Appendix D

Deploying the Software Solution

Objectives

Upon completion of this appendix, you should be able to:

- Explain system deployment
- Create an instance Deployment diagram

Process Map

This module describes a few activities of the Deployment workflow. Figure D-1 shows the activity and artifact discussed in this appendix.

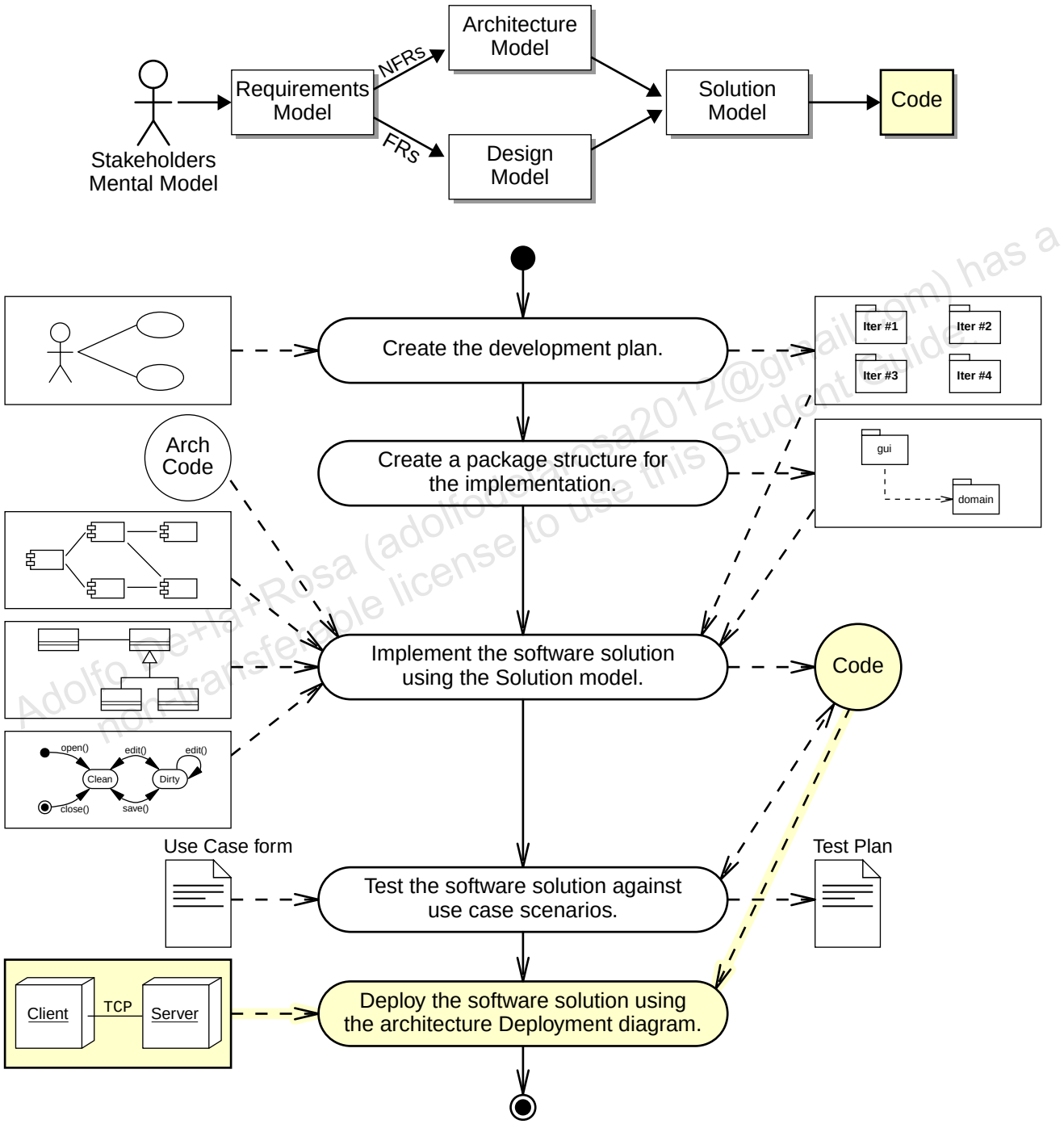

Figure D-1 Deployment Workflow Process Map

Explaining System Deployment

The purpose of the Deployment workflow is to place the software system into the production environment and to make sure that all users have been properly trained to use the system. Relative to the Unified Process, the Deployment workflow occurs in the Transition phase, but some of these activities can be performed in the Construction phase.

Deployment of the software solution is usually performed by the client company's system administration team. The development team usually only has a consulting role.

The following are a few of the many possible deployment activities:

- Purchase necessary hardware and software.

  The Deployment diagram and the Tiers/Layers diagram record the necessary hardware machines and software packages that need to be acquired to construct the production environment. The descriptive Deployment diagram must be transformed into an instance Deployment diagram which specifies the hardware constraints for each machine in the system.

- Construct the networking infrastructure.

  Buy and connect the networking hardware to construct the topology specified in the Deployment diagram. This might require buying additional hardware and software for routers, firewalls, and so on.

- Install the third-party and custom software.

  Using the architecture tiers and layers Package diagram as a guide, install all necessary software on the production hardware.

- Migrate legacy data to new system (if necessary).

  Convert any legacy data of the old system into the new system. This usually requires a set of scripts or programs for performing this conversion. This can be a complex task, so it must be scheduled during the Construction and Transition phases.

- Test the installation.

  Perform all tests (functional and nonfunctional) on the production environment using the test databases instead of the production databases.

- Train the staff.

  Create all necessary documentation and train the users on the new system. This activity is extremely important and should be performed well in advance of the *go live* date.

- Go live.

  It is important to celebrate the creation of the system. Many organizations throw a party for the development team (as well as the client-side stakeholders). This is not frivolous because it boosts the morale of the development team and builds better rapport between the client-side stakeholders and the development team for future projects.

Using the Deployment Diagram

The following describe how to use the Deployment diagram in the Deployment workflow:

- The Deployment diagram of the system is the architect's vision of how the system should be deployed.

  The Deployment diagram specifies the production environment as specified by the Architect. This vision might need to be altered to integrate with the client's existing production environment.

- The Deployment diagram should provide both the structure of the network and the set of software components.

  It is important to note that the Deployment diagram is probably not comprehensive. The Deployment Specialists should be able to take the Deployment diagram and determine the missing pieces, such as additional networking hardware.

- The "descriptive" Deployment diagram (created in the Elaboration phase) needs to be elaborated with details in the "instance" Deployment diagram before deployment can proceed.

  The instance Deployment diagram shows the specifications for the hardware nodes in the descriptive Deployment diagram. This will guide the client's purchases or upgrades.

Creating an Instance Deployment Diagram

Creating an instance Deployment diagram from a descriptive Deployment diagram requires adding the following features:

- Detailed network configuration

  The descriptive Deployment diagram only shows the links (network connections) between the major hardware nodes. In the instance Deployment diagram you should also show the additional networking hardware for the production environment.

- Detailed hardware configuration

  In the instance Deployment diagram you should specify the hardware constraints (number of CPUs, clock speed, memory (RAM) and storage (disks) requirements) for each piece of hardware.

- Detailed software configuration

  In the instance Deployment diagram you should specify the major software components, such as JAR files, application files, scripts, and other data files.

Figure D-2 shows the descriptive Deployment diagram for the Hotel Reservation System.

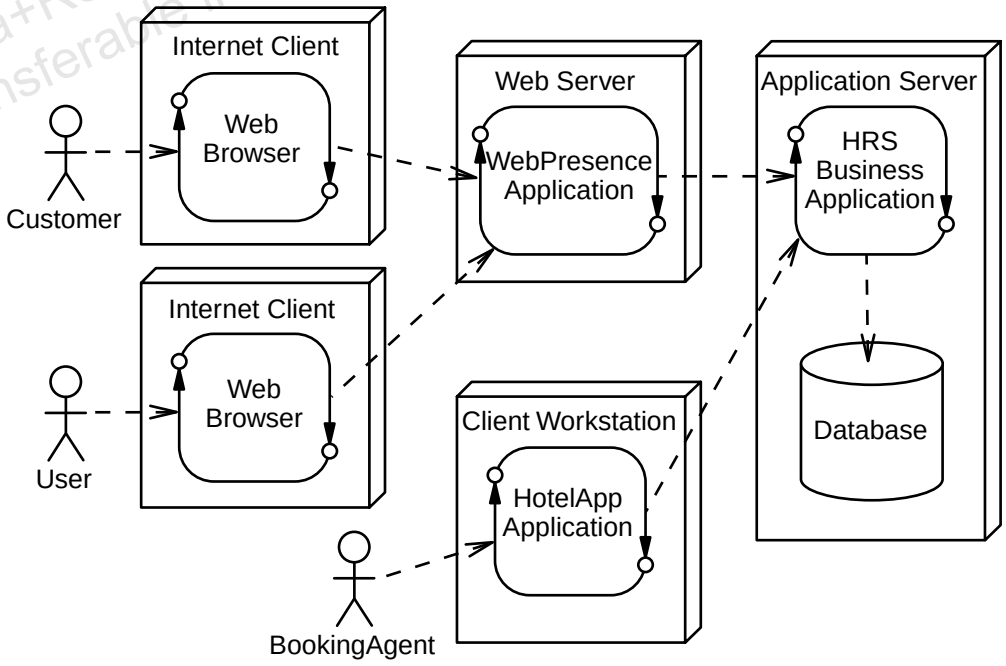

Figure D-2 Hotel Reservation Descriptive Deployment Diagram

The following subsections describe the instance Deployment diagrams for each major hardware component.

Kiosk Application Instance Deployment Diagram

The KioskApp is an internal web server. These are a few requirements on the kiosk machines:

- One kiosk in the lobby of each property.
- The kiosk does not include a connection to the Internet.

The KioskApp is very basic and does not require expensive hardware, except that the monitor must be a touch-sensitive. Figure D-3 illustrates the hardware constraints and software allocations.

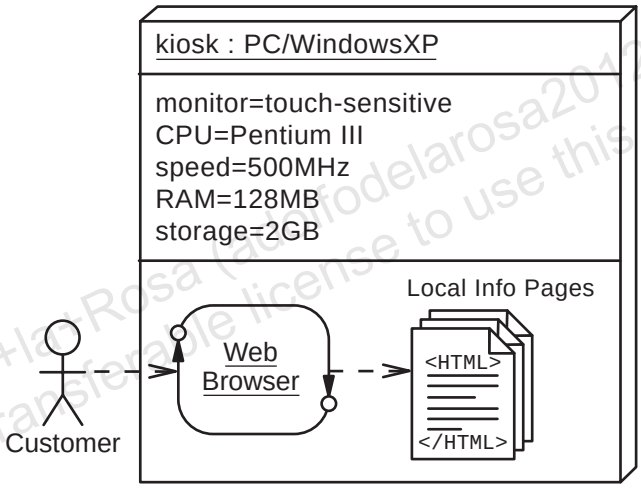

Figure D-3 Instance Deployment Diagram for the KioskApp

WebPresenceApp Instance Deployment Diagram

The WebPresenceApp is an external web application server. The Apache web server software will be used to server the static Web site pages and the Tomcat Web container will be used to server the dynamic elements of the Bay View system. The WebPresenceApp software requires the services interfaces and RMI stub classes, which are stored in the `services.jar` file. Figure D-4 illustrates this configuration.

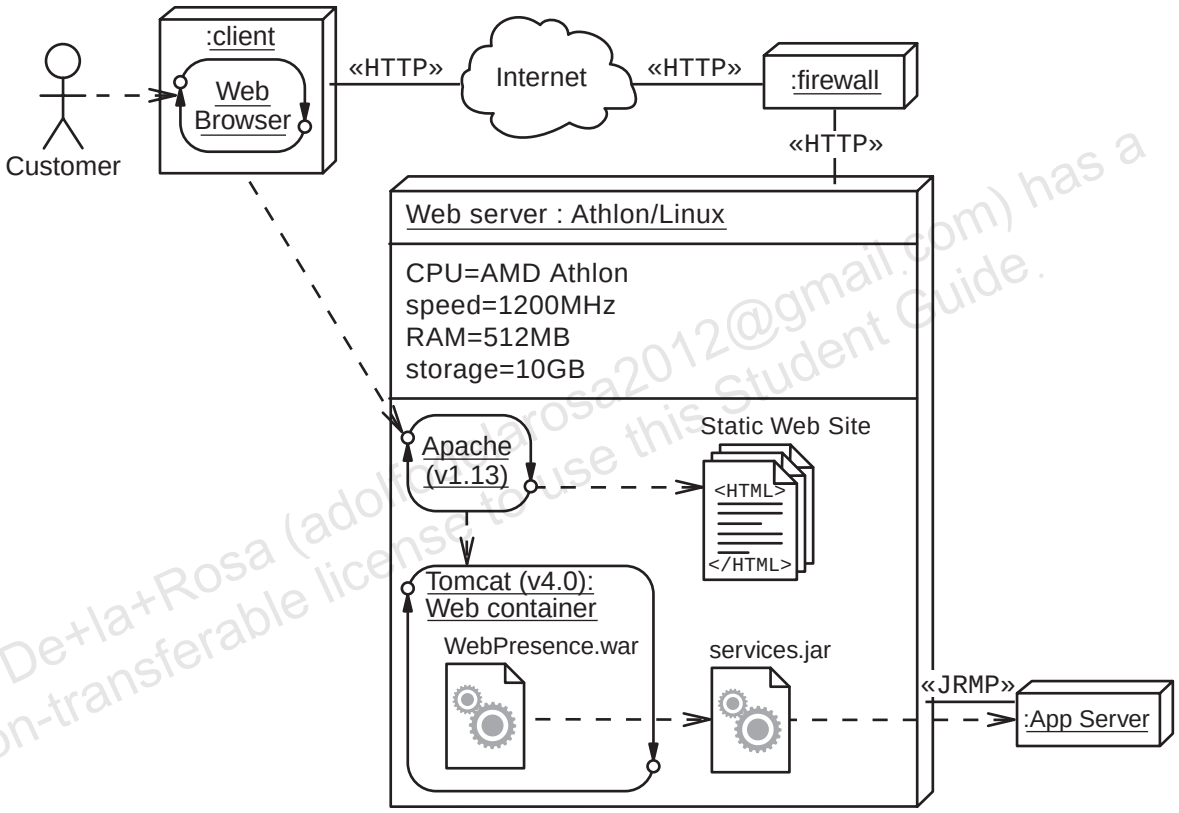

Figure D-4 The Instance Deployment Diagram for the WebPresenceApp

HotelApp Instance Deployment Diagram

The KioskApp is an internal application. These are a few requirements on the user workstations:

- HotelApp clients exist at each hotel location.
- HotelApp communicates to the server over a Virtual Private Network (VPN) using the RMI protocol (JRMP).

  Because the hotels are distributed over a great distance, these workstations must be connected to the Internet to communicate to the remote business tier (at the Santa Cruz facility). Because the system will be sending confidential data, the Deployment Specialist has decided to use VPN software to secure the communication path through the Internet.

Figure D-5 illustrates this configuration.

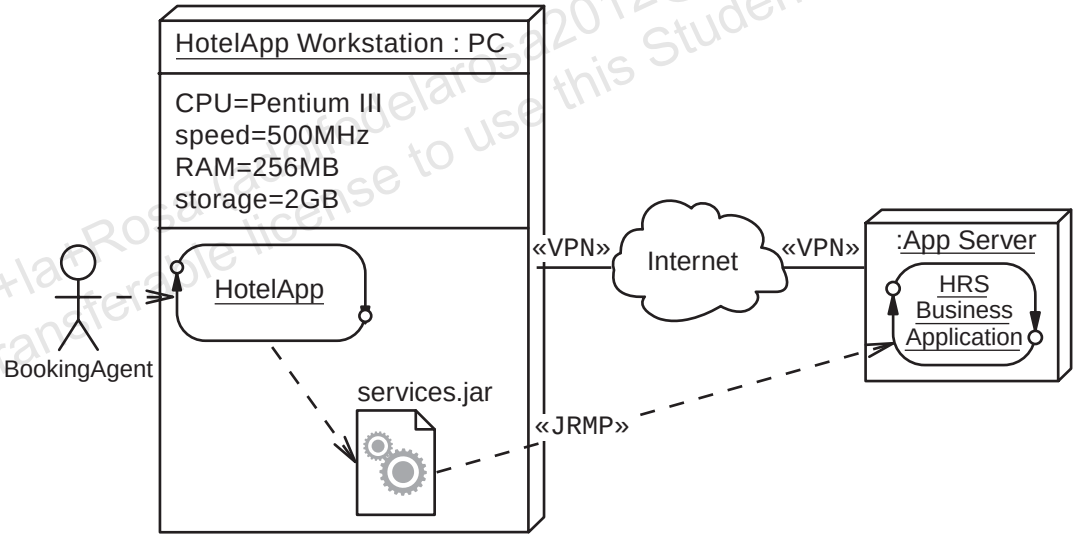

Figure D-5 The Instance Deployment Diagram for the HotelApp

Remote Application Server Instance Deployment Diagram

The remote application server runs the business services and the database management system. The HRS Business Application acts as the remote services implementation; therefore, this application binds the service objects to names in the RMI registry. This application requires the actual implementation classes of the services. The implementation is stored in the `services_impl.jar` file. This application also requires access to the PostgreSQL database software. Figure D-6 illustrates this configuration.

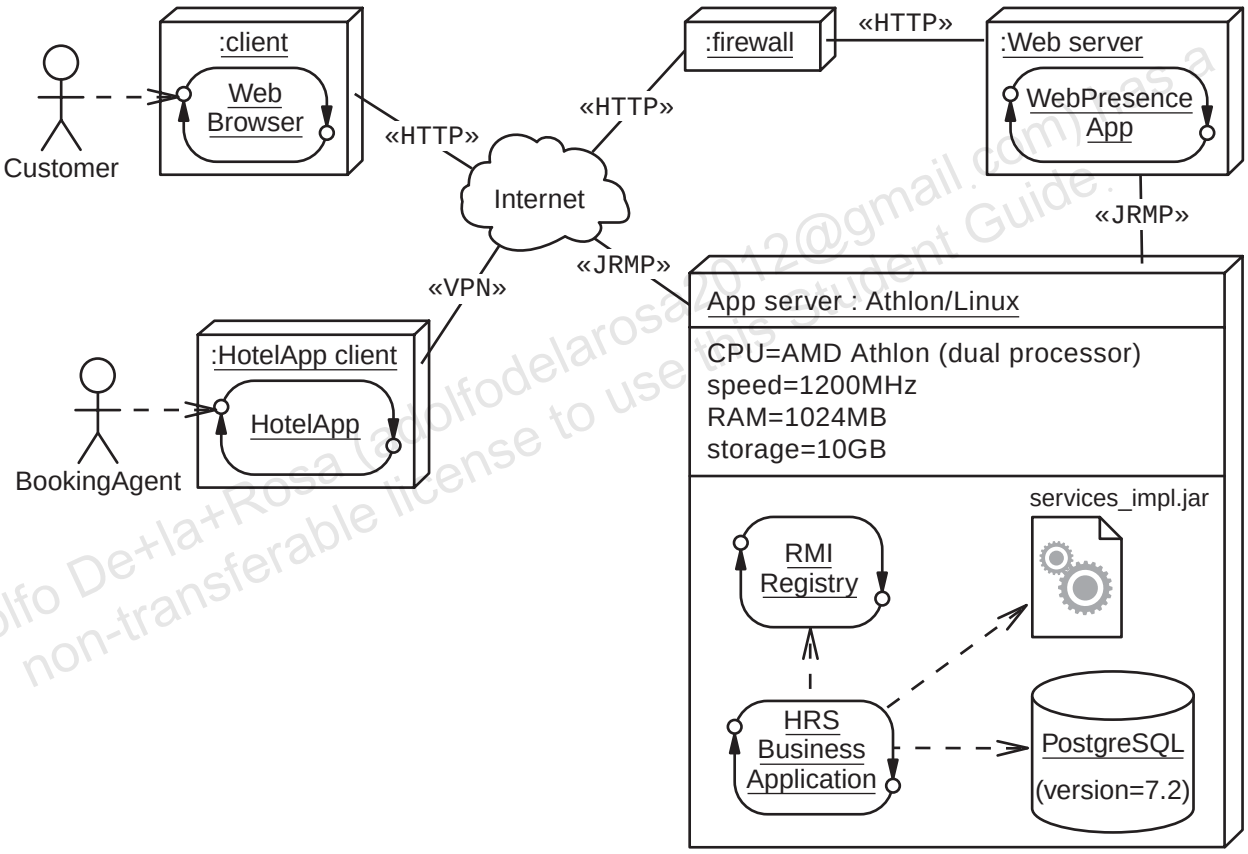

Figure D-6 Instance Deployment Diagram for the Business Tier Host

Summary

In this appendix, you were introduced to the Deployment workflow. Here are a few important concepts:

- The Deployment workflow takes the system implementation into production.
- The Deployment diagram specifies how the system is to be deployed in the production configuration.
- An instance Deployment diagram elaborates the descriptor Deployment diagram by clearly specifying the configuration of:
  - Networks
  - Machines
  - Software components