Chapter IX
Application Integration within the Enterprise Context

Ronda R. Henning
Harris Corporation, USA

ABSTRACT

The application software life cycle considers the functionality of a given collection of components within the context of a consumer's requirements definition. One set of requirements that are frequently overlooked are the requirements for application integration within the context of the enterprise environment. If an application creates vulnerabilities for other applications, is an administrative nightmare to maintain, or does not consider the security context required for execution; the application may not fulfill the intended requirements. This chapter addresses the question of the consideration of the enterprise information system's administrative and execution context as a component of the application software development process. The potential impact these considerations have on the acceptance of an application by the application's user community is presented, with illustrations of some representative problem areas for the reader's consideration.

INTRODUCTION

During the software application development life cycle the primary emphasis is on fulfilling the user's functional requirements. For example, if an enterprise specifies a transaction based service that is initiated in response to a user request, the software development team takes great care in defining the format and contents of a transaction, the initiation logic, and the criteria for completion. The primary emphasis is placed upon application functionality, user and process interaction, and system verification. These processes are used to measure the success or failure of the application development team in meeting the user's expectations.

Beyond the basic software development processes, however, there are implicit requirements that require integration of an application within the enterprise's operational infrastructure. An application that is deployed on an independent server, with unlimited bandwidth, and its own user interface is a relative rarity in the current enterprise computing environment.

To address the fact that various applications may share a processing infrastructure, the concept of a technical reference model has been developed. Technical reference models provide a blueprint for the coexistence of applications within a given enterprise. However, a technical reference model may not address all the integration issues that can be anticipated in during the integration of a new application with the existing infrastructure. An enterprise may support a specific user authentication process that relies upon digital certificates and directory services. If the development team does not have access to these capabilities, the application may not execute correctly when placed in the enterprise infrastructure.

BACKGROUND

The focus of the application development life cycle is the satisfaction of the functional requirements that must be fulfilled to address the user's vision. Applications are developed in response to perceived user needs, or requirements. They collect, process, and present information from one or more sources to the user in a specified format. The application is tasked with collecting the information, providing analytical or presentation functions, and interacting with either the user or other applications as required. The development process ensures that the application addresses the requirements as specified, and that these requirements are traceable and correctly allocated to various component elements of the application.

Few applications exist as independent entities that do not integrate with the other applications within an enterprise. Provisioning is defined as "a preparatory step in anticipation of some need" (Microsoft, 2003). In the case of an enterprise application, this preparatory step is the integration of the application with the enterprise's information technology infrastructure. This infrastructure includes the network, servers, and other software applications. The enterprise user or subscriber must also be made known to the application. For example, Xuan Shi (2006) defines a provisioning context in which the application is responsible for providing:

- A valid user name and authentication mechanism
- The data source used in the application
- Using the combination to find the data requested.

An application can only perform as expected if it has an ability to communicate within the enterprise to find its data, determine the authenticity of its users, and process the information requests. If an application cannot access and exercise the enterprise infrastructure, then it cannot function successfully within the enterprise's information processing context.

ADDRESSING THE ENTERPRISE CONTEXT

The provisioning process facilitates enterprise application integration and verification activities. Provisioning should be the final step in the application integration planning process. Well executed, the enterprise infrastructure will be transparent to an application's users. Poorly accomplished, enterprise context integration activities can result in a cost and schedule nightmare. If additional network capacity, hardware, or software licenses are required, these must be acquired through the