

12-1-2005

Enterprise Systems Manager

Edward Aractingi

Marshall University, aractingi1@marshall.edu

Follow this and additional works at: http://mds.marshall.edu/it_research



Part of the [Databases and Information Systems Commons](#), and the [OS and Networks Commons](#)

Recommended Citation

Aractingi, Edward. "Enterprise Systems Manager." Marshall University. Huntington, WV. Dec. 2005. Address.

This Presentation is brought to you for free and open access by the Information Technology at Marshall Digital Scholar. It has been accepted for inclusion in IT Research by an authorized administrator of Marshall Digital Scholar. For more information, please contact zhangj@marshall.edu.

Enterprise Systems Manager

By
Edward Aractingi

Agenda

- ◆ Introduction
- ◆ Objectives
- ◆ Requirements
- ◆ Technologies
- ◆ Implementation
- ◆ The Future of the system

Introduction

The need of remote SysAdmin

- ◆ Run commands on remote systems.
- ◆ Troubleshoot networked computers.
- ◆ Monitoring events
- ◆ Centralized Logging and Auditing
- ◆ Patch & Change Management.

History of remote management

- ◆ Consoles: Telnet – rlogin – rsh – SSH
- ◆ Graphical: Terminal Services (Windows 2000)
- ◆ Remote Desktop (Windows XP-2003)
- ◆ Management Tools:
 - Microsoft Systems Management Server
 - Dell OpenManage
 - IBM Director
 - HP OpenView

Limitation of current systems

- ◆ Need for client agent
- ◆ Manage proprietary products
- ◆ Steep learning curve
- ◆ Performance Issues

Users Profile

- ◆ Customers: Medium/ large enterprises
- ◆ Users: Systems Administrators/ IT Managers
- ◆ Multiple Hardware Vendors
- ◆ Multiple Operating Systems

Objectives

- ◆ Reduce system administration operating time
- ◆ Increase efficiency.
- ◆ Increase SysAdmins' response time
- ◆ Increase end users satisfaction rate.
- ◆ Increase quality of service
- ◆ Store hardware and change history

Major Features

- ◆ Environment Discovery
- ◆ Status Polling
- ◆ Inventory Collection
- ◆ Monitoring & Event Management
- ◆ Control and Management

Functional Requirements

Environment Discovery

- ◆ Scan the network
- ◆ Scan Active Directory
- ◆ Schedule discovery (date/time, interval)
- ◆ Save personalized settings
- ◆ Discovery runs as Windows Service.

Status Polling

- ◆ Poll systems using ICMP ping
- ◆ Store history of availability
- ◆ Respond to events by sending alerts
- ◆ Status poll on interval

Assets management

- ◆ Collect software inventory
- ◆ Collect hardware inventory
- ◆ Software inventory history
- ◆ Hardware inventory history
- ◆ Pre-defined views
- ◆ Reports

Monitoring & Event Management

- ◆ Monitor Windows events.
- ◆ Catch SNMP traps (Passive Monitor)
- ◆ Define actions
- ◆ Notify Users (email & SMS alerts)
- ◆ Group users
- ◆ Reporting

Management and control

- ◆ Shortcuts to remote desktop tool
- ◆ Computer management tool
- ◆ Shutdown / Restart / Logoff / Lock
- ◆ Troubleshoot networking
- ◆ Link to vendors management tools
- ◆ Run processes, Create shares, environmental variables, stop services.

Non-functional Requirements

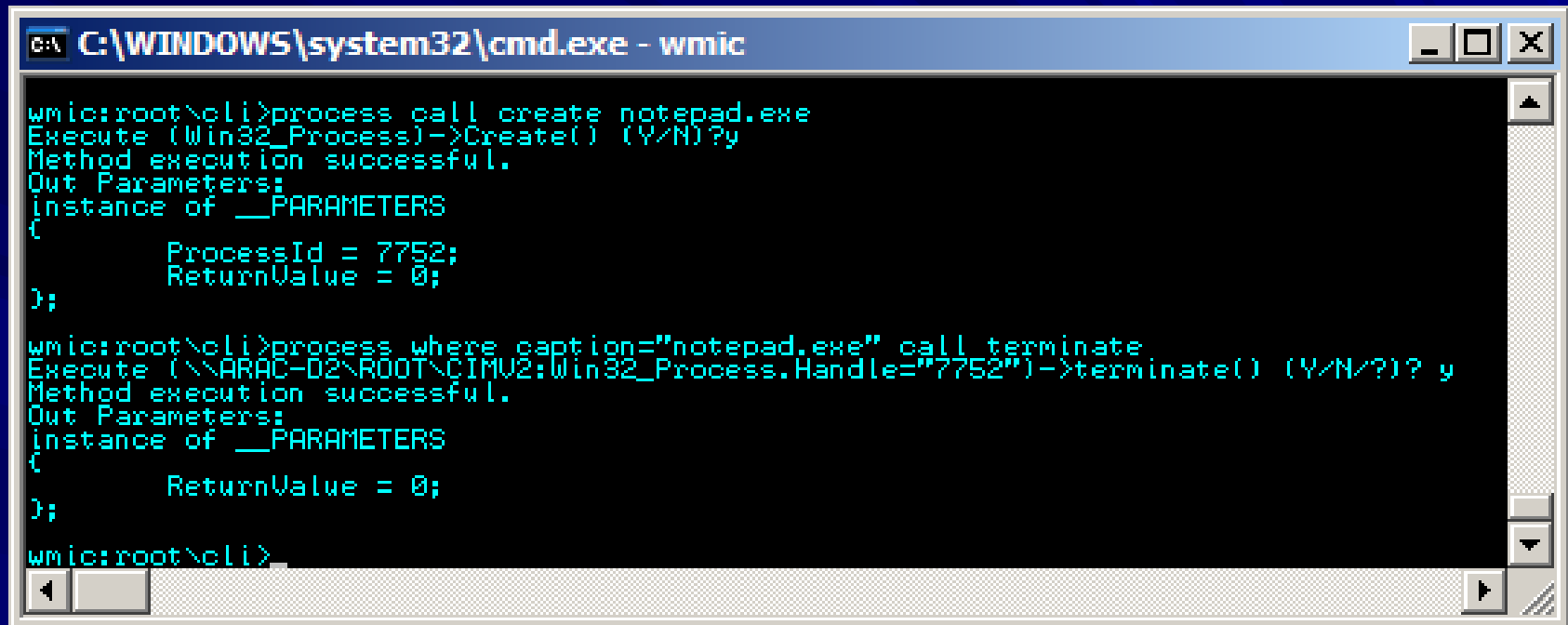
- Reliability
- Security
- Maintainability
- Binary Compatibility
- Portability
- Extensibility
- Usability
- Resource Utilization

Design & Implementation

Technologies

- **DMTF**
- **WBEM**
- **CIM**
- **SNMP**
- **WMI (wbemtest demo)**
 - **Classes**
 - **Methods**
 - **Properties**

WMIC



```
C:\WINDOWS\system32\cmd.exe - wmic

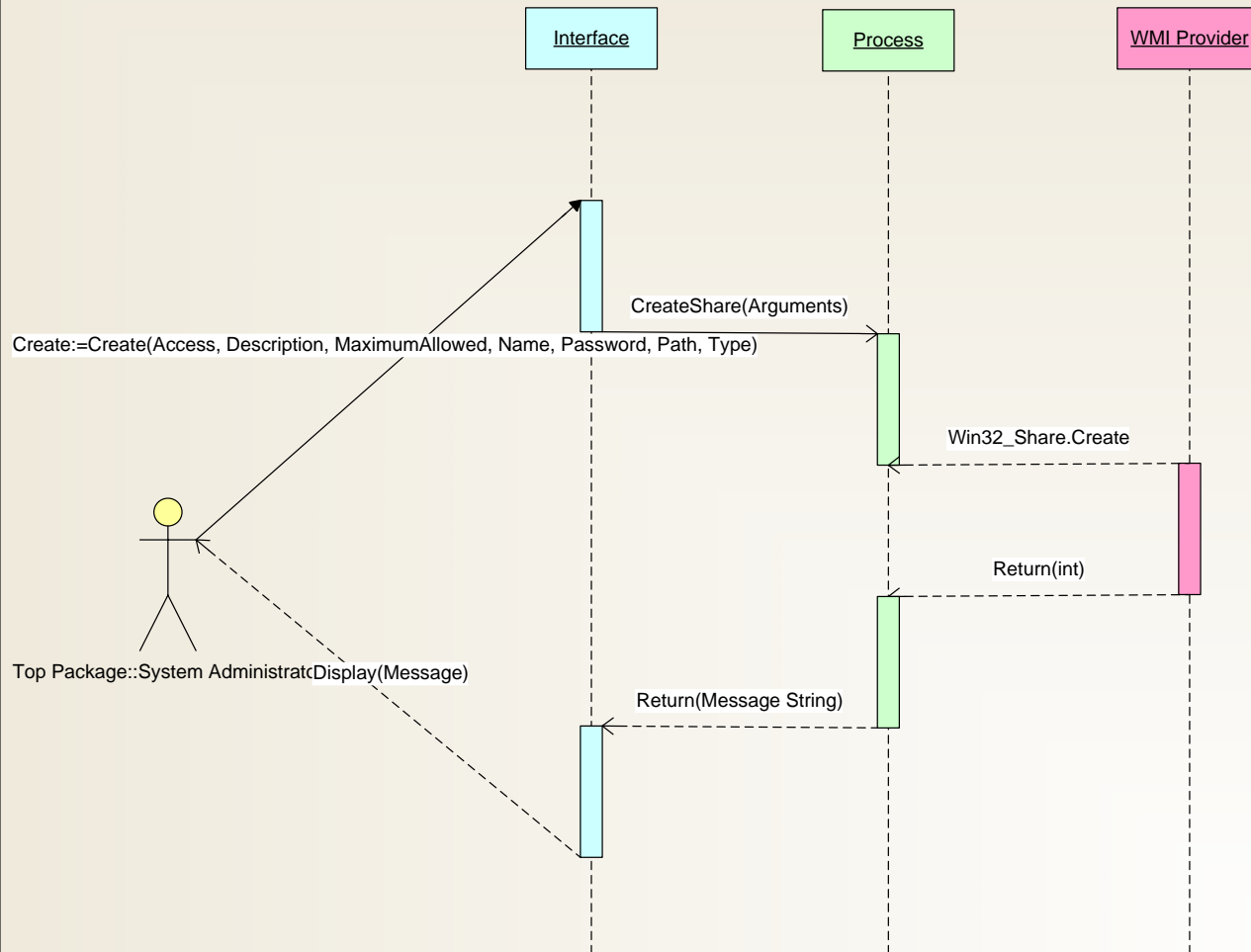
wmic:root\cli>process call create notepad.exe
Execute (Win32_Process)->Create() (Y/N)?y
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ProcessId = 7752;
    ReturnValue = 0;
};

wmic:root\cli>process where caption="notepad.exe" call terminate
Execute (\\ARAC-02\ROOT\cimv2:Win32_Process.Handle="7752")->terminate() (Y/N/?)? y
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ReturnValue = 0;
};

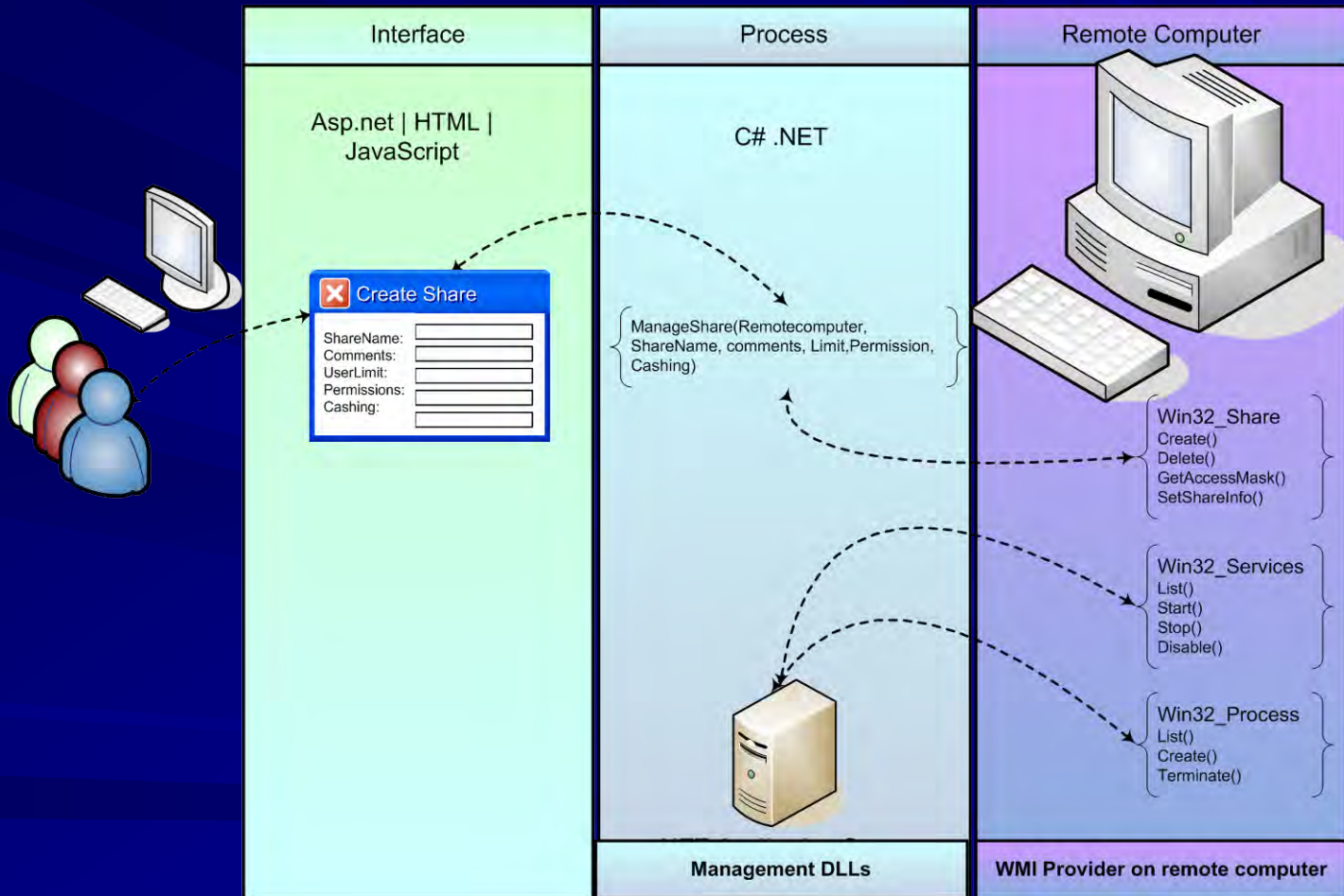
wmic:root\cli>
```

System Architecture

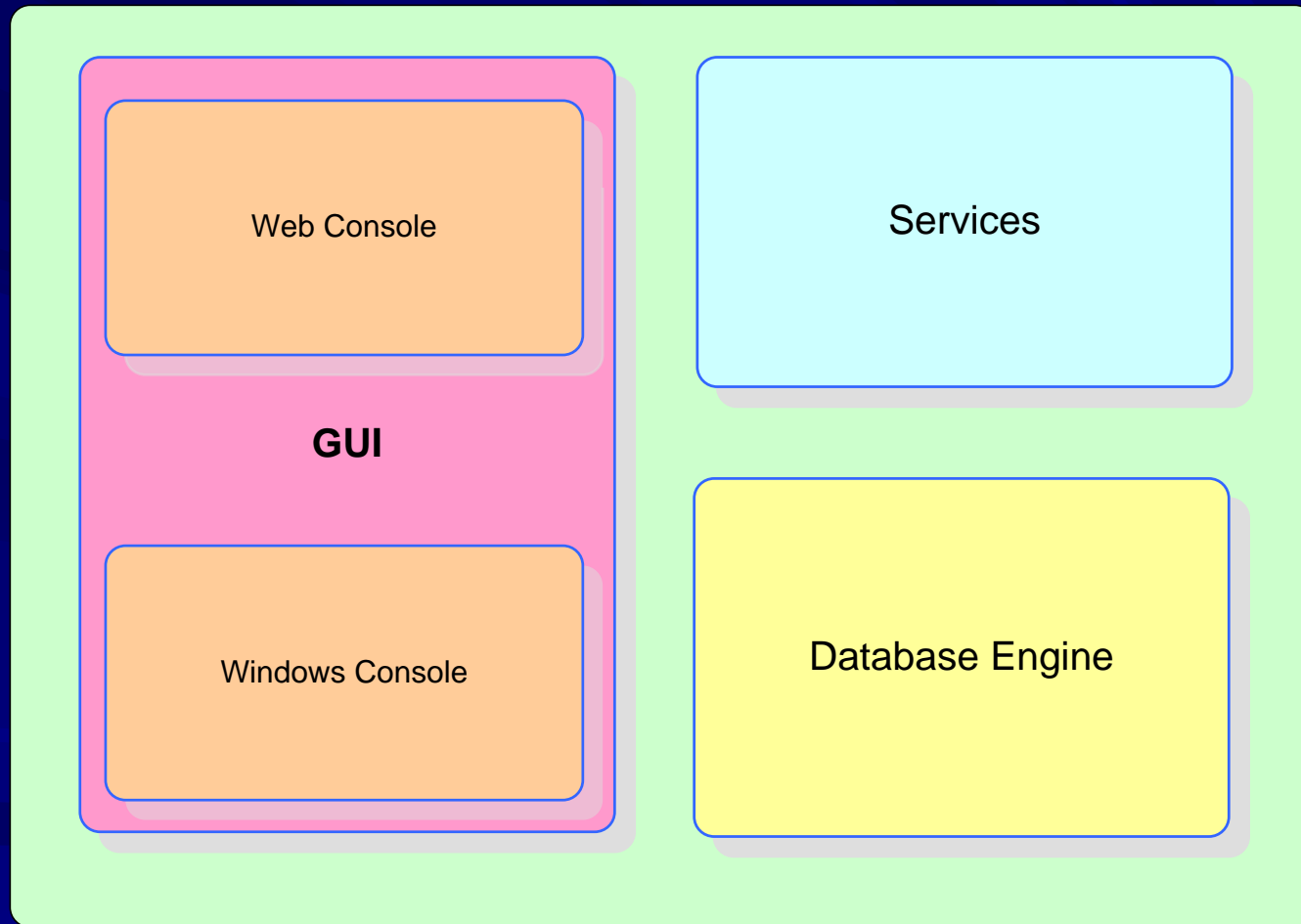
Sequence Diagram representing interaction between classes from different subsystems



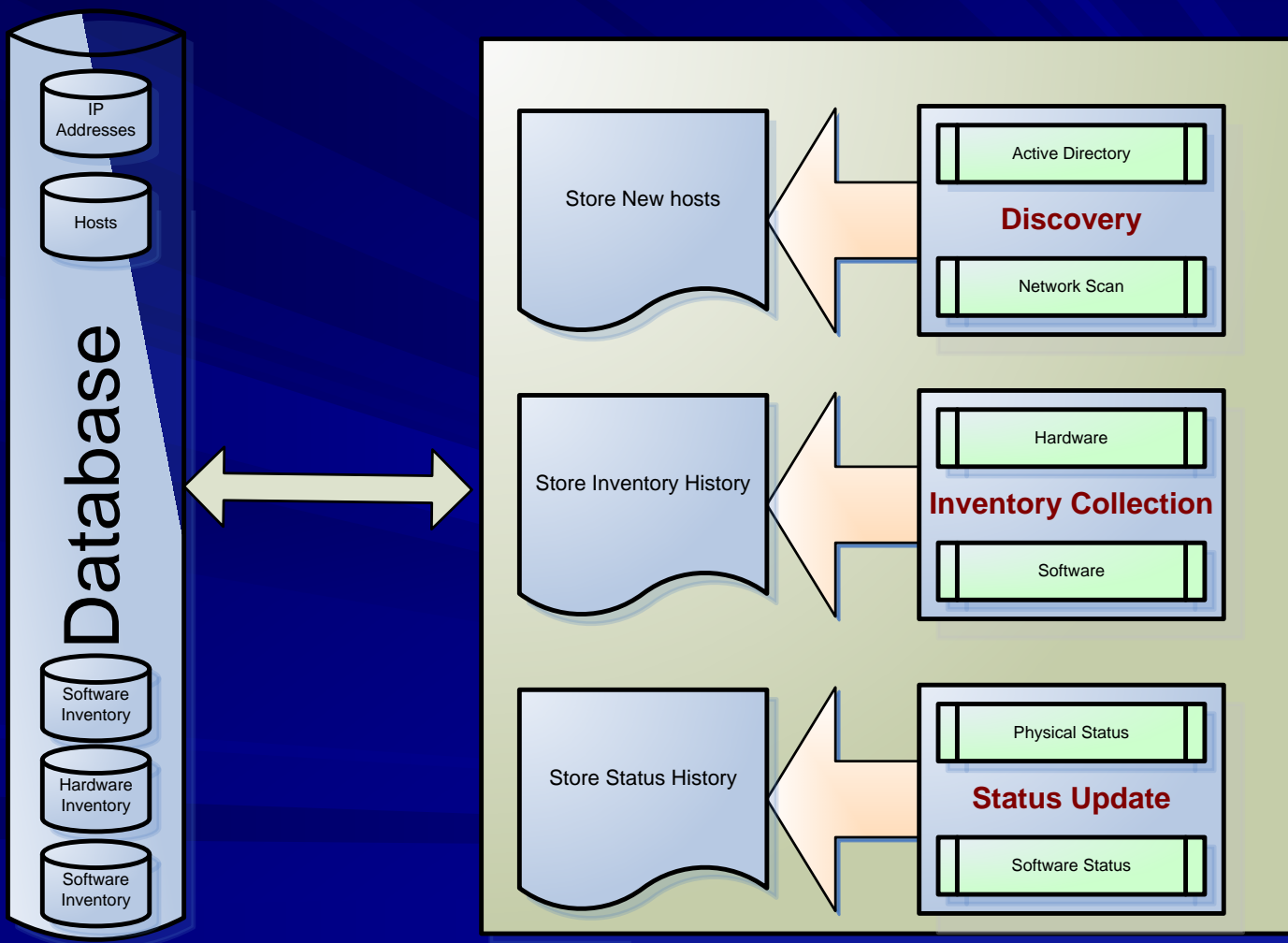
Layers Interaction



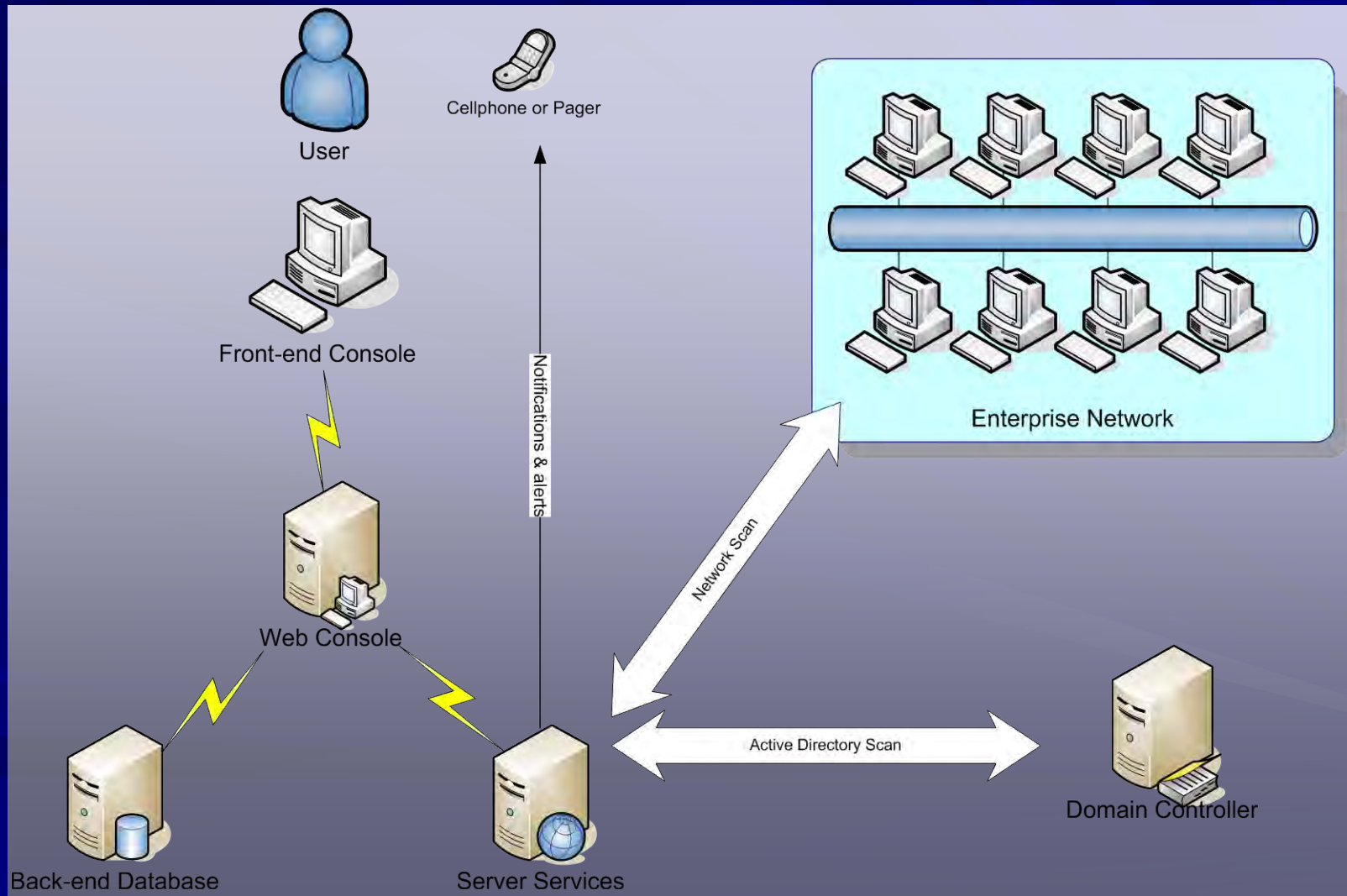
System Main Components



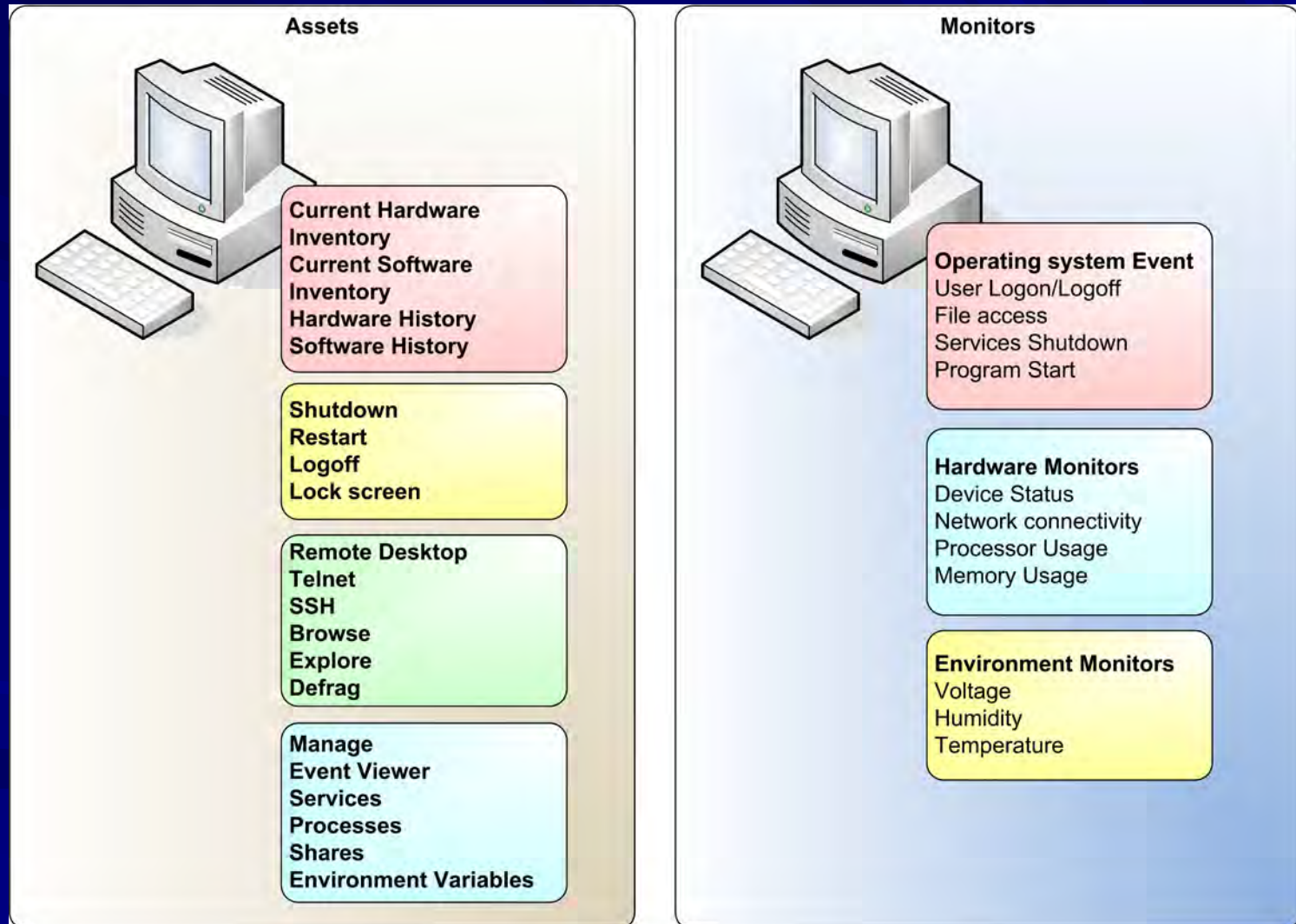
System Architecture



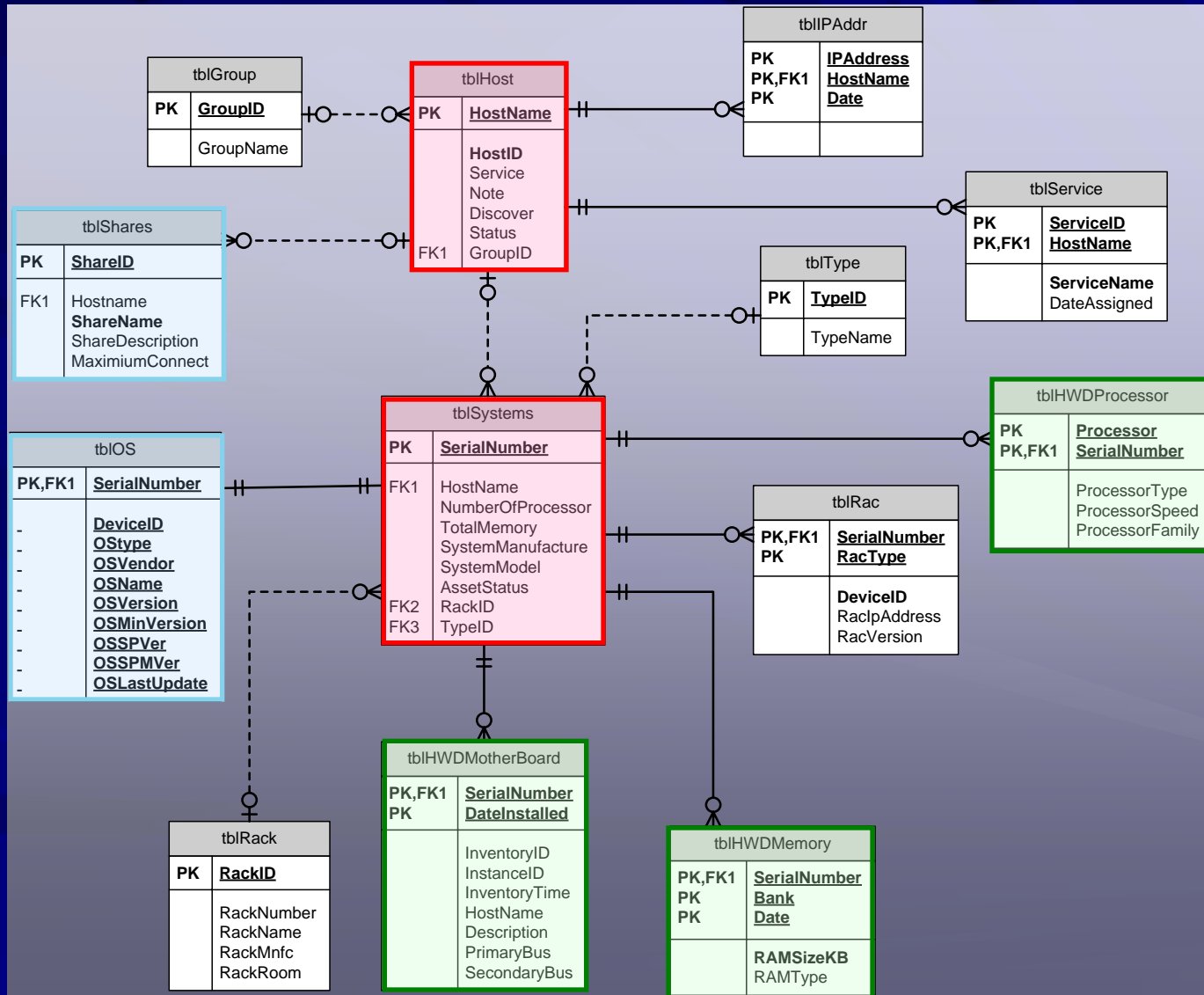
System Distribution



Major Functions

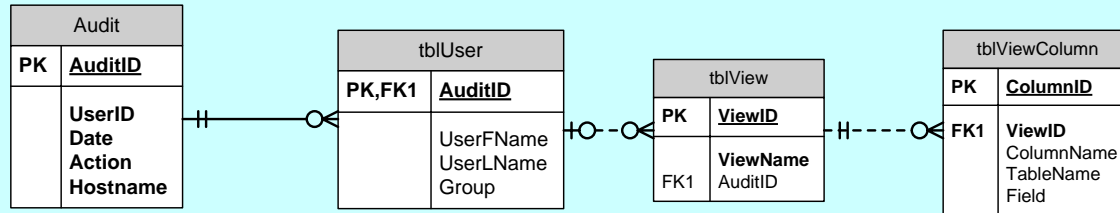


Entity Relationship Diagram 1

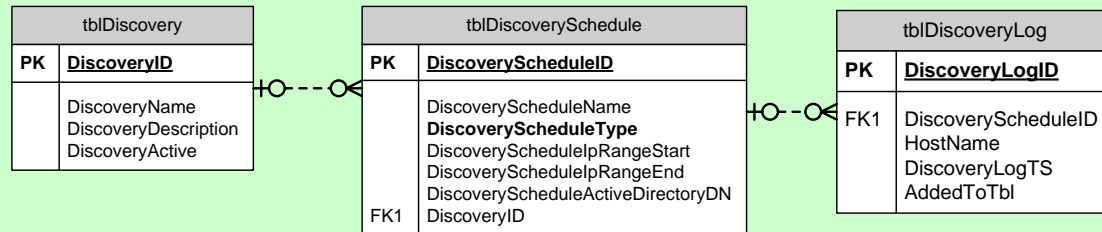


ERD (Continued)

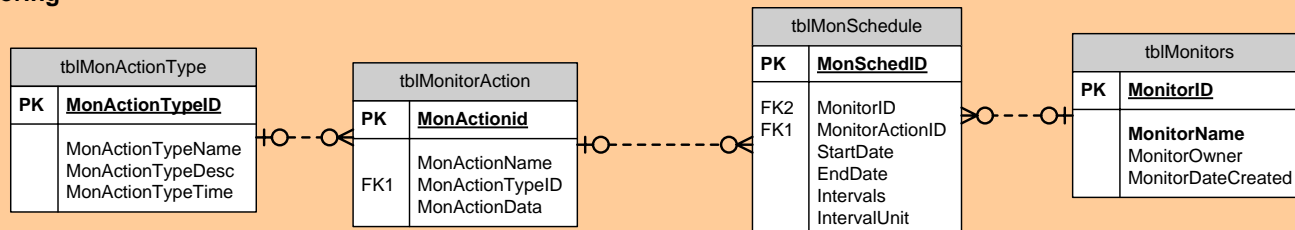
User Module



Discovery



Monitoring



Design Issue: Security

- ◆ Impersonate the current user
- ◆ Save alternative credentials for later use
- ◆ Encrypt communication with RPC-DCOM

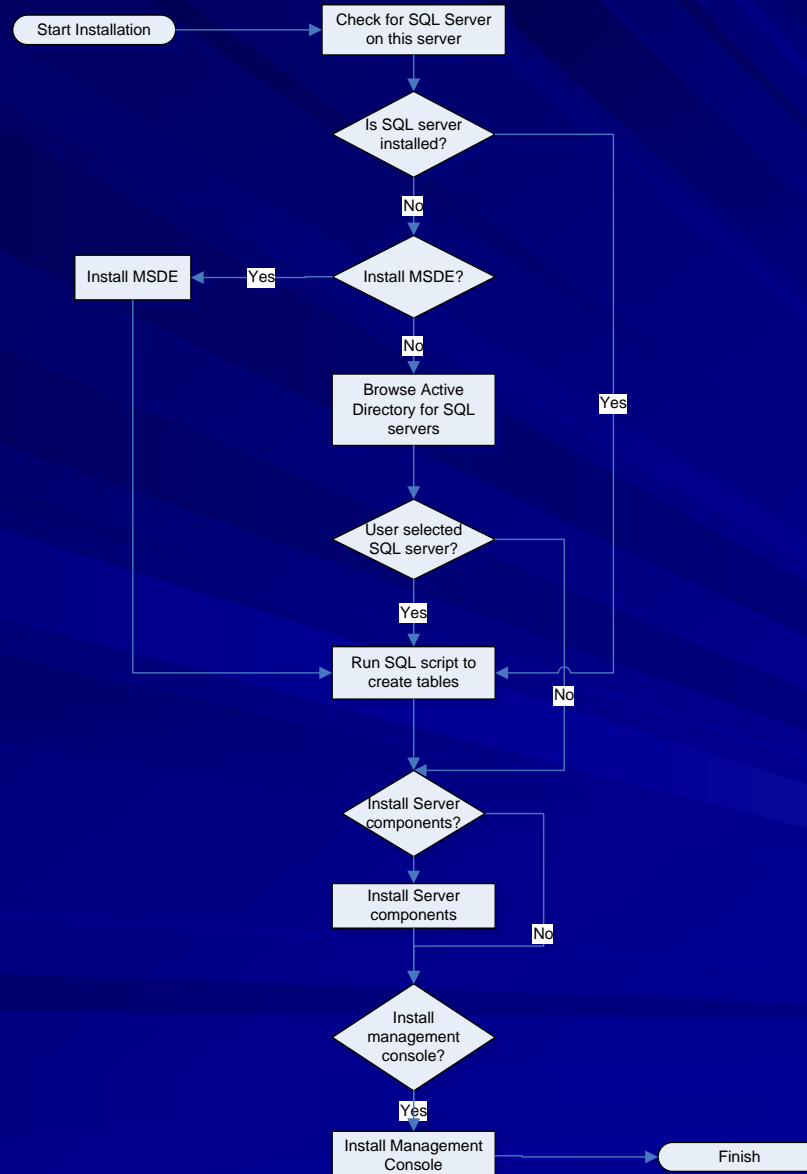
Upgrade

- ◆ Connect through HTTP to the software website.
- ◆ Compare the release numbers
- ◆ Check for .NET framework updates
- ◆ Install components
- ◆ Restart the system

User Interface

- ◆ Intuitive – XP style
- ◆ Saved Views
- ◆ Docked lists
- ◆ Documents Container
- ◆ Floatable controls
- ◆ Outlook like navigation.
- ◆ Context Menus and Shortcuts

Installation Flowchart



Prototype Demo

The Future of ESM

- ◆ Support for Linux (Client & Server)
- ◆ Support Virtualization
- ◆ OS deployment
- ◆ Blade Servers
- ◆ Manage networking devices
- ◆ API

Questions