Example

This example shows an administration model for the rental-tracking system.

Monitoring and Control

The monitoring and control facilities are as follows.

- *Server Message Logging*: All server components will write information, warning, and error messages to the Windows Event Log of the machine they are running on.
- *Client Message Logging*: The client software will log messages if an unexpected error is encountered. The log will be written to the hard disk of the client machine for later manual retrieval.
- *Startup and Shutdown*: No system-specific startup and shutdown facilities will be provided because the software will run in the context of the IIS and SQL Server servers, and their facilities are considered to be sufficient.

Operational Procedures

Routine operational procedures are as follows.

- *Backup*: Operational data in the SQL Server database will need to be backed up. This will involve backing up the transaction logs every 15 minutes and backing up the application’s databases every day. Details of this procedure will be left to the database administrators.
- *Pruning of Summary Information*: The Reporting Engine does not remove the summary reporting information it creates. This information is left in place and is available to users of the Windows client interface. Database administrators will need to monitor the performance of the Reporting Engine and the management reporting aspects of the Windows client components and manually prune the summary information when its volume starts to impact performance. A written procedure will be supplied to explain how the pruning should be performed.

Error Conditions

The error conditions that administrators should be expected to handle are as follows.

- *Database Out of Log Space*: If transaction volume rises above a certain point, it is possible that the transaction log will fill. This will cause the system to suspend operation. Database administrators will need to recognize log space problems and manually back up the logs to free space. If this happens routinely, the backup interval for the transaction logs should be reduced.
- *Database Out of Data Space*: If the database runs out of data space, the system will stop operating. Again, database administrators will need to recognize this condition and either prune the summary information (see above) or add more data space to the system. A written estimate of the amount of space required for various volumes of workload will be provided.
- *IIS Failure*: If the IIS server fails, the system will completely fail, and Windows clients will lose contact with the server. Administrators need to recognize this condition and restart IIS. The system will recover automatically once IIS is restarted. The Windows clients will automatically reconnect once the server is available again.

Performance Monitoring

No application-specific performance monitoring facilities are planned. System performance monitoring should be achieved by using the following facilities.

- *SQL Server Counters*: The SQL Server 2008 product allows a wide range of performance counters to be collected and viewed via the Windows Server 2008's Reliability & Performance Monitor and the SSMS Activity Monitor. These performance metrics should be used to assess the volume of workload on the database and the time taken for the application's transactions to complete.

- *IIS/ASP.NET Counters*: IIS Server and ASP.NET produce a wide range of performance counters to be collected via the Windows Server 2008's Reliability & Performance Monitor application. These counters should be used to assess the number of Web requests being serviced and how long it is taking to service them.
- *.NET Counters*: The .NET runtime allows a wide range of performance counters to be collected via the Windows Server 2008's Reliability & Performance Monitor application. These counters should be used to establish the amount of non-Web-request workload that the application is performing and how long it is taking to perform the operations.

Support Models

Once your system is running in production, at least some of the system's stakeholders are likely to need help using or operating it, and other parties will need to provide assistance to them. The support model should provide a clear abstraction of the support that will be provided, who will provide the support, and how problems can be escalated between parties when searching for a resolution. This means defining the following in your support model.

- *Groups needing support*: The model must clearly define the groups of stakeholders who will require support, the nature of the support they need, and the appropriate mechanisms for delivering that support.
- *Classes of incidents*: The model must also define what sorts of support incidents are likely to be encountered and what sort of response is reasonable to expect in each case. The definition of each class of incident should clearly state the characteristics of an incident in that class, typically in terms of operational, organizational, or financial impacts.
- *Support providers and responsibilities*: Each type of support incident needs to be handled by at least one support provider, who must accept responsibility for resolving the incident. The model should capture who the support providers are and their responsibilities for incident resolution.
- *Escalation process*: A serious incident often requires a number of different support providers to resolve the situation because it is too complex or specialized for a single provider to handle. Your model should define how incidents are escalated between support providers and the responsibilities of each when this happens. This will help ensure that incident resolution does not stall because of confusion over responsibilities or a lack of expertise by a particular provider.