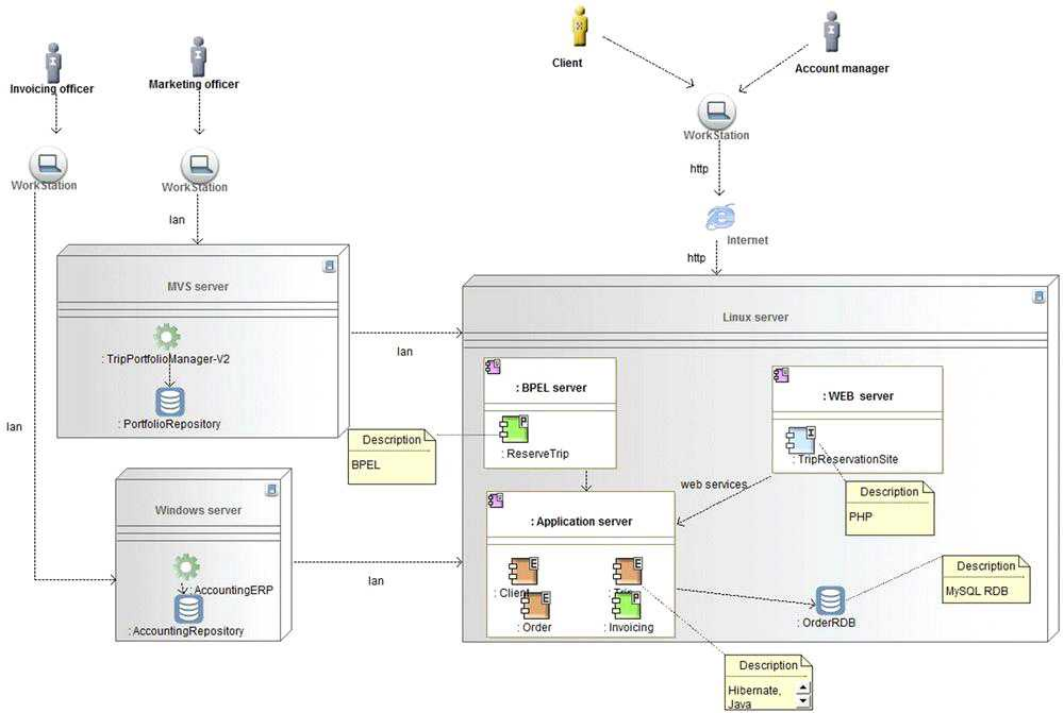


Figure 10.3: Diagram showing component deployment, network connections between servers, and the relative position of actors

This creates a diagram that is highly representative of the future or current configuration of the system.

10.5 FUNDAMENTAL CONCEPTS

The following fundamental concepts were introduced in this chapter:

- Technology architecture: Describes the logical software and hardware capabilities that are required to support the deployment of business, data, and application services.