San Onofre 2/3 Simulator: The Move from Unix to Windows

Christian Paquette
Christian.Paquette@cae.com
CAE Inc.

Cherif Desouky
Cherif.Desouky@cae.com
CAE Inc.

Vincent Gagnon
Southern California Edison
gagnonv@songs.sce.com

ABSTRACT

CAE has been developing nuclear power plant (NPP) simulators for over 30 years for customers around the world. While numerous operating systems are used today for simulators, many of the existing simulators were developed to run on workstation-type computers using a variant of the Unix operating system. Today, thanks to the advances in the power and capabilities of Personal Computers (PC's), and because most simulators will eventually need to be upgraded, more and more of these RISC processor-based simulators will be converted to PC-based platforms running either the Windows or Linux operating systems. CAE's multi-platform simulation environment runs on the UNIX Linux and Windows operating systems, enabling simulators to be "open" and highly interoperable systems using industry-standard software components and methods. The result is simulators that are easier to maintain and modify as reference plants evolve.

In early January 2003, CAE set out to upgrade Southern California Edison's San Onofre Unit 2/3 UNIX-based simulator with its latest integrated simulation environment. This environment includes CAE's instructor station Isis™, the latest ROSE® modeling and runtime tool, as well as the deployment of a new reactor kinetics model (COMET™) and new nuclear steam supply system (ANTHEM2000™). The chosen simulation platform is PC-based and runs the Windows XP operating system. The main features and achievements of the San Onofre 2/3 Simulator's modernization from RISC/Unix to Intel/Windows XP, running CAE's current simulation environment, is the subject of this paper.

INTRODUCTION

It is because the actual simulation system was more and more difficult and expensive to maintain that Southern California Edison decided to migrate the simulator onto a Windows-based PC platform. CAE was selected based on its technical expertise, advanced modeling tools and knowledge of the SONGS simulator.

The project was divided into 2 distinct phases. Phase 1 consisting of the following:

- Re-host the simulator software from obsolete computers and proprietary Operating System (OS) to a low-cost, scalable, easy to maintain platform (hardware/software).
- New state-of-the-art, user-friendly Instructor Station (IS) compatible with the new platform.
- Re-host several Plant Process Computers (PPCs) and improve the communication interfaces.
- Upgrade the ROSE® modeling tools and port existing ROSE database from Unix to Windows.

Phase 2 consisted of replacing the Reactor Core model, the Reactor Coolant System (RCS) model, the RCP Oil system model, the Steam

Generator Blowdown system model, the Containment model, and several Radiation Monitors.

The following describes in more detail the work required and performed during the two phases.

LEGACY SIMULATOR SYSTEM

The previous SONGS plant simulator was hosted on an SGI Challenge Server where the majority of the modeling programs were running. The Simulator Server operating system was SGI IRIX (Unix) version 5.3. NCD X-terminals as well as standard Personal Computers (PCs) were used as workstations running the older CAE Unix-based instructor station.

The interface with the hardware panels was accomplished using an HSD VMIC VMIVME-5610 IO interface.
The simulator included interfaces to several plant computers:

- **COLSS** Core Operating Limit Supervisory System
- **PMS** Plant Monitoring System
- **CFMS** Critical Functions Monitoring System
- **QSPDS** Qualified Safety Parameter Display System
- **CEAD** Control Element Assemblies Display
- **DAS** Data Acquisition System (Radiation Monitoring System)
- **SWCS** Secondary Water Chemistry System
- **FMS** Fire Monitoring System
- **SCMD** Sub Cooled Margin Display

Figure 1 shows the SONGS simulator system before the recent re-host.

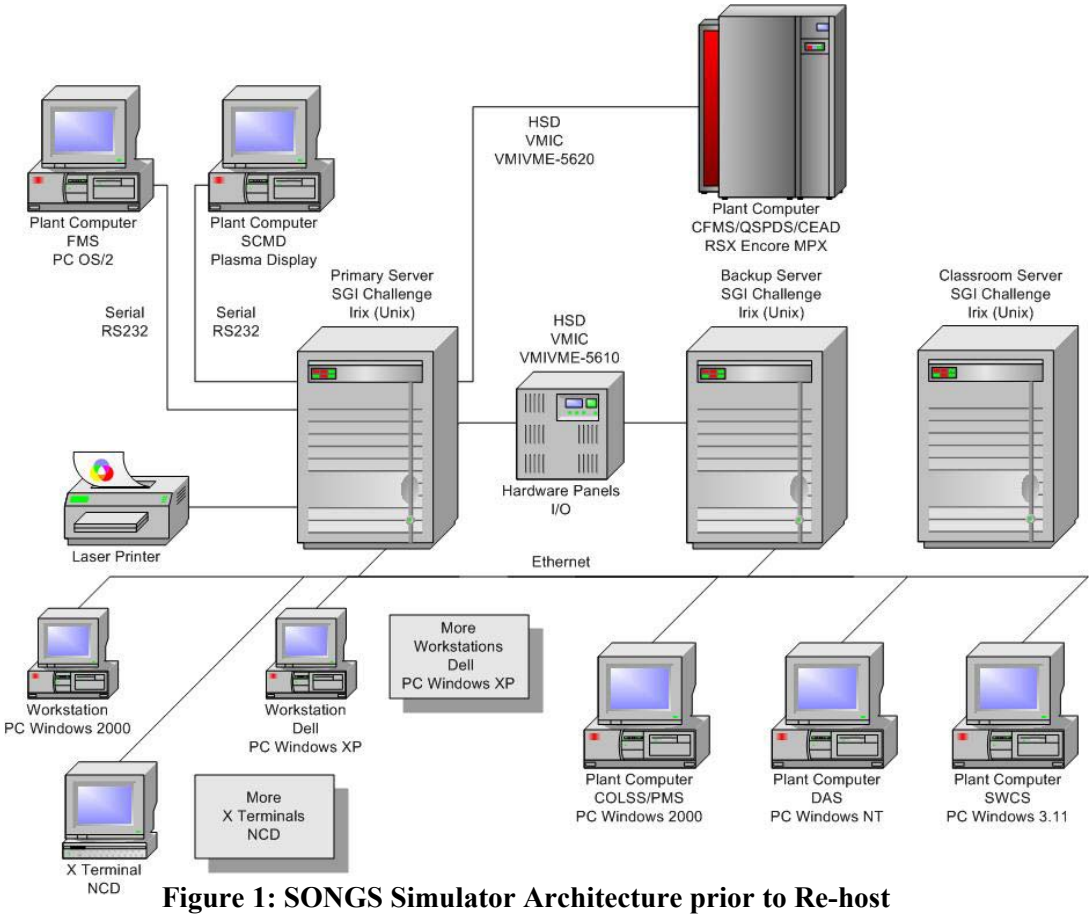

Figure 1: SONGS Simulator Architecture prior to Re-host

NEW SIMULATOR SYSTEM

The new system consists of three (3) servers, six (6) new workstations, and approximately four (4) existing workstations.

The servers have been designated primary, backup and classroom servers, all located in the simulator computer room. The network is a switched Ethernet TCP/IP Local Area Network (LAN).

The simulation computer is a HP/Compaq ProLiant DL580 G2 Server with the following specifications:

- Quad Intel Pentium 4 Xeon processor at 2.0 GHz, 2MB cache memory
- 2 GB 200Mhz DDR ECC SDRAM Memory
- Qty.2 18 GB Ultra-320 SCSI, 15,000 RPM Hard Disk Drive
- Qty.4 73 GB Ultra-320 SCSI, 10,000 RPM Hard Disk Drive
- 48X/20X/48X IDE CD-RW drive
- 1.44 MB 3.5" Floppy Disk Drive
- Integrated 10/100 Network Interface Card
- 3Com 3C905C-TX Network Interface Card
- VMI PCI Interface Card (IO Communication)
- Integrated 8MB, VGA Graphics Card
- NEC MulltiSync LCD 1920NX

The two 18 GB hard disk drives are configured in a RAID 1 (mirrored) configuration. The disks are partitioned in two partitions (c:\ and d:\).

The c:\ partition contains the operating system and commercial-off-the-shelf applications (SQL Server, Visual Studio, etc.). The d:\ partition contains CAE's software.

The four 73 GB hard disk drives are configured in a RAID 5 configuration. Three disks have data striped across them and one disk is reserved as a hot spare.

The Engineering Station is an HP Workstation xw5000 with the following specifications:

- Intel Pentium 4 Processor at 3.06 GHz, 533Mhz FSB, 512Kb Cache
- 512 MB PC2100 ECC DDR SDRAM

- 80GB EIDE 7,200 RPM ATA/100 Hard Disk Drive
- Matrox G45 Dual Head PCI Graphics Card
- 48X/20X/48X IDE CD-RW Drive
- 1.44 MB 3.5" Floppy Disk Drive
- Integrated 10/100 Mbps Network Interface Card
- NEC MulltiSync LCD 1920NX
- Microsoft Intellimouse 2 button, with scroll mouse
- Standard PS/2 Keyboard
- Minitower Option

Figure 2 shows the SONGS simulator system after the re-host effort was carried out.

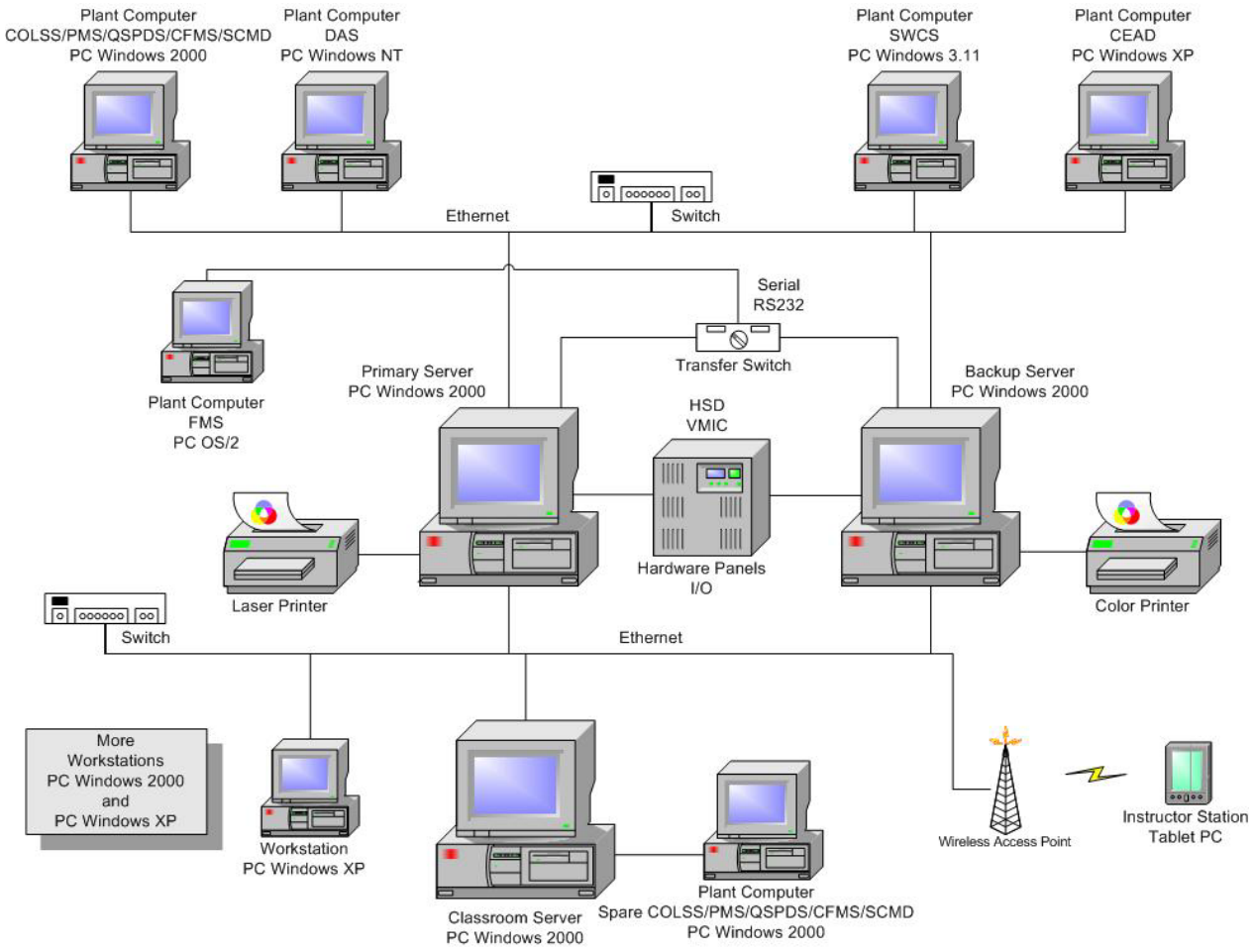

Figure 2: SONGS Simulator Architecture following Re-host

TOTALSIM™ SIMULATION ENVIRONMENT

The computing world is constantly changing. Computers become more and more powerful and at the same time, less and less expensive. This, combined with the recent emergence of serious economical constraints, resulted in an evident migration trend, in the simulation world, from workstation type computers using a variant of the Unix operating system to PC-based platforms running under either the Windows or Linux operating systems. Sometimes, the migration is total, that is, the whole simulation platform is replaced. Sometimes, the migration is partial and only some parts of the simulation platform are replaced, creating highly heterogeneous computing environments.

CAE's multi-platform simulation environment – TotalSim™ - has been designed to accommodate this kind of heterogeneous computing world. CAE's TotalSim™ environment, supported on Unix, Linux and Windows, provide simulators that are designed to be "open" and highly interoperable systems using industry-standards like Java where portability and security are paramount and XML (eXtensible Markup Language) for the communication between the different components making up the system.

TOTALSIM COMPONENTS

CAE's TotalSim is made of the following components:

- jSimex™ (part of CAELIB)
- "Connectors"
- Simulation Database
- DBM™ (Database Manager)
- Isis™
- CTS (Computerized Testing System)
- PFU (Performance Utility)
- IC Editor
- SCL (Simulation Component Loader)

A brief description of each component follows.

jSimex™

In CAE's simulation world, a configuration is made of:

- ROSE® schematics and non-ROSE modules
- Panel and System schematics
- Simulation Database

jSimex can be considered the heart of the TotalSim environment. It is a Content Management System managing simulator configurations through standard commands like Configuration Create, Update, Copy, Delete and so on. Note that CAE's graphical model development and runtime tool, ROSE, has its own built-in Content Management System and that jSimex doesn't attempt to duplicate ROSE 4 content management functionality. A jSimex-ROSE connector (see below) was designed to let jSimex 'see' the ROSE model builder as a service provider (offering among other things a Build Schematic service) and its database as an external data store where the schematics reside. Since jSimex is also responsible for creating runtime simulation processes out of these simulation configurations, the connector has been made such that it can detect changes made to the schematics belonging to any given configuration. In the case changes are detected, jSimex instructs ROSE to rebuild the modified groups of schematics before retrieving the generated codes and firing the actual simulation processes build command.

During the jSimex runtime simulation build process, the simulation database (see below) is built and a binary image (runtime database) is created on disk, ready to be loaded into memory. ROSE-generated modules as well as non-ROSE modules are mapped onto the runtime database, compiled and then linked to create executables also ready to be loaded. The actions described just above are Simulation Executive dependent and are executed through another connector, which connects the TotalSim environment to a Simulation Executive. For the SONGS project, CAE's CAELIB Simulation Executive has been selected.

jSimex supports simulation control commands like Simulation Load and Unload. It has been designed as a client-server application. The client side is written is Java. The server is written is standard C. jSimex currently runs on Windows, Linux and Unix.

"Connectors"

As explained above, the TotalSim environment communicates with the external world using dedicated software connectors. "Connectors" exist now to interface jSimex with CAE's ROSE model development facility and CAELIB Simulation Executive. The connector-based architecture employed here makes the TotalSim software highly adaptable to any third-party operating environment.

Simulation Database

A key feature of CAE's TotalSim environment is the simulation database, which is now an off-the-self relational SQL database. In the case of this project, Microsoft SQL Server has been selected. Other databases like PostgreSQL are currently being used on the Linux/Unix platforms. The simulation database contains all the information needed by the simulator, such as:

- variables used by simulation models
- control room panel information (switches, lamps, meters and annunciators, …)
- data required by the instructor station (panel override labels, malfunction and remote function labels, switch check associations, …)

A software tool has been created to give and control the access to the simulation SQL database. This tool, the DBM, is described below.

At runtime, an image of the SQL database is loaded into memory. The various simulation processes communicate through this memory-resident image.

DBM™

CAE's Data Base Manager software tool, DBM, has been designed to provide and control the access to the simulation SQL database. It is used as a front end to input to, delete from, or otherwise manipulate the database. Any SQL-type commercial database application can be used.

The following basic commands/functionalities are supported:

- Add *label*
- Delete *label*
- List/Search *label*
- Modify *label*
- Clean Database (Reduce fragmentation)
- Support Batch files as inputs
- Support files as output

CAE's DBM has been designed as a client-server application. The DBM thin client runs inside a browser (like Internet Explorer and Netscape) and can be securely used from anywhere without having to install any special software. The server side is built on top of a fully portable Java-based Web server (Apache/Tomcat).

Isis™

Isis is a client/server instructor station application used to monitor and control CAE simulators. Completely Web enabled, Isis incorporates a state-of-the-art Java server offering a portable, flexible and functionally rich software solution. Isis's XML communication layer allows any type of client (from a Linux client to a PDA) to connect to the simulator server with minimum effort. The following clients are available:

- A very rich Windows-based client that runs on both desktops and Tablet PC's
- Pocket PC and Palm which can be used as Instructor Remote Controller

The rich Windows-based client offers state-of-the-art interface views to the Instructor to maximize training session management and development. These views are equipped with search and filter features, which are provided to make the operation of Isis extremely efficient. The Lesson management system, which is of prime importance in the nuclear power plant-training world, incorporates advanced built-in security features. Lesson management is performed using a Flow-Chart type editor which lets the user build comprehensive lessons and execute them in a very graphical and intuitive manner, with minimum effort.

Isis will soon extend industry-training standards by offering a framework for managing Training/Testing sessions in a very efficient manner, the Training Performance Review system (TPR) (not implemented on the SONGS project). The TPR comes with a user-friendly Graphical User Interface through which the user identifies pertinent information like:

- The Scenario used: filename, location, creation date and time.
- The Date/Time of the Lesson Plan execution.
- The intended purpose of this exercise i.e. Training or Testing.
- Personnel that are taking part in the exercise, i.e. students, instructors, NRC visitors etc.,

along with their positions as it pertain to the exercise.
- Simulator DR's open at the time of training.
- At the end of the scenario, the Instructors will be able to add additional comments and insert Pass/Fail status for students participating in the exercise.

Based on information provided by the Isis Lesson Manager, Simulation Action Monitor and Performance Review systems, the TPR will be able to generate different types of reports.
The TPR framework also provide for automatic regression testing. Regression Testing is paramount for achieving proper development and maintenance of simulator software. Automated Regression Testing is essential in making this process efficient. Redundant and repetitive functions usually associated with regression testing are simply eliminated, which allows the testing engineer to concentrate on the analysis of the results.

CTS™

CTS is a front-end application that provides a friendly graphical user interface to the engineers for real-time monitoring and debugging purposes. It can read and write to CDB variables on the fly, it can also monitor any CDB variables in real-time.

PFU™

PFU is a graphical utility provided to monitor the performance of individual and/or groups of real-time programs. It provides accurate measurements of execution time. Results can be presented either in tabular or chart forms.

IC Editor

CAE's IC Editor has been designed as a client-server application. The IC Editor thin client runs inside a browser (like MS Explorer and Netscape) and can be securely used from anywhere without having to install any special software.

The IC Editor software tool provides a quick and easy way for maintenance engineers to examine and/or modify label values of existing store points (ICs) without having to load the simulation. Given an init file that contains a list of labels and their corresponding values, store points can be updated according to these new values either one by one or by group.

Simulation Component Loader (SCL™)

The Simulation Component Loader (SCL) provides generic users with a quick way for setting-up the simulation environment. That includes loading the Web context servers for Isis, CTS, DBM, IC Editor, as well as the simulation itself including any Plant Process Computer software.

The SCL has been developed on top of JXTA, a well-known peer-to-peer framework. Computers equipped with an instance of SCL become part of a virtual network where each member automatically discovers other members, and then exposes its services to be used by others in a peer-to-peer manner.

ROSE® MIGRATION

It was critical that SONGS safeguards its investment in the existing ROSE models and engineering expertise. Therefore, one of our larger challenges was to port all existing ROSE schematics from a Unix environment to a Windows environment. Basically, the older system was designed using Unix-based ROSE 2.3d, which is about 10 years old. CAE had to convert all these old schematics and libraries to ROSE 2.4a (Unix) and then to ROSE 3.3, the original Windows package. That version of ROSE had been deployed and used in several other projects so its maturity permitted us to save a tremendous amount of time in that part of the re-host. The real challenge started when ROSE 4 was introduced. The ROSE 4 design is based on SQL and all the generated sequential codes are written in C++. This state-of-the art new technology has lots of new features to offer and has room for even more improvements. A conversion tool was written in order to convert

the ROSE 3.3 schematics and libraries to ROSE 4. This conversion tool can also be used for existing and future projects.

SOFTWARE DEVELOPMENT AND MAINTENANCE TOOLS

TotalSim uses the latest, most popular software programming, integration, and debugging tools to develop and maintain the simulator software. The tools include:

- Borland JBuilder X to develop and maintain TotalSim server-side software written in Java.
- Microsoft Visual Studio to develop and maintain TotalSim client software whenever this software is written in C++ as well as the software drivers, written in C, used for the communication between the hardware and software components in the simulator
- Compaq Visual FORTRAN to develop and maintain the FORTRAN modeling programs
- Microsoft SourceSafe to control the configuration of the entire simulator programs.

CONCLUSION

So far, the project is on track for completion on time due mainly to the healthy cooperation between CAE and Southern California Edison. Although this project involved significant challenges related to the integration of several new tools (the enhanced Isis, ROSE 4, etc.), the upgraded SONGS simulator should be ready for training by mid-March 2004, 15 months after the contract was awarded.