Microsoft System Management

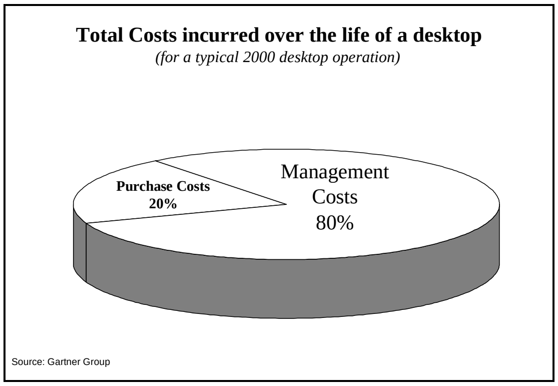
Windows “Chicago”: Systems Management Architecture

Overview

Managing a networked PC is difficult and costly today. In fact, management of the PC is the largest cost of deploying a PC in an organization. As the number of networked PC's in organizations grows, the importance of effective system management solutions increases. The components of a complete system management solution include systems (both servers and clients) with an effective management infrastructure, and applications which use that infrastructure to manage different types of systems located throughout an organization. The role of the operating system is to provide an effective infrastructure for system management, while protecting the investment customers have made in existing management schemes. This paper describes the system management infrastructure and features being provided in future versions of Windows, beginning with the release of Windows "Chicago".

The System Management Challenge

Ever since corporations began down-sizing from mainframes to LAN-based desktop systems, network managers have faced a daunting task of managing a burgeoning base of computing assets spread across the enterprise. When LAN’s were small and specialized, managing them was both less difficult and less important. But as organizations move mission-critical production, decision support, accounting, and other applications to LAN-based desktops, the importance of managing them cost-effectively has increased dramatically. A recent study revealed that today Information Services (IS) departments spend up to 80% of the life-time costs of a PC on activities relating to managing the PC.[1]



Total Costs incurred over the life of a desktop

(for a typical 2000 desktop operation)

Source: Gartner Group

To reduce these costs, IS departments require effective tools for managing the systems in their computing infrastructure. On every system, there are essentially three types of resources to be managed: the hardware (motherboard, add-in cards, hard drive, monitor, mouse, keyboard); the operating system software (drivers, system services, user interface components) and application software. Managing the hardware involves installation, configuration and inventory monitoring. Managing the operating system software encompasses a variety of tasks including system software distribution, system and user configuration management, and data backup. Application software must be installed, and it’s use must be licensed and metered. Finally, systems must be maintained over time, which includes tasks such as monitoring desktop performance and troubleshooting end-user problems.

[1] Costs include both software (OS, applications) and hardware costs

A Complete Solution to System Management

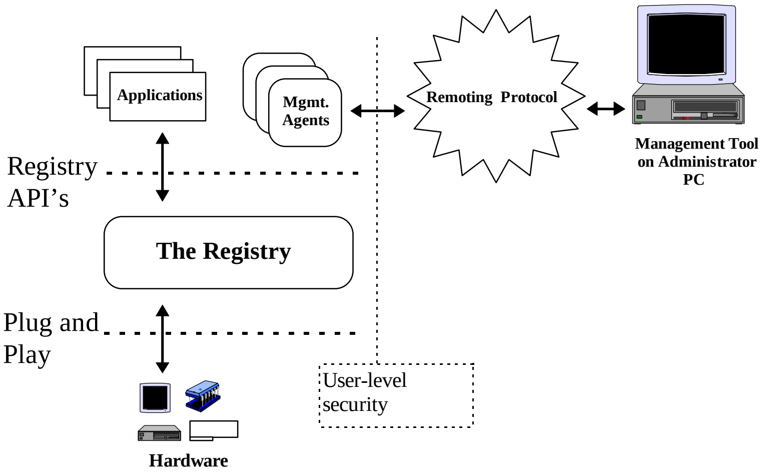
A complete solution to managing all the systems in an enterprise requires a combination of operating systems with the right management infrastructure and management applications which leverage that infrastructure to manage different types of systems located throughout the enterprise.

Future versions of Windows operating systems will provide a comprehensive infrastructure for system management. This infrastructure will collect information about the hardware, operating system, and applications and make that information available to system management applications. Access to management information will be provided through standard, documented API's (Win32) and support for transmitting that information over the network will be provided through standard remoting protocols (RPC and SNMP). These services will be based on Microsoft's Windows Open Service Architecture (WOSA) for networking to make them available over networking products from multiple vendors. This approach will protect the investment customers already have in system management products such as Novell NMS, HP Openview, Sun Net Manager, Intel LanDesk, or IBM LAN NetView, while at the same time facilitating the development of more full-featured management applications from any vendor.

Microsoft currently has under development a system management application called Microsoft System Management Server (also known by the code name "Hermes"). System Management Server will provide features to ease management of multiple types of systems across the enterprise, including MS-DOS, Windows 3.1, Windows for Workgroups, Windows NT or NT Advanced Server systems, and systems from other vendors. These features include software distribution and installation, network application management, inventory management, remote control troubleshooting, and performance tuning. System Management Server is a Win32 application which runs on Windows NT Advanced Server and records data about the systems it manages in a SQL Server database. It provides the required management agents for systems which lack a management infrastructure, and will use the infrastructure built into future versions of Windows to deliver a complete cross-platform solution.

Windows “Chicago” System Management Architecture

The fundamental components of any system management architecture include a store of data about the resources to be managed, open interfaces for entering and retrieving data from that store, and a protocol for communicating that data over the network to a system management application. In addition, the components on the system must be protected from unauthorized access by a security system. In Windows “Chicago” these components are provided by the Windows Registry and it’s Application Programming Interface, the Plug and Play system, and industry-standard Remote Procedure Call protocol. Windows “Chicago” will also provide user-level security to control access to these components.



Windows “Chicago” desktop Management Architecture

The Windows Registry

The Windows operating system will consolidate configuration and status information for all hardware and software components into a single, structured database called the Windows Registry. Management applications can use Win32 API’s over the network via Remote Procedure Call (RPC) to access configuration and status information about all system components.

The Registry will contain both static and dynamic information. For hardware components, information will include manufacturer and device identification, resources allocated to each device, and device specific configuration. Applications will store configuration information, including manufacturer name, package name and current version number. Operating system settings and user configuration information will also be recorded in the Registry.

The Plug and Play System

Windows "Chicago" will be the first operating system product to support the Plug and Play architecture. The Plug and Play architecture enables automatic installation and dynamic reconfiguration of Plug and Play devices.

Plug and Play devices record their configuration information in INF files. The INF file contains a list of system resources required by the device, the drivers needed to operate it, and device-specific configuration and status information. When the device is installed, the Plug and Play system allocates system resources (DMA channels, IRQ's, base I/O addresses, etc.) to the device, loads the device driver, and writes the information about allocated system resources and the device specific information into the Windows Registry. Once installed, the device settings can be configured via the Registry API's.

The Plug and Play system also places information in the Registry about system resources allocated to non-Plug and Play devices. However, if vendors of legacy devices want to report management information beyond that exposed by the Plug and Play system, they will need to modify their drivers to do so.

Remote Procedure Call (RPC)

Microsoft Windows operating system products from Windows "Chicago" forward will include support for applications developed using the Remote Procedure Call (RPC) protocol. RPC allows applications running on one computer to execute functions, or processes, on another computer across the network regardless of the underlying network protocols in use or the type of software running on the remote computer.

Microsoft's implementation of RPC is compliant with the specifications of the Open Systems Foundation (OSF) Distributed Computing Environment (DCE), and works with other compliant products. This enables programs running on Microsoft Windows to access services running on other platforms such as HP® and IBM® AIX®. Similarly, administrative applications running on other platforms will be able to remotely control processes on Windows systems.

Because RPC will be included with Windows, and supports all popular networks, developers will be able to build advanced network management tools, such as software distribution systems, which perform tasks remotely that previously could only be done by physically visiting each system.

Securing Access to Managed Systems

To prevent disruptive or unauthorized access to systems that are being managed, a system management architecture must implement an effective security model. Beginning with Windows “Chicago”, Microsoft will provide user level security for desktop systems.

User-level security associates access privileges with individuals or groups. When a user attempts to access a shared resource, an associated list of access permissions is checked before access is permitted. Windows “Chicago” implements this model by using access permission data maintained on a Novell NetWare or Windows NT Advanced Server system. Each Windows “Chicago” machine running user-level security has a companion authenticating server. When the remote user attempts to access the shared resources on the Windows “Chicago” machine, it authenticates him by communicating with its authenticating server before permitting access.

This implementation has several advantages. First, the administrator needs to maintain only one database of users and groups. This database is then used by all Windows desktops to enforce security -- there is no need for the administrator to create lists of users and groups for each desktop. Second, account management is centralized under the network manager’s control. If a user leaves or joins a department, account changes need to be made in only one location. Finally, authentication of users is performed with a single password on secure central server.

The Windows “Chicago” security model provides security for more than peer file and print servers. Other Windows services, such as RPC and backup agents, also permit access to the desktop. Each of these services can use the security model to control access. The security system can even be used to restrict access to specific system capabilities, such as modifying or deleting program groups. And it has open, undocumented interfaces so it can be used by developers to prevent unauthorized access to their products.

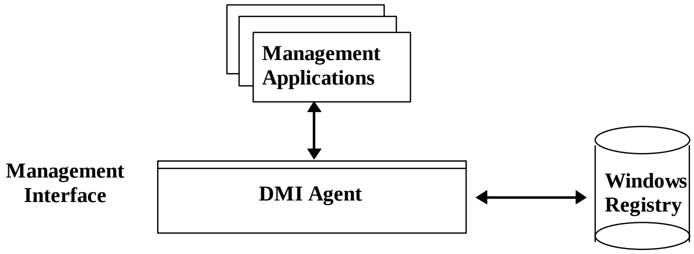
Desktop Management Interface (DMI) Support

The Desktop Management Interface (DMI) specification has been proposed by the The Desktop Management Task Force (DMTF) to provide a solution to some of the problems of system management. DMI defines components which perform several of the functions described above, including collection and storage of management information and defined interfaces for reporting management information for use by management applications.

DMI defines a Management Information File (MIF) which vendors provide with their hardware or software component. The MIF contains mandatory component identification information and a mechanism for optional data that is component specific. A local agent, the Service Layer, resides on each managed system and provides the services for registering components with a MIF database and providing access to that database through the Management Interface (MI). Management applications use the MI to query the MIF database, set attributes, or execute management processes.

Microsoft will implement support for DMI through the architecture defined above. Like the MIF in DMI, The Plug and Play INF format will provide information about hardware components to the Windows Registry. And Plug and Play performs device installation and configuration as well. Since IHV's and OEM's already develop drivers for specific operating systems, the incorporation of management information in the INF file format will make it easier for hardware vendors to expose information about their components.

For the developers of management applications, DMI provides a solution for operating system platforms that do not provide a management infrastructure, such as Windows 3.1. With future versions of Windows, management applications will be able to use the infrastructure provided in the operating system. The Windows Registry implements a data store that is functionally equivalent to the MIF database in DMI. Software components will provide information to the Windows Registry through the Registry API, which is analogous to the Management Interface in DMI. Windows goes beyond the DMI specification to provide RPC and user-level security for reporting information to remote management applications in a secure fashion.

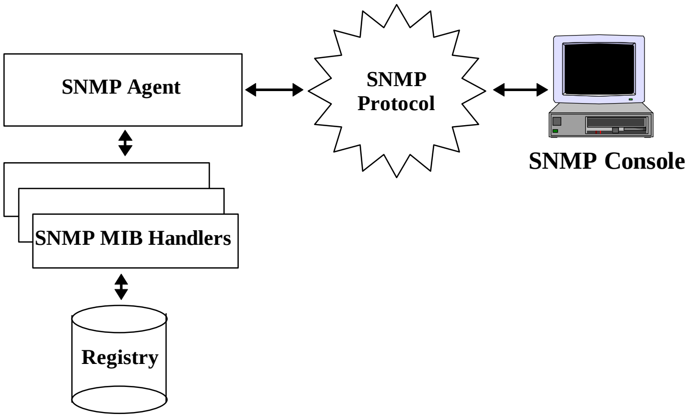


DMI support in Windows "Chicago"

Applications which have already written to the Management Interface will run on future versions through a DMI compatibility agent. When a DMI application requests information about a managed entity, the DMI agent will use the Registry API's to retrieve the requested information from the Windows Registry. Microsoft will make the DMI agent available soon after the release of Windows "Chicago".

Simple Network Management Protocol (SNMP) Support

Like RPC, SNMP is an open, standard protocol used to transport management information and commands between a management console and a managed entity. Several network management products use SNMP as their remoting protocol, including HP OpenView, Novell NMS and IBM NetView and SUN SunNet Manager. Although RPC is Microsoft's preferred remote communication protocol, Microsoft will also include support for SNMP-based applications in Windows "Chicago" and future versions of Windows.



SNMP support in Windows "Chicago"

With SNMP-based systems, information about a system is made available through a component called the Management Information Base (MIB). Each MIB consists of a list of object identifiers describing a managed entity. The MIB Handler retrieves object values from the managed entity and sends them to the SNMP console. In Windows, the SNMP Agent - MIB Handler interface is open so third parties can plug in their own handlers. In Windows "Chicago" Microsoft will provide handlers for industry standard MIBs, like the MIB-II, which describes information about the TCP/IP protocol.

In the long run, Microsoft plans to eliminate the need for MIB handlers in Windows by extending the definition of the **INF** file format to include information in MIB's today. The Windows SNMP agent will also be enhanced to automatically provide the information requested by the console if it is described in an **INF** file. With this approach, if a hardware vendor supplies an **INF** file his device will be automatically manageable through SNMP and the Registry API's.

“Chicago” Desktop Management Features

In addition to the systems management infrastructure described above, Windows “Chicago” will include tools to perform many essential system management tasks.

Hardware inventory monitoring

Few network administrators know the precise types and quantity of hardware deployed in their corporations.   A recent survey by Forrester Research found that over 50% of Fortune 1000 MIS managers feel inventory management “urgently needs a fix.” In future versions of Windows, all hardware configuration and status information will be stored in the Windows Registry. This information will be available locally via the Registry API’s and remotely over RPC. Vendors of management tools can then write applications to gather inventory information from Windows desktops across the organization.

Hardware installation and configuration

IS personnel spend considerable amount of time and money installing and configuring hardware.   Automatic installation and configuration of Plug and Play devices and Plug and Play operating systems will dramatically reduce this cost.

System software distribution

Windows “Chicago” will make it easier to roll-out Windows across an organization. Administrators will be able to place a copy of Windows “Chicago” on a server and have users run Windows setup over the network. The administrator can decide beforehand which components and capabilities should be installed on the desktop. Improved hardware detection technology will identify hardware and configure the system automatically to minimize the need for user intervention.   Windows “Chicago” can also be run from a network server to support diskless workstations.

User configuration management

In many companies employees share multiple PC's. Chicago will support user profiles to enable people to access their personal groups, applications, and data from any system on the network. This "multiple user mode" can also be provided on a stand-alone system. This capability is provided through the Windows Registry, which stores per-user configuration information separately from system information so that each of these can be managed separately. Per-user configuration information can include preference data such as favorite screen colors, mouse click speed, program groups, etc. These preferences will be centrally stored, accessed when the user logs into a Chicago system, and used to install the appropriate configuration so that user will be immediately productive working on a familiar environment tailored to whatever hardware they are using.

Desktop configuration control

Windows "Chicago" will provide administrators the ability to lock the system configuration and restrict access to the user interface to prevent even knowledgeable users from making changes. The information about locked and hidden configurations is saved on a central location. An administrator will be able to remotely remove the locks to permit users to modify this configuration.

This capability will enable administrators to prevent users from installing certain components, set usage policies centrally, and define a "safe," configuration for Windows to go back to in the event that a user inadvertently alters their desktop configuration.

Data back-up and restore

Windows "Chicago" will facilitate desktop data backup by shipping backup agents with the operating system. A backup agent is software installed on a client which allows the client to be backed up by a central backup system. Microsoft will include backup agents for Cheyenne ARCserve and Novell Storage Management System-based backup solutions.

Peer server administration

Administrators require the same degree of control over peer servers that they have over departmental server systems. In Windows "Chicago," administrators will be able to view the connections, shares, open files and audit log on a remote peer server. They will also be able to control connections and shares, and close opened files. An administrator can install a peer server without allowing the user to share files or printers, so the administrator can share resources or download driver upgrades and patches without user interaction.

Application distribution

In a client-server environment, important business applications must be installed on thousands of systems rather than a single mainfram. Future versions of Windows will contain several technologies that make it easier to distribute software.

Software distribution is a two-step process. First, application files need to be downloaded to the desktop. Windows “Chicago”’s backup agents or peer server can be used to move files from a central server to the desktop. Administrators can use server login scripts to instruct the client PC’s to “pull” down the application files and configure the application. The second step is to configure the desktop. This includes creating new program groups, icons or Chicago “links” for the new application and placing the application’s configuration information in the Registry.

Application vendors can use these capabilities to implementations software distribution products, and current software distribution products such as Microsoft “Hermes” product and Novell NetWare Navigator will work without modification in this architecture.

Application metering and licensing

Most software applications today are furnished with a license agreement that dictates the terms of how that software may be used. The agreement determines the policy of how the application should be used, either on a standalone PC or on a network. Because of a lack of central control over the software administrators have to resort to spot checks to make sure no licensing agreements are being violated.

Future versions of Windows will contain support for the Licensing Services API (**LSAPI)**. These API’s are an industry standard and founded by leading software vendors like Microsoft, Brightwork, Digital Equipment Corporation, Gradient Technologies and Novell. LSAPI is also part of the larger WOSA framework. Using them, applications can ask for licenses from back-end “license servers” independent of the vendor of the license server. Any third-party vendor of a license server can plug-in under these API’s to provide licensing support.

System performance analysis

Windows “Chicago” will include a tool which displays desktop performance information. The performance monitor will have an open interface so components will be able to expose their performance information. Administrators will also be able to use the performance monitor to measure remote systems.

Troubleshooting end user problems

Microsoft will provide a tool for network managers to view the Windows Registry on remote systems. Using this tool, support personnel will be able to view and modify desktop hardware and application configurations. This will help diagnose and resolve end-user problems.

If the operating system fails during initialization, Windows “Chicago” permits automatic reinitialization of the system into a safe configuration. Once the system started with “clean boot”, an administrator can use diagnostic tools to fix the problem.

Summary

Future versions of Windows, beginning with Windows “Chicago”, will include a comprehensive infrastructure to enable effective system management solutions. Developers of management applications will use the Windows system management infrastructure to build complete system management solutions. These applications, combined with management features built into Windows, will reduce the cost of deploying Windows systems in the enterprise.

Information in this document is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Microsoft Corporation.