Heterogeneous Console Management for the HP Enterprise

William Johnson
President, TECSys Development

Imagine -- it's a sunny day, you're on the beach, flying a kite, soaking up the sun, relaxing, knowing that your data center is being watched by your in-band management tool of choice. The management station is consistently polling remote agents, gathering data, analyzing it with rules, and issuing actions or alerts to the operations center.

Then it happens – the machine crashes, a router needs to be rebooted – agents don't work, machines are down and applications are hung. Operations calls their superhero – YOU -- to come in and fix the problem. You don't don a cape, fly through the sky, or anything like it. You're the superhero because only you know the technical magic incantation to restart the machine, get the network moving, and get applications going.

You reel in the kite, gather your things, dash for the car, and drive furiously to the data center to enter the cold, gloomy sound of fans running, fluorescent lights, and dead machines in the computer room. You scramble for the crash cart to find the cable, which will be used to bring life new life to the dead machine. It doesn't fit, another connector, another adapter and finally with a stroke of luck it works – commands are issued on the keyboard in record time, the machine begins to come alive, the network is un-hung, andanother day is saved by you… superhero.

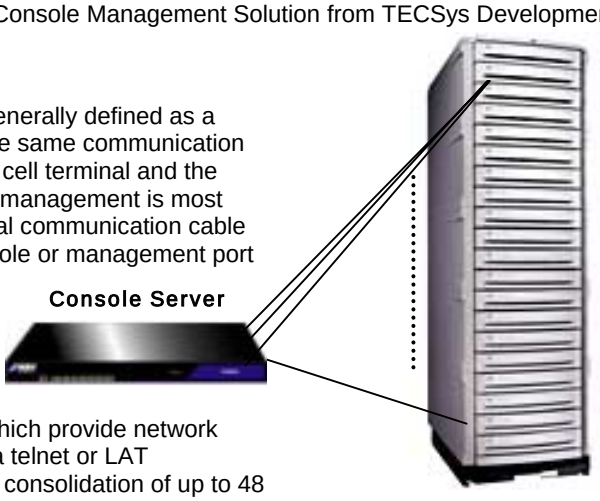
Sound familiar? Yep. The problem is you are still the weak link in the chain. The business still has to call you out of bed, away from vacation, or out of meetings, and your job is never done. Your knowledge is specific to you as are your training and capability and skills. If only you could leave a trace of what you did, how you did it, and what the responses were…and be able to access the critical console remotely, via dial-up, the web, or wireless for those critical situations. No longer would you be tied to the machine, instead you would be free, free to manage the system from anywhere.

In order to provide that ability you MUST be able to reach out to any console, whether to a server, storage controller, network router or switch, SAN Controller, or application management screen. To do this, SNMP or In-Band is out of the question. Only an Out-of-Band solution provides the ability to do such a mission critical task. Coupled with an In-Band solution like OpenView, you can have complete visibility of your enterprise whether it's up or down, on or off, running or not.

Simply put, both In-Band and Out-of-Band or Console Management are required to manage the whole enterprise and they must be integrated like OpenView and ConsoleWorks from TECSys Development. This document discusses Console Management in the HP Environment from a function and requirements perspective using ConsoleWorks, the HP Console Management Solution from TECSys Development.

What is Console Management?

Console or Out-of-Band management is generally defined as a management scheme that does not use the same communication path to communicate between a character cell terminal and the managed device or console. Out-of-Band management is most thought of as being implemented by a serial communication cable attached to the (RS-232, DB9, RJ45) console or management port of a managed device and a character cell interface such as a VT100 terminal.



In recent years, the definition of Out-of-Band management has been extended to include serial console/device servers, which provide network access to a serial console connection via a telnet or LAT connection. Often console servers provide consolidation of up to 48 separate serial consoles in a 1U rack-mounted device. This type of connectivity provides networked remote access to an otherwise distance limited serial console connection. By utilizing a console server to interface the network to serial console connections, a device administrator can connect to the console over the network from a remote location and execute any management command for the device without having to be physically present.

Management of Console Data

Like information collected by an agent for a SNMP management station, console output from a device needs to be logged, audited, and archived for historical and reporting purposes. Many sites require that console information be printed and archived for several years so that auditing and reporting may be done at a later time should it be necessary.

Because the console terminal for a device is generally considered a “privileged” session and allows privileged functions to occur, access to the console port generally requires physical and local connectivity and is closely guarded as well as audited. This would prevent or at least provide a log of someone changing a device’s configuration, restarting, or shutting it down, denying service.

Information on the console port also generally contains information about the device that could be considered sensitive to the environment and should only be accessed or reviewed by authorized personnel. As a result, having physical access does not always guarantee that unauthorized personnel are not reviewing sensitive information from a console terminal or over a telnet port via the network.

Benefits of Console Management

No Additional Load on the Monitored System

Effectively monitoring a heavily used device with In-Band tools or agents while overcoming the lack of CPU and network bandwidth is one of the largest challenges to overcome with In-Band technology. In-Band monitoring requires the execution of a software agent which generally has to be installed on the managed device by the customer and then the agent is polled every ‘n’ minutes for status and performance information. All the while the agent uses a small portion of CPU, Memory and I/O capacity on the managed device to perform its job.

Console Management simply requires a console subsystem on the device to be responsive, which does not assume that the host operating system is in any condition to respond, but only that the BIOS that controls the port can communicate. This fact alone provides a sobering case for Console Management: connectivity to the device and a chance of recovery even though the software being hosted on the monitored hardware is damaged.

Booting or Bootstrapping a device

When a device is first turned on, it checks the hardware to make sure that it’s working. You must use the console port in order to initially start the operating system or “Bootstrap” the device, change the configuration and add additional interfaces.

Firmware upgrades

To upgrade firmware on a device, the console port is generally used since it is the only reliable interface to the device when the device is in standalone mode. The console port becomes the interface to transfer new firmware code to the device or it is at least used to manage the process of uploading new firmware to the device.

Standalone System Access

Some functions require a device to be in a quiescent state where general access is prohibited. The console port is the only way to access the computer in a standalone configuration

Operating System Maintenance

When operating systems are being upgraded, backed up, or maintained, the console port must be used to accomplish this task.

Crash and Machine Register Access

The console port is where the device reports failures, whether the failure is related to the operating system, application, or hardware. It uses the console port to display the processor registers and other critical environmental information about the failure. It is this information that will be required by field service or other technicians to determine the cause of the failure and whether or not the device must be repaired before being placed back into production.

System Backups

System backups generally are accomplished when the operating system is down, and general production access is stopped. This type of standalone system backup is completed on the system console.

System Configuration

Adding hardware, changing console configurations, changing bootstrap characteristics or which device a computer boots from is only completed from the console terminal.

Privileged Access

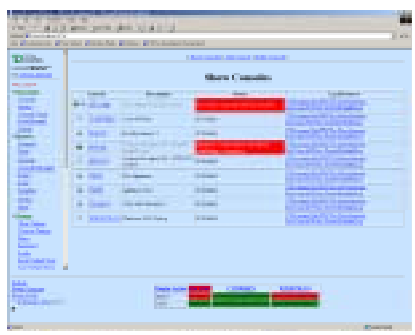
Privileged functions such as real-time debugging, operating system patches, stopping and starting the device, configuration, and power on-/power off of a device are completed on the console port.

Console Management Requirements

If you're going to implement console management, the following requirements should be considered. They are the capabilities that allow complete management of and provide value to console management when it is most needed – when everything else has failed.

Web-Based Console Management

ConsoleWorks is a web server developed to deliver Console management. ConsoleWorks does NOT use any other web server technology like Apache or IIS to deliver its web interface. As a result, security exploits or concerns associated with either of these web servers may be eliminated. It may be hosted on any of the following platforms:



- OpenVMS 7.2 or greater VAX or Alpha
- Tru64 Unix 8.0D or Greater Alpha
- HP-UX 7.x or greater HP RISC
- Linux 7.0 or greater Intel or Alpha
- NT 4.0 SP4 or greater Intel
- Solaris 8.0 or greater Sparc 32/64 or Intel

This provides system administrators and operators remote access to managed devices from anywhere via a web browser or dial-up command line terminal. Remote connectivity is significantly enhanced with the additional secure communication provided by secure socket layer (SSL V1 or V2) communication between the web server and browser/terminal providing almost hack-proof communication with up to 1024 bit encryption strengths determined by the browser's capabilities. Usernames and passwords are also protected from "sniffing the wire" because the SSL connection is established before the user logs on using his or her access control information.

Web based management and monitoring means the support staff takes their management capability wherever they go, as long as a browser or terminal capable client is available to them -- which is most everything from PC's, to Workstations or PDA's. Never again will you worry about a management client being incorrectly installed on a PC or workstation or being out of date. An administrator can do his or her work from home via virtual private networking (VPN) or Serial Link Internet Protocol (SLIP) via dial-in, internet or intranet.

Console Access Restrictions

Several types of console access restrictions can be implemented based on a user's security profile.

Read-Only – Provides the ability to monitor output from the device.

Read/Write – Allows read and write access but may prevent special or protected characters from being sent to the console like BREAK, Control-P unless the security profile has "Control" access and the special character is typed twice consecutively. This allows an operator to interact with the console to perform backups etc, but not accidentally crash or halt the device.

Multiple Read/Write Connections – Basically KEYBOARD WARS! That's right, multiple people connected at the same time to the same console from different places via different mechanisms like x-term, command line and web browser with Java terminal emulator and web browser with third party emulator all at the same time seeing the same thing and interacting at the same time. It's like a telephone party line – individual courtesy is proper and required. This allows a senior engineer or field service person to be on the same console with an operator to help solve a problem with everyone interacting at the same time.

Exclusive Connect – Causes the first person with read/write access to be granted read/write access while others only get read only access even though their security profile may have read/write capability. This allows operating system upgrades to be done without someone accidentally connecting and pressing Return to answer questions for you by default.

Lock Console – Allows a user to lock a console to prevent its use with a reason or cause defined with the lock function. The console may only be unlocked by the user issuing the lock and requires administrative privileges. This allows an administrator to lock a console, preventing its use by an operator doing backups when the console will need to be used at midnight for firmware upgrades or operating system upgrades.

Console Broadcast – Allows a single command to be sent to multiple consoles at the same time without having to connect to each console. It is assumed that each console is in a state where the command would be valid to each device as issued.

Integration into and with In-Band Management Infrastructure

In-Band management systems are often deployed long before the need for console management and monitoring requirements are recognized. While console management solutions are easily integrated with the In-Band infrastructure by simply generating SNMP traps, most In-Band solutions provide for a complete integration using application programmer interfaces as well. This type of integration provides for seamless program-to-program communication and interaction.

Immediate Detection of Monitored Events

Console management systems rely on text characters being generated on the console port of the managed device. Character scan techniques are used to process the text and are limited only by the performance of the monitoring software. As the managed device generates text asynchronously, the console management software immediately sees it. There is no waiting on a poll interval to complete.

Event Management

When an event is detected it is tracked for: what the event means, the event context, who acknowledged the event, when the event was acknowledged and what they did to solve the event. Tracking this information provides a complete and total event history which is used later to provide reporting and also may be used to seamlessly integrate detected events with other tools like helpdesk and other network management solutions like HP OpenView, BMC, Tivoli and Unicenter TNG

Event definitions are defined in Scan files (just like PCM) except that TDI took each vendor's error messages and recovery procedures (from vendors like Compaq for VMS, Tru64, All Storageworks controllers, Alpha, Wildfire, SAN switches or Cisco) and created scanfiles from the vendors based on each vendor's definitions of the event, what they say the event means, and what they say to do to fix the problem. This vendor help is provided with the event as event context sensitive help.

Also, ConsoleWorks allows you to customize the event-specific help to include your own business's best practices, including a log of the previous time the event was solved, so that not only do you end up with event help from the vendor, you also have what your best practices are, examples of when a problem was fixed and the results. All this eliminates ambiguity for an operator or technician working on a problem. They have at their fingertips what the problem is, what is means, how to solve it and an example of what it looks like when it is solved.

Once events have been associated with specific device consoles and configured properly, action routines may be configured to perform well-known tasks that have been identified by the system administrators. These tasks can be as mundane as moving and/or purging log files or as complicated as adding nodes to an active massively parallel processing configuration when active nodes reach some percentage of resource utilization. Event action routines are often used to interface Console management systems into other infrastructure systems including In-Band management systems that may need to know what is going on in the enterprise.

The functionality of action routines is greatly increased when combined with a scheduled event capability. This means that the health of any component, local or remote, can be tested on a predetermined time schedule. If any of these components are found to be inoperable, or unresponsive, events can be generated by the action routines back into the management system to alert an administrator about component failures.

Actions

Actions may be initiated when an event is ***detected*** to provide notification, send email, and execute any user written script based on a timeframe. Additional actions may be initiated when the event is ***acknowledged*** and when the event is ***purged*** from the active event list. As a result of actions being associated with an event state, things like help desk trouble tickets may be automatically created when the event is detected, updated when the event is acknowledged, and closed when the event is purged from within ConsoleWorks. Actions may be initiated based on event detection on a single console, set of consoles, or logical group or groups of consoles. Any given action may also be *scheduled* to occur on a set of consoles or group of consoles. This allows scheduling of actions based on an event occurring e.g. causing an event called "reboot" to cause an action which reboots a machine on a scheduled basis like each night at midnight.

Automatic Reporting Capabilities

Any good management solution will allow a large number of versatile and configurable reports to be run against the data received from the monitored systems. This data, while not very useful on a day-to-day basis, presents very interesting and often significant data about system behavior trends in the enterprise. Mining this data with special emphasis on time correlation between monitored systems often presents surprising insights into unexpected positive and negative feedback that exists between network elements.

Since ConsoleWorks tracks all event activity, this allows a report to be run later that describes what events happened on a console and how many times. A more detailed report can show which events happened on a console, who solved the event, what was done to solve it and how long it took to acknowledge the event since it was detected, effectively providing a Service Level Agreement time for each event. ConsoleWorks also has reports for *Scanfiles, Users, Events, Consoles* and user-generated reports based on a text pattern to match in a log file or set of log files.

Supporting Multiple Connection Types

ConsoleWorks has the ability to support many different connection types for console connections like:

Telnet

Telnet connectors provide a network connection to most if not all terminal servers. This type of connection is generally used for managing a continuous session with remote devices.

Telnet on Demand

Telnet on Demand is used to provide a dynamic telnet session to a managed device through ConsoleWorks. This is typically used when a device is capable of generating syslog output like a router or SAN switch and upon detecting an event a telnet session is required to manage the device. Rather than having to remember where to telnet to or the IP address and port of the remote device, a telnet on demand session may be established. Further, the device may be set up to accept only telnet sessions from the ConsoleWorks host providing additional security for telnet on demand sessions.

Syslog Listener

SYSLOG consoles are receive-only consoles and can be used to receive syslog output from routers, SAN switches, and NT servers where the ConsoleWorks NT event agent has been installed. Once a syslog console receives a syslog packet, it is scanned and managed the same as data from a serial or telnet console. A SYSLOG console may also have an associated Console tied to it so that even though syslog is a receive-only console, when the operator clicks on the console icon for the device, it actually generates a telnet session to the device based on the "associated console" definition. This eliminates the need for an operator to interact with a syslog console any differently than with a telnet console or for the operator to have to distinguish the syslog console from a telnet console.

SNMP trap Receiver

SNMP Trap receiver allows in-band tools or devices to send in-band events or information to ConsoleWorks so that it can be scanned and acted upon by ConsoleWorks. This allows in-band tools

detecting a rule about a device characteristic to be sent to ConsoleWorks and an operator can understand both in and out-of-band characteristics for a given device.

LAT Service

Where LAT is supported in a local area network and LAT services are used to connect to a device where console ports are connected to a terminal server, ConsoleWorks will listen to all LAT service announcements and allow the administrator defining a console to use a LAT service announcement to define the console definition.

LAT Port Definition

When a LAT Service is not used but LAT is the preferred or only supported protocol on a terminal server, the ability to define the server name and associated port on the server is required. By defining this type of console, it eliminates the need to have LAT service announcements on a network.

Interserver – Hierarchal Stacking

The ConsoleWorks interserver connector allows a remote ConsoleWorks server to export a console and all associated traffic to a local server from a remote server and manage the remote console as though it were local. Traffic from the remote console and all console connections to the remote console also show up on the local and remote ConsoleWorks server. This allows data to be logged in both places, scanned for different sets of events or the same set, and can initiate different actions or set of actions on different ConsoleWorks invocations. By using the ConsoleWorks interserver connector, a site may build a hierarchy of ConsoleWorks invocations and selectively export consoles to a higher level in an enterprise view.

Pseudo

TDI recognizes the need to have ConsoleWorks not only manage consoles but possibly also output from an application. By using the PSEUDO console, ConsoleWorks can be used to run an application underneath ConsoleWorks, capture the application output, and trigger events based on that output. This allows ConsoleWorks to track user applications directly instead of scanning an application log file on a polled time interval or rely on an application log file being accessible. ConsoleWorks would of course route application output through it and then log that output in the application's log file generated by ConsoleWorks.

Serial Port

Where ConsoleWorks is being run on a server with multiple direct connected serial consoles, a serial console port would be used. This is particularly useful when a Digi-Board or similar device is installed in a SUN, Linux or NT server and the consoles to manage are in close proximity to the ConsoleWorks host.

User Interfaces

Java VT100 Terminal Emulator Because ConsoleWorks is a web server you may use any web browser to connect to and interface with ConsoleWorks. When you request ConsoleWorks to connect you to the console of a managed device, ConsoleWorks causes a Java based VT100 terminal emulator to be launched from the browser for console connectivity. This eliminates the requirement to install terminal emulation software on the client running the browser.

Third party terminal emulator ConsoleWorks also allows the end user to specify a third party terminal emulator to use instead of the Java terminal emulator on a console-by-console basis.

X-Term Interface If you have an X-window workstation the X-term application may be used instead of the terminal emulator. All of the above allow the end user to preserve keyboard mapping, and utilize their existing environments without additional expense or installing other software.

Command line interface When a browser is not available, ConsoleWorks also supports a command line interface for all supported platforms, OpenVMS, Tru64 Unix, Solaris, NT, Linux, and HP-UX.

Administrative Functions

Put Console in Maintenance Mode ConsoleWorks allows the console to be placed in maintenance mode where access and logging of console traffic are still performed, but scanning for events and initiating actions are not. This allows a device to be shut down, rebooted, have maintenance performed, and then be brought back into production without having to disconnect the console, remove Scanfiles, or trigger actions which notify personnel of deliberate scheduled activity.

Backup or Restore Configuration Database ConsoleWorks provides the ability to arbitrarily save or restore the configuration of the console management solution at any time based on the then current configuration settings or console definitions**.**

Hide Console ConsoleWorks can hide consoles from user view. This feature allows administrators to define console connections, test and verify connectivity and complete a console setup prior to a ConsoleWorks

user seeing it. The console status line may also be user defined, indicating maintenance or field service is working on a console that is indicated down or unavailable.

Expunge Events When multiple consoles have multiple events, they may expunged from one screen without having to deal with each event directly. Also, you may selectively clear events based on filter criteria.

Grouping of Consoles Consoles may be grouped into logical hierarchical groups and the same console may belong to different groups at the same time. This allows groups to be defined based on machine type, operating system, location, function or any business grouping criteria. Groups may have sub-groups or parents. ConsoleWorks allows navigation to a console via group names**.**

Severities Events have several attributes associated with them that are user definable. Severity is one attribute that is user definable, along with the severity color, severity name and a comment-required attribute. This means that any event associated with the severity must have a comment entered to acknowledge the event.

User Sessions ConsoleWorks tracks when a user connects to the web server from any method, and even though the user may not actually be connected to a console, the user's session is monitored and tracked for activity. When a user's session is idle based on a ConsoleWorks administrator timeout value, the session is timed out for security reasons. The ConsoleWorks administrator can turn this feature off if needed. This prevents a user from logging into ConsoleWorks and leaving their session logged in and someone else using that session to gain access to a console.

Graphical User Defined Interface

Enterprise Editor allows a customer to take existing drawings of their network or enterprise, export those drawings as JPG or GIF format files and use them with the enterprise editor tool to define where consoles should be represented on the drawing and where other Groups or Drawing details should be displayed when selected.

Enterprise Viewer is used as a Java viewer interface to navigate ConsoleWorks with the drawings already developed by a customer. This allows customers the ability to navigate to a console based on a graphical representation of an environment with rollup of events to the highest level displayed.

High Availability

ConsoleWorks may be configured for high-availability environments by having an active-active failover configuration. Active-active means that either ConsoleWorks invocation may access any given managed device from the other ConsoleWorks invocation. In effect there isn't a primary and secondary server since either server can take over at anytime if failure occurs. Also, both can provide a user access to a managed device irrespective of which one is ACTUALLY managing the device. Connectivity is transparent to the end user.

Enterprise Hierarchical View

ConsoleWorks has an interserver connector, which allows consoles being managed by one ConsoleWorks instance to be exported and provided seamlessly to another ConsoleWorks instance as though the remote instance was actually managing the exported console locally. This type of connection allows consoles from many instances and associated events from those consoles to be consolidated to a single ConsoleWorks instance, grouped accordingly, and to have information and events processed locally.

Managing NT Servers

ConsoleWorks NT Event Service

NT servers do not generally use a serial console for operator or administrative access. Instead they rely on the keyboard, mouse and video adapter connected directly to the NT server for reboots and administrative access. Several types of remote access have been developed for NT servers, most notably the "screen scraper" applications like PC anywhere, VNC viewer or others. ConsoleWorks will allow usage of those applications by associating them to a console type via the pseudo console or via the informational web link on a console definition if the interface is web enabled.

In general, because NT does not use or understand a console serial port and by default places all events related to an NT environment in the NT machine's event log. TDi built an application that runs as an NT service to listen to NT events for Security, Application and System level events allowing filtering and forwarding of NT events to the ConsoleWorks invocation where the event can be scanned and matched

with well known NT event patterns. As a result, this allows ConsoleWorks to scan NT servers just like any other operating system. When a NT server needs a service restarted or shut down and restarted, a telnet session may be established providing command line access to the server. This is primitive but does allow most administrative functions to be initiated. Also, when the machine needs to be rebooted or BIOS level access needs to be provided, ConsoleWorks works with remote management cards which may be either integral to or installed in most NT servers. Below is a list of Remote Management Cards ConsoleWorks can use for booting and accessing BIOS level functions remotely for NT servers.

Remote Management Cards

Insight Manager Board

Compaq's Insight Manager Board is generally installed in older Compaq PC servers and provides serial access to the onboard interface, which is capable of shutting down and rebooting the server, capturing BLUE SCREEN crash information, and monitoring the server through the boot process. Once the machine is up, ConsoleWorks can run the web interface to the remote Compaq Insight Manager for the machine from within ConsoleWorks.

Insight Manager ASIC

Newer Compaq servers, in particular the new 1u high servers which get racked and stacked have a built in RILO ASIC which supports remote telnet connectivity eliminating the need to plug the serial console into a terminal server. This ASIC allows reset, power on/power off and BIOS configuration via the telnet session or via a web interface. ConsoleWorks manages the integrated ASIC RILO interface using the telnet session. Also, the web interface to Insight Manager may also be linked under the telnet console where ready access to Insight Manager may be achieved.

DRAC Card

When a Dell server is used, Dell has its own version of the Compaq Insight Manager Board for Dell computers. It's called the DRAC card and performs the same function as the Insight Manager Board except for Dell Computers. ConsoleWorks integrates with it the same way as the Compaq Insight Manager boards.

LanDESK

Intel has its own version of the same functionality built into the LANdesk interface for Intel Servers. Again, ConsoleWorks manages those interfaces the same as the Compaq Insight Manager card.

PCWEASEL

PC Weasel is a third party BIOSlevel remote manager board that is generic for ISA and PCI buses. It will generally work in most PCs but does not contain the vendor specific functions that the DRAC and RILO cards do. It does generally provide for BIOS management and reset and reboot capabilities. It is also user customizable or programmable. ConsoleWorks manages the PC weasel card just like the rest.

Microsoft .NET Servers

Microsoft has finally gotten the message and has allowed the .NET servers to recognize and utilize the serial console of a NT server just like other operating systems. As a result you can now start and stop services and process and perform administrative BIOS level functions on the console serial port. This is particularly valuable when a customer has many 1u high PC servers racked in a high-density rack. TDi feels this has opened a whole new market for NT server management for ConsoleWorks.

Managing Storage Controllers

HSD, HSJ, HSC, HSZ and HSG Storage Controllers

All HS series controllers prior to the HSV controller used a serial console for initialization, setup and configuration via a CLI. While Compaq did introduce the Storageworks command console, it generally required the use of the agent on a host system and also required the use of the SCSI bus for communication.

By taking the serial console port and plugging it into a terminal server, ConsoleWorks provides access to the controller, the CLI and all out-of-band events. ConsoleWorks HSx controller scan file provides the translations of all HSx event codes for HSx series controllers. As a result when the hex event code is detected, ConsoleWorks translates the event code into user readable form without having to look up the event code in a book for its meaning.

HSV Storage Controllers

With the introduction of the HSV controller, Compaq Storage engineering did away with the CLI interface much to the surprise of many customers. Instead the end user is required to use an In-Band management tool to query the controller with SNMP or a remote CLI interface for some command line access. ConsoleWorks supports the ability to use SNMP and query the console or execute the CLI client. Compaq's HSV controller supports the fiber alliance MIB and SNMP processing from which ConsoleWorks is capable of querying and processing SNMP events .

Clarion, EMC Symmetrix

Clarion and EMC both support CLI level access to manage, configure and monitor their controllers. Similarly to the HSx controller, a serial console may be connected to a terminal server and ConsoleWorks can then manage these controllers as well.

Managing Network Devices (Routers, Switches)

CISCO

Cisco routers and switches all either support a telnet management connection or a serial console connection. ConsoleWorks understands all Cisco error messages, which come from its management interface. Using the Cisco Scanfile for ConsoleWorks allows ConsoleWorks to react to any Cisco out-of-band event.

Managing SAN Controllers

Brocade

Most SAN controllers support either the Fiber Alliance SNMP MIB, a telnet management session, or a serial console connection. Like other host systems, storage controllers or network devices, these devices also generally use a CLI interface for configuration and management.

ConsoleWorks has specific Scanfiles for Brocade SAN Switches used by most Compaq installations. TDI developed a close integration with the events in the Brocade infrastructure. From providing a monitored out-of-band console interface to scanning the SNMP status of the switch itself, ConsoleWorks provides the monitoring and management needed.

Managing Power Controllers

Telnet and Serial Interface

Like other devices, power controllers that are "Smart" have a console subsystem with a command line interface to allow monitoring and management of the power controller. This includes power off, voltage monitoring and power on of the whole unit, specific ports or devices connected to the power controller. ConsoleWorks manages and interfaces with this type of device just like anything else. The difference is that ConsoleWorks can provide a definition of which device is plugged into which port on the power controller.

Support for multiple Vendor Terminal Servers

TDI has not run across a terminal server we do not support. However, TDi does have partnership agreements with Cyclades, Lantronix, and DNPG (formally Digital Terminal Servers). We have customers using Cyclades, Digi, Cisco, Pearle, ITouch and others. The general rule of thumb is: if the terminal server has a reasonable IP stack and you can telnet from the host where ConsoleWorks is going to run, then ConsoleWorks can support that terminal sever.

Where SSH is supported on the terminal server, whether it is SSH V1 or SSH V2 or no SSH support, ConsoleWorks can work with it. This is because you may use your chosen SSH client under the console port definition for each console. This enables some consoles to support SSH V1, others SSH V2 and others no SSH support at all without impact to ConsoleWorks.

Conclusion

Console management is easily integrated into the enterprise whether in-band management systems exist or not, giving the enterprise architect a valuable tool to extend the command and control capabilities available to infrastructure operations staff. Utilizing both technologies in the HP Enterprise provides a comprehensive integrated solution for total visibility of the complete infrastructure utilizing either technology.

Additional Information

For additional product information, a live demonstration, or current pricing, please call 1-800-695-1258 or visit our web site at http://www.tditx.com/.