SOFTWARE DEPLOYMENT AND CONFIGURATION OF THE CERN ACCELERATOR CONTROL ROOMS CONSOLES

M. Albert, G. Crockford, E. Hatziangeli , S. Lopienski CERN, Geneva, Switzerland

Abstract

The Windows 2000 configuration project was launched to set up the operational consoles, accounts, deployment procedures and packages of accelerator controls software on the new Windows 2000 operational consoles in the SPS Accelerator Control Room, the SPS surface buildings and the SPS experimental zones, in an automatic and reproducible manner. The configuration of the operational console environment is achieved by a combination of Windows profile settings for the Operations user account and group policy settings applied to security groups. A major restriction for all operational consoles (being located either in the CERN Accelerator Control Rooms or in one of the SPS surface buildings), is the fact that only a small set of users are allowed to perform actions in a restricted domain, in order to ensure secure access to the Accelerator Controls application software.

INTRODUCTION

The C-based operational software for the SPS accelerator is undergoing a major re-engineering to allow the most efficient exploitation of the SPS machine. Object-oriented technologies and design principles are consistently applied for the new control system application software. Java is the programming language chosen for the implementation of the high-level services and control room applications, as it enables platform independent development. In the SPS control room, surface buildings and experimental zones, desktop PCs were chosen as a suitable platform for the operational consoles to run the Graphical User Interfaces (GUI). These PCs are running Windows 2000 Professional using the NICE computing environment [1] (CERN Windows 2000 client-server architecture).

MOTIVATIONS

Moving away from UNIX X-terminals to Windows based PC consoles required serious changes in the present strategy for software deployment. In addition, the vulnerability of the Windows 2000 operating system to local modifications necessitated a critical look in the set up and configuration of the consoles. Issues like protection against viruses, stability of the consoles' running environment and good start-up performance of the application software had to be addressed. Besides having to deploy Java, X-Motif legacy applications and general-purpose tools, the deployment had to be done in consoles located outside the SPS Control Room in remote locations. This required an investment in automated deployment procedures in order to minimise manual interventions.

OPERATIONAL CONSOLES SET-UP

In order to achieve a reliable and safe operation, standard Windows administration procedures were used to configure and manage the operational consoles through the use of customised profiles and security groups with restricted privileges. Groups in the Windows 2000 Active Directory [2] manage domain user access to domain resources by assigning permissions once to a group rather than multiple times to individual users or computers. There are two types of groups in the Active Directory, security groups and distribution groups.

Security Groups

Security groups are used to assign permissions to groups of users and computers. Users or computers can be members of more than one security group. Two security groups have been created and populated with computers and users:

- *Control room users and computers*, for all control room consoles and operational user accounts.
- *Non-control room users and computers*, for all consoles located in the SPS surface buildings, experimental areas and their associated operational accounts.

The operational consoles are classified according to their geographical location and usage in the context of accelerator control. Different security settings were separately applied to the corresponding groups via Windows 2000 Group Policy Objects [2].

Group Policies

Windows 2000 Group Policy (GP) is a technology that provides administrative control over users and computers in a network. By using GP, one can define the state of a user's work environment once and then rely on Windows 2000 to continually enforce the settings that are defined. Operational user and computer settings are controlled via dedicated Group Policy Objects (GPO), which were created by the NICE 2000 domain administrators and apply to the above security groups. These two GPOs are:

- *Control room Operations*, applied to security group *Control room users and computers.*
- *Non-control room Operations*, applied to security group *Non-control room users and computers.*

Group policy objects were used to automatically assign general-purpose applications and to enforce specific desktop and security settings to ensure that all consoles provide the same look and feel and behave identically.

Deployment of services

General-purpose services and tools (JVM, JaWS, NetBeans) are deployed in an automated manner, using

Microsoft Installer (MSI) [2] files. These are Windows installer packages that include the software application to be distributed, package configuration and identification information. MSI packages were created using the *snapshot* technology, which comprises of the following steps:

- Make a "before" snapshot of disks and registry contents of a cleanly installed machine using Veritas WinInstall tool [3]
- Install the software to be packaged on the machine and make an "after" snapshot
- Take the difference between the "before" and "after" snapshots, convert it to an MSI file, and edit it using the MS database table editor Orca [4].

The MSI file is added to the group policy, which was used to deploy the MSI packages on computers that are associated with these policies. Once a package is assigned to the console computers, the software installation and maintenance feature installs automatically the application packed within the MSI file, when the computer is restarted and removes an older version, if necessary.

Operational Console Environment

One CERN Nice 2000 domain account is set up to be used by the operations crew to log on to the operational consoles in the SPS Control Room for the sole purpose of controlling the particle accelerator. Additional domain accounts were provided for equipment experts and operators requiring access to consoles outside the control room. To ensure the operational account environment remains stable and unaltered and to enforce a common and unchangeable look and feel across all operational computers, the following settings were enforced:

- Mandatory user profile to avoid de-synchronisation of profile settings on the various operational consoles, which might be introduced by a roaming profile. The centrally stored profile is restored to the consoles after each logon. Certain modifications (for example: adding of shortcuts onto the desktop, mapping of network drives) during a session are allowed but will be irreversibly lost at logoff.
- Limited privileges to avoid users altering the environment of the operational consoles, thus ensuring stability of the consoles
- Users have no software installation privileges. They are prohibited to run commands that could alter the settings of the consoles and to edit the local registry.

Running in a secure environment

The configuration of the operational console environment is achieved by a combination of profile settings for the operational accounts and GP settings being applied to security groups to which the operational accounts belong. A major restriction for all operational consoles, being located either in the SPS Control Room or in one of the SPS surface buildings or experimental zones, is the fact that only a small set of users are allowed to perform a domain login. A strict separation between operational consoles, destined to control the SPS accelerator and its transfer lines, and machines for software development was enforced. This meant that operational accounts were denied access to non-operational consoles and respectively personal accounts were prohibited from logging in to operational consoles. This list of users is specified in the policy setting ***Log on locally*** which is defined in the computer configuration settings of both group policies. To restrict the login rights of the operational accounts to the operational consoles only, a list of computers where operational accounts may logon has been explicitly named.

In order to avoid compromising the operation of the accelerators, all operational consoles are connected to a dedicated Controls network, which is not accessible from outside CERN. Operational consoles don't have access to the Internet. For remote installed consoles outside the SPS Control room, an automatic log off was enforced after 30 min of idle time to limit the possibility of access by unauthorized users.

Moreover, unauthenticated applications will only run in a restricted environment, using limited resources; access to the local hard disk and the network is restricted for non-trusted applications. Only CERN certified applications have unlimited access to resources.

JAVA SOFTWARE DEPLOYMENT

Choosing Java as a programming language and a platform for the new SPS operational software raised the question of how to deploy those Java applications. Due to the distributed nature of all the operational consoles, the software deployment and installation process had to be automated.

In addition, when deploying Java software, one should consider not only the executables and their related resources, but also a unique-to-Java piece, the Java Virtual Machine (JVM) and its runtime support. This makes Java deployment rather complicated, as the client platforms do not include a pre-installed, up-to-date Java Runtime Environment (JRE). The issues on performance, and the automatic software installation and distribution were resolved by employing standard Java deployment techniques based on the Java Network Launch Protocol (JNLP) [5] and royalty-free Sun's Java Web Start (JaWS) [6] software that implements this protocol.

The Java Network Launch Protocol and Java Web Start

Java Web Start is a mechanism that allows Java-technology-based applications to be downloaded and launched directly from a standard Web server. Users, who have Java Web Start installed, just click on a link pointing to a JNLP file to launch a desired application. Java classes and all necessary resources are copied over HTTP to a client's hard disk and executed locally outside the scope of the browser (Figure 1). Once deployed locally on the console, the programs do not need to be downloaded again. The next time the user starts the application, JaWS would compare the cached copy with the files available on the Web server and automatically download updates, if available. This process is transparent for the user and always guarantees that the latest version of the software is used. Another advantage, apart from good start-up performance after the first-time download, is that the users can still run an application from a cached copy in case the network is unavailable.

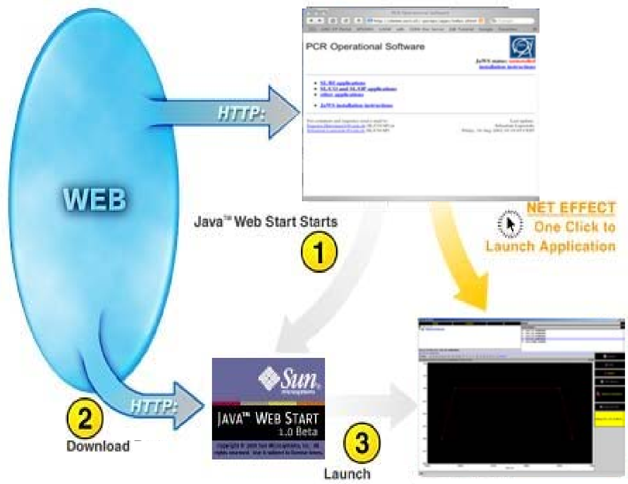

Figure 1:Java Web Start Deployment

Packaging for deployment

For JNLP deployment, Java classes and application resources have to be packed into one or more JAR files. In addition, the developer has to prepare a JNLP descriptor file for each deployed application. This is an XML file that describes an application and contains all information necessary to run it: name of the application, location (URLs) of JAR files, name of the main class, program arguments, system properties etc. One can also specify which JRE (if present) should be used to run the application. Other options (like security information) are optional and only required when the applications need to work outside the *sandbox*. SPS operational software requires unrestricted access to resources and hence it has to run outside the *sandbox*. In this case the corresponding JNLP file contains a request for all permissions and application's JAR files need to be signed by a certificate recognized by JaWS.

Software security and certification

All the SPS operational JAR files are signed, with a trusted CERN certificate, which is recognized by JaWS and allows them to run outside the protected container. In the case of unsigned JAR files, JaWS provides a collection of interfaces and classes for working locally on the client machine, even from a non-trusted environment, making tasks like printing possible.

BENEFITS AND CONCLUSIONS

The benefits of using Windows security groups in combination with group policies are:

- A new console needs only to be added to a corresponding group via the standard Windows administration tools and after a reboot it will automatically receive all applications and resources needed.
- Any modification in the policies will be distributed to all computers, which are members of the associated security group.
- Manual intervention to the set up and configuration of the consoles is reduced to a minimum.
- General-purpose tools are installed automatically during the next reboot, once it is assigned to the security groups.

The benefits of using JaWS and JNLP as deployment technique are:

- A Web-centric approach to deploy and run Java applications
- Automatic installation of any resource (jar files, extensions, native libraries)
- Transparent incremental update, as only the changed resources are downloaded when an application has been updated
- Centralised management of different JRE versions.
- Security features like signed jars, hence singed applications and the tuning of permissions allow implementing an environment with different levels of restricted execution.
- Application executables and resources are cached locally, which ensures high start-up performance.

In summary, the use of Windows 2000 administration techniques enabled us to ensure stability and integrity of the operational consoles settings allowing for a secure and reproducible environment. Software deployment using JaWS required a low cost investment for the developers with a minimum implementation effort. There is a small cost to the users during the first time launch as the application and its resources are downloaded locally. Subsequence launches are very performant as the application remains always ready to be launched from the local cache.

REFERENCES

[1] http://winservices.web.cern.ch/winservices/
[2] http://www.microsoft.com/windows2000/en/advanced/
[3] http://eval.veritas.com
[4] http://msdn.microsoft.com/
[5] http://java.sun.com/products/javawebstart/jnlp-spec-log.html
[6] http://java.sun.com/products/javawebstart