Full-Scope Nuclear Training Simulator -Brought To The Desktop

D. J. LaPointe, A. Manz, G. S. Hall
SIMULATOR SERVICES DEPARTMENT
TRAINING AND SIMULATOR SERVICES DIVISION

Introduction

RighTSTEP is a suite of simulation software which has been initially designed to facilitate upgrade of Ontario Hydro's full-scope simulators, but is also adaptable to a variety of other roles. It is presently being commissioned at Bruce A Training Simulator and has seen preliminary use in desktop and classroom roles. Because of the flexibility of the system, we anticipate it will see common use in the corporation for full-scope simulation roles.

A key reason for developing RighTSTEP (Real Time Simulator Technology Extensible and Portable) was the need to modernize and upgrade the full-scope training simulator while protecting the investment in modelling code. This modelling code represents the end product of 18 years of evolution from the beginning of its development in 1979. Bringing this modelling code to a modern and more useful framework – the combination of simulator host, operating system, and simulator operating system – also could provide many spin-off benefits. The development (and first implementation) of the RighTSTEP system was cited for saving the corporation 5.6M$ and was recognized by a corporate New Technology Award last year.

The most important spin-off from this project has been the desktop version of the full-scope simulator. The desktop simulator uses essentially the same software as does it full-scope counterpart, and may be used for a variety of new purposes. Classroom and individual simulator training can now be easily accommodated since a desktop simulator is both affordable and relatively easy to use. Further, a wide group of people can be trained using the desktop simulator: by contrast the full-scope simulators were almost exclusively devoted to front-line operating staff. The desktop is finding increasing use in support of engineering applications, resulting from its easy accessibility, breadth of station systems represented, and tools for analysis and viewing.

As further plant models are made available on the new simulator platform and further tools are developed to enhance the system, all users of the system will benefit from these improvements. We feel this will have meaningful benefit to training and engineering analysis users who will in turn be better able to contribute to nuclear recovery in Ontario Hydro.

Major Features of RighTSTEP

Core Simulation Executive

At the heart of a real-time simulator system lies the simulator executive. Its primary function is to manage the computer resources necessary to achieve that end.

The RighTSTEP simulator executive was designed essentially from scratch. Although this was necessitated by a complete platform change - new computer, new vendor operating system - from the old simulator , a secondary intent was to avoid by design some long-standing problems with the latter. In particular, one of the main tasks of the executive is to ensure the plant process models are scheduled and executed in a predictable manner consistent with real-time behaviour. Unpredictable operation, in the form of different outcomes in response to identical inputs, was an issue with the old system. Though this was not common, it was a source of some concern in high-profile simulator uses, for example, authorization or refresher testing. Deign features implemented to combat this included:

- Memory locking of key resources.
- An enforced sequence of module scheduling.
- A cross-check to ensure the rate of simulator time passage is a constant of real time passage. Typically this is a ratio of 1:1. If the simulator departs from real-time execution, it should be shutdown immediately as unpredictable operation may ensue.

As an alternative to real-time execution, a development mode was included where the simulation runs as quickly as possible but at normal priorities.

Other key features of the simulator executive include:

- Industry-standard functions used wherever possible. POSIX 1003.4, ANSI, X11R6 are the most applicable standards and have been followed.
- Multiple independent simulators to operate on any machine, and multiple users of any one simulation either locally or across the network.
- Provide for 'tightly coupled' multi-CPU expansion through symmetric multiprocessing (a number of CPUs within a single computer) or expansion to 'loosely coupled' processes via reflected memory unit (a method of connecting separate computers).

Graphic User Interface

As used here, the term 'user interface' connotes the facility which allows a user to control and monitor the running of the simulation. This includes not only its execution, but also the insertion of malfunctions or the control of manipulated parameters. In RighTSTEP, this takes the form of a suite of full-interactive graphical displays. The application does this is called 'GLIMPSE' (Graphical Look Interface for Malfunctions, Panels and the Simulator Environment). GLIMPSE is written using X11/Motif library with specialized graphics produced by the DataViews graphics library. It is able to display a wide variety of graphics: ranging from interactive process schematics to virtual control panel displays. An extensive library of graphic symbols has been developed and over 250 displays have been made so far for Bruce A simulator.

GLIMPSE is able to access simulators running either on the local workstation, or on another workstation across the network. It affords the user the ability to switch between any two simulators 'on-the-fly'. Its ability to do so is made possible by using the network standards of remote procedure calls (RPC) running on TCP/IP to acquire data for the displays. In addition, GLIMPSE is implemented using X11 graphics so that the program can be displayed remotely. This allows the simulator to be operated from any PC properly equipped with X terminal software such as XVision or eXceed.

GLIMPSE offers a user-customizable pull-down menu which can access displays, dialogs, or trigger commands on either local system or simulator host. For specific training applications, the menu might be tailored to only certain items. Alternatively, for maintenance work, a range of debugging applications would be available on the menu.

Automatic Scripting Facility

An extensive automatic scripting facility known as SIMMACRO has been implemented for RighTSTEP. This features a wide set of commands which can control the simulator, implement operator actions and simulator events, and trigger monitoring processes. SIMMACRO has conditional and branching instructions such as IF, TEST UNTIL and WHEN. It also has commands which allow control of the graphics displays for switching displays and popping up text and pointers. Furthermore, sounds and synthesized voice may be scripted in SIMMACRO.

Simulator events are automatically recorded as scripts when operating the full-scope or desktop simulator. For example, operator actions at the control panel will be recorded as will instructor-input malfunctions. These scripts can be further augmented to add looping, display control, and other advanced features.

The same scripts will operate on either the full-scope or desktop simulator. These scripts can be used to:

- assist and automate operation of the simulator for training use (as lesson plans)
- record and playback simulator runs
- debug the running simulator
- compose computer-based training exercises
- conduct automatic tests – for simulator validation for example
- through its display and analysis tools, aid in the engineering analysis of modelled systems
- perform regression testing on modelled systems

SIMMACRO is a very powerful facility which will continue to be expanded with ideas from users. Further improvements in capability will arise with planned data acquisition tools. We encourage users to freely exchange scripts developed for the benefit of all.

Evolution of the Project and Spin-Offs

Project Initiation

The project for which RighTSTEP was developed was the Bruce A Simulator Rehabilitation. This project began in 1993 and is currently in its commissioning stages at the Western Nuclear Training Centre.

It became clear in the late 1980s that the original simulation computer system would need replacement. Chief amongst the factors leading up to this conclusion was the heavy CPU loading. Secondary factors included a decreasing reliability and the vulnerability of an earlier inter-computer communication device, a so-called shared memory system. Problems related to obsolescence became noticeable in other simulator systems such as the interface computers and DCC display emulator. Although overall simulator reliability remained relatively high (98%) the unforeseen outages resulting from equipment failure were of particular concern in examination contexts.

The replacement of the Bruce A Digital Control Computers, Safety System Monitoring Computers, and addition of the Plant Display System planned under the station Bruce A Rehab programme provided the stimulus needed to initiate the project. The significant increase in I/O activity needed to support these systems was simply beyond the system capability and all expansion capability was already used. The decision was made not only to replace the simulation computers, but also to replace the 'simulator operating system' with a new design. One item which would not be replaced was the simulation models - by and large these 250 models of various systems had been refined and continuously improved since the simulator's inception in 1981 and represented an investment of several million dollars. It was highly desirable that these FORTRAN models run on the new system with as little change as possible: this was reinforced by the simulator trainers' high rating of its training capability. Thus, while the simulator operating system would be rewritten, the simulation models would not.

Platform Selection

A difficult evaluation of various computer platforms followed which focused on this problem of preserving the integrity of the FORTRAN models. Several special coding techniques had been used in these models to minimize CPU loading. Removing these coding tricks was difficult and undesirable. Compatibility for these coding techniques was highly dependant on processor architecture (e.g. RISC vs CISC).

It was also desirable that the new simulator be built around UNIX as the (vendor) operating system of choice. Several factors went into this decision, but two of note were that UNIX (as distinct from the original VMS) was seen as holding real potential for the future, and secondly was compatibility with the new Rehab add-ons noted above. "Open" programming standards were desirable to enhance portability, minimize costs and software development time. Windows NT was not yet a viable option at that time, and still lacks some critical elements for the task today. The platform selected was the DEC Alpha running Digital UNIX (then called OSF/1) principally on the strength of FORTRAN compatibility and CPU performance. Extensive checking ensured that almost all of the models would run unchanged on this platform. As it turns out, this platform has been an excellent all-round choice for many reasons.

Simulation Executive

The first key software task was to develop a minimal simulation executive, debugging tools and a core database engine which would provide access to global simulation variables. While this effort is the core and heart of any real-time simulator, it had very little visual impact to anyone but a specialist. After some months of effort we were pleased to show a few numbers changing on a text-based display.

Simulator Model and Data Porting

The next step was to port all the existing models and data to the new platform. Since the tools to maintain the simulator data were poor and would be lost with the old operating system, significant effort was made in porting the various simulation databases and importing it into a relational database. Examples of this simulation data are global simulation variables, simulator inputs and outputs, simulator malfunctions, station fuse and electrical bus dependencies, and control panel handswitch information. Since the 'garbage in, garbage out' adage applied here much effort has been expended in finding and correcting existing errors in this data.

Similarly, the simulation models must undergo extensive testing to ensure that simulator performance after the rehabilitation matches that before, which is in turn compared with station data where available.

Development of Display System and Graphical Tools

The third main element in the simulator is the user interface. The user interface of the old simulator was primarily text-based, and had very crude graphic capability and no system schematic displays at all. Preservation of the old user interface was not a fruitful option, and the decision to create a graphical user interface was fairly straightforward. To produce the corresponding interactive displays, not only would the displays have to be drawn, but some research had to be done on the models and control panels to determine what data was actually available to be displayed. Unlike the reference station, which has a limited number of definite and well-known instrumented points, on the simulator anything modelled is available for display.

The DataViews graphics library was employed as the graphics engine for the display system combined with a custom X/Motif front end and a TCP/IP based data acquisition mechanism. The graphics system has some limitations but offers the best compromise of features and flexibility to display development time. Over a period of a year or so, 150 process schematic displays were developed for various Bruce NGS A systems and 100 virtual control panel displays were developed. These displays are easily adaptable and customizable.

It is important to note that unlike a monitoring system, the simulator graphical displays are fully interactive. On schematics the simulator user may click on items to insert malfunctions or impose instructor controls. On virtual control panels the user may click on items to operate the device (such as turn a handswitch).

Desktop Simulator as Intermediate Design Step

While the main goal of the project was to replace the full-scope simulation computers, essentially at this point in the development cycle we had produced a desktop training simulator running the exact same software as the full-scope would. The special hardware interfaces, which would connect to the digital control computers and control panels, were still in development. The system ran with satisfactory real-time performance on a desktop workstation. While on the old simulator, development facilities were very limited, the availability of many small simulators provides several fundamental shifts in the way simulator work may be done:

- the development environment is very similar to the actual full-scope simulator environment
- the amount of computer power and usage time available to any developer is significantly greater
- some full-scope simulator training can be augmented or even replaced by usage of a desktop simulator in personal or classroom settings
- technical training of both licensed and regular staff may be augmented by usage of a desktop simulator
- engineering analysis, often aided by use of the full-scope simulator, could be assisted with use of the desktop simulator

Comparison of Desktop and Full-Scope

While simulators have become significantly more user-friendly from their early days, they remain a very complex computer task. An understanding of the capabilities of the desktop and full-scope simulators is necessary to fully benefit from them.

Unchanged Items

The vast majority of the software and data are unchanged between the full-scope and the desktop. In particular, the following items are all compatible:

- simulation system software
- data collection and analysis tools
- FORTRAN models
- simulation databases
- graphic displays
- automated lesson plans / macro scripts

The key benefit of this is that development on one system benefits the other. As such, engineering users will benefit from the ongoing maintenance of the full-scope simulator, and the full-scope simulator will benefit from engineering models and macro scripts developed by those users. This offers real benefit to all involved.

Users should be aware of fundamental limitations in all training simulators. Training simulators were designed from a training perspective. They are designed to simulate events with a level of accuracy which would be tolerable to control room operators. As such, they are not intended to compete with safety code as an analysis tool. While training simulators typically offer the widest simulation scope available, the degree of detail in this scope varies. For example, systems which have little control room 'visibility' typically are less detailed. Given an understanding of these limitations, excellent results can quickly be obtained from the full-scope training simulator for many analysis tasks. [From a quantitative viewpoint, prudence would dictate that alternative analysis should always be used to check results of impact.].

DCC Implementation

The digital control computer (DCC) implementation is a key difference in the two types of simulators. The full-scope simulator normally operates with near-replica hardware DCCs, running a station software (with minor modifications to meet simulator-specific needs). Normally it is not desirable to stimulate hardware such as station computers within a simulator environment. However, this does ensure a high degree of fidelity in operator interactions which are the basis of simulator training. It also offers a better degree of utility when DCC modifications are tested by design staff. The hardware DCCs are a very high cost item – the hardware is rare, specially engineered, and requires an intensive and specialized maintenance effort. Implementation of a hardware DCC for a desktop simulator essentially makes it a full-scope simulator without control panels – thus not generally affordable. Hardware DCCs generally do not allow simulator speedup or slowdown.

On the desktop simulator, functional models of the DCC control programs are used. These are based on program rules and are an interpretation of those rules (i.e. automated code regeneration was never an objective). Such models of control programs are far more flexible than hardware DCCs (the code can be easily changed) but lack fidelity for DCC engineering work. For example, program changes cannot be tested on a DCC model except as a general algorithm. Similarly, dual DCCs are not implemented nor is any simulation of disk or I/O subsystems. One key limitation is that no DCC-generated displays or annunciations are available at present. The modelled DCCs may be used on either the full-scope or the desktop and may be sped up or slowed down in synchronization with the rest of the simulation.

A third alternative has recently been developed which has been dubbed the virtual DCC. This is a Varian 72 essentially implemented in software. The virtual computer interprets opcodes, performs virtual I/O and produces displays just like the physical computer. The virtual DCC is currently being commissioned for unit 0 use on the full-scope simulator. A virtual version of the upgraded Second Source DCC computer is not yet available. When fully completed, the virtual DCC will offer some of the flexibility of the modelled DCCs while offering a more maintainable system than the hardware DCCs. This comes at a cost of considerable CPU time. A virtual Varian consumes one CPU on a multi-CPU 250MHz Alpha system. A slower system can be used at less than real time.

System Sizing Guidelines

Digital Equipment Corporation (DEC) has a wide range of computer systems capable of running RighTSTEP ranging from about 20K$ to 250k$ (and upward) inclusive of operating system software licenses . A full look at computer specification is beyond the scope of this document, but a brief look at general capabilities follows here. In each case, different levels of the DEC Alpha product line are being examined.

Alpha Server 4100, 4 CPU, 300MHz+, 1GB memory typical

- using multiple personal simulations:
    - support for a large number of users
    - support for a number of concurrent independent simulators each with graphics
    - large simulation speedup capability – up to about 25 times real time
    - a rich development environment for simulation models and excellent department server
    - ample I/O capacity for disk farms
- as a full-scope simulator:
    - ample expansion capability for high fidelity models
    - support for dual virtual DCCs
    - capacity for porting other station computer systems like SSMC
    - ample I/O capability for disk farm, stimulating station computers
    - expansion as needed through reflected memory I/O to additional Alpha computers

Alpha Server 4100, 2 CPU, 300MHz+, 256MB+ memory

- similar to above, with reduced capacity
- suitable as a engineering test bed, supporting virtual DCCs and full-scope models
- significant speedup capability depending on situation

Alpha Station 250 series, single CPU, 266MHz+, 96MB memory+

- suited as a single user workstation for engineering, classroom demonstrations, or technical training
- suitable for running a single simulation in real-time mode (two simulations may be run if 128MB of memory is installed) with some capacity for high-fidelity models.
- some speedup capability (approx 2 times) for single simulation.
- large simulations or virtual DCCs could be run at reduced speed or in non-real-time mode

Entry Level Alpha Stations (under 266MHz, 96MB memory)

- suitable as single user workstation with limitations
- may not be able to run new models at realtime, but will operate fine at slower speed
- for better performance should omit some peripheral simulation models
- very suited to partial-scope simulation

All levels of workstation fully support X-Windows so users may access the simulator via an X11 emulation package running on Windows 95/NT. This eliminates the need to be physically seated at the workstation and offers some benefits when integrating use with other Windows applications.

Early RighTSTEP Users

A number of users have adopted RighTSTEP already:

Western Nuclear Training Department:
Aside from the full-scope training simulator which is currently being commissioned, WNTD has used the desktop simulator in a classroom setting to conduct Secondary Control Panel Operator (SCPO) training using a computer projector. A further intent to use this set-up in technical training has been expressed as the full-scope simulator sometimes is used for this task when available.

Atomic Energy Control Board:
The Atomic Energy Control Board has purchased the RighTSTEP system for the primary purposes of training and engineering 'scoping'. One extra capability within the category of training which is of particular interest is the development of operator examination scenarios. Use of the desktop in this capacity benefits both Ontario Hydro and the AECB. Staff from the latter can reduce travel time to Bruce county for exam development and improve their own staff's understanding of station system in a very accessible environment. This will also relieve pressure on both the relatively costly simulator time and the time of operations staff supporting the preparation of examination scenarios.

Bruce NGS 'A' Engineering:
Bruce 'A' Engineering has installed a desktop simulator at the technical unit which has been used for familiarization with various station systems by engineering staff. Future plans are to expand the system into a comprehensive engineering test bed which might include station computer systems.

Reactor Safety Operations and Analysis Department:
RSOAD department has been evaluating RighTSTEP and has made a formal recommendation that it be adopted for the role of general technical training within the department. Charles Olive of RSOAD has been an invaluable help in testing the lesson plan facility and he developed the first lesson plan scripts which have already seen second use at WNTD for SCPO training.

Future Directions

Porting Other Plant Models

While Bruce 'A' models are now available, many potential users have inquired as to the prospect of other station models being ported. Each port constitutes the following items:

- porting simulator models
- porting simulator data
- developing DCC models
- developing schematics and virtual control panels

Of these tasks, the last three are fairly substantial. Darlington models and some data has already been ported to RighTSTEP by the Darlington simulator section. They are in the process of building on this foundation by developing DCC models and other necessary items for use as a development facility.

Bruce B simulator upgrade is planned to begin after Bruce A simulator Rehab is complete and early versions of Bruce B simulator will likely be available in 1999.

Improved Fidelity Models

With additional computing capacity available, an obvious area of improvement would be to upgrade selected process models to provide additional realism for training. The first of these is the new reactor model which is currently in work. This new reactor model an adaptation of the SMOKIN developed by RSOAD to the real-time environment of the simulator. As such, it has 19 modes fully cross-coupled and 15 delay groups. It is currently being commissioned. The model offers significantly better realism for local effects such as zone fill/drain and single shut-off rod effects and large effects like flux tilt.

Adaptation of the SMOKIN code is one instance in which we have received benefit of the desktop simulator environment for simulation development work. In comparison to reactor model upgrades done in the past (with analysis and test tools then available), the integration and commissioning of this upgrade indicates an overall saving in time of about 60%. This is wholly attributable to the current test environment - in essence a 'private copy' of the full-scope simulator - and the current generation of testing and analysis tools within GLIMPSE and SIMMACRO.

Other model fidelity improvements as also possible and will be driven by training demand.

Data Collection and Analysis Tools

While some data collection and analysis tools currently exist and are adequate for training usage – a more complete collection is very desirable for long-term maintenance of the simulator, with natural spin-offs for engineering use. Continued efforts will be made in this area. In particular, a integrated facility for data and event logging along with limit detection ability is scheduled to be developed later this year.

Currently, data export is available from the simulator graphing system to Microsoft Excel. We would anticipate expanding on this as needed.

Improved Display Composition Tools

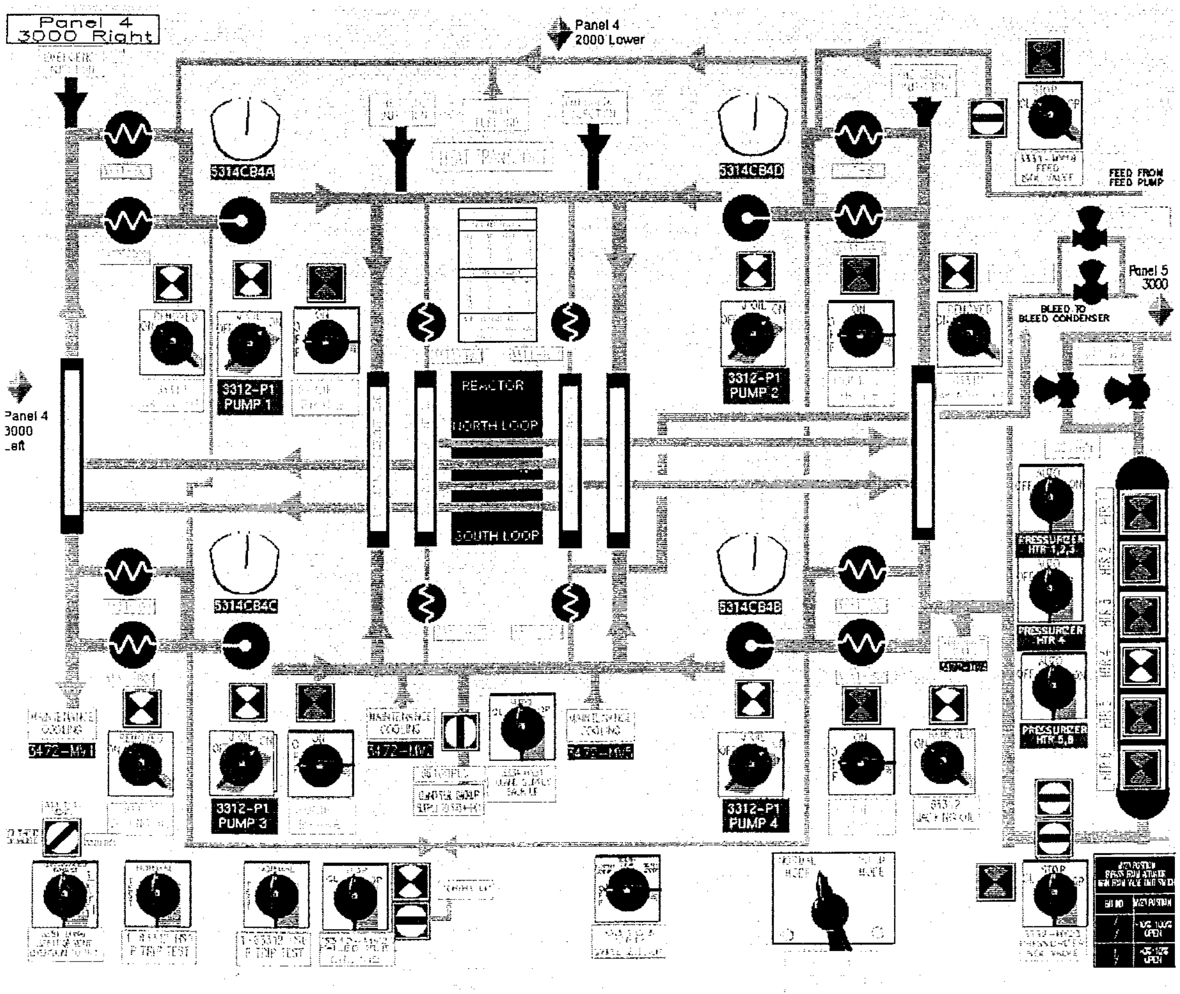
A major task in developing the simulator is the composition of the numerous graphic displays. The present drawing tool used to develop these is non-simulator specific. Developing a drawing tool which allows tighter integration with the simulator database and drawing library would greatly aide in the development of displays, and could definitely improve the schedule for the availability of Darlington and Bruce B models. Development of this tool will likely occur in early 1998.

Conclusion

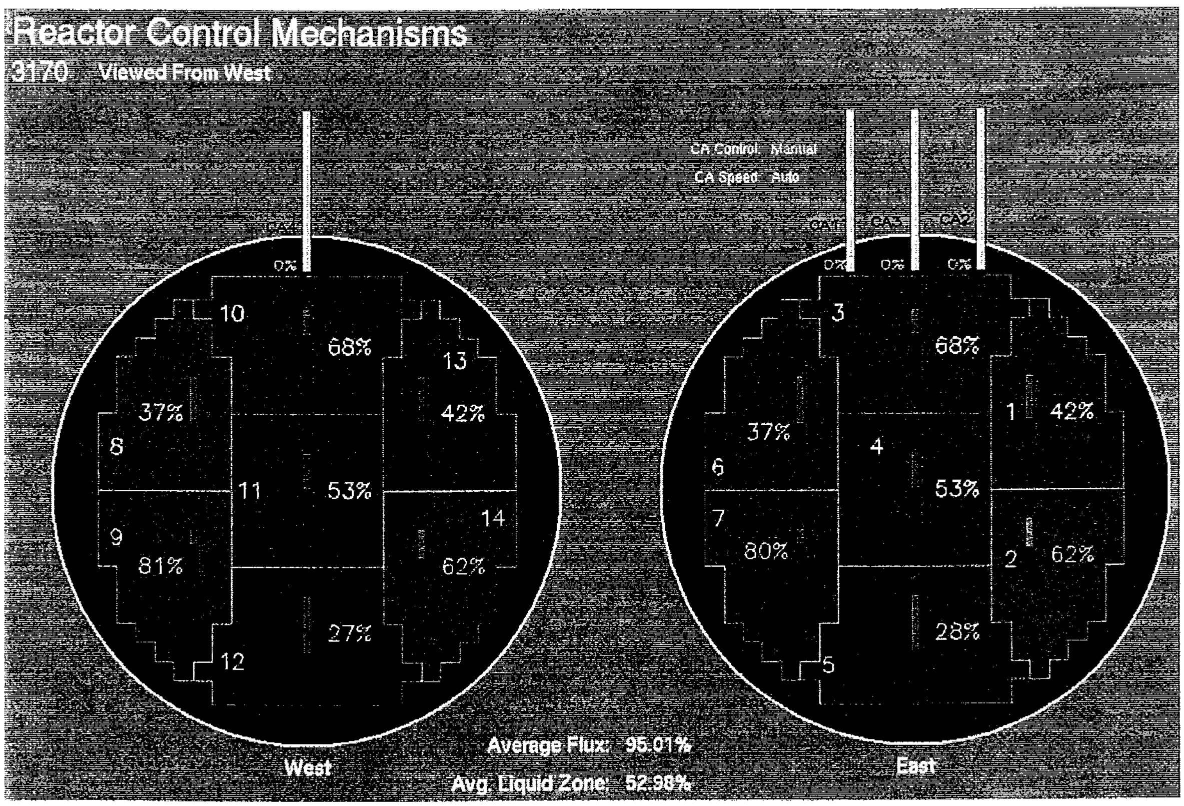
We have given an overview of the main functions and present usage of the RighTSTEP system. This system is currently being commissioned on the full-scope training simulator and is available in a desktop simulator form for training and engineering usage. This platform is planned for the future direction of training simulators in Ontario Hydro and we hope to expand on the system and provide economies of scale to all users.

We have found for our own purposes that this is the right technology for upgrading our own real-time simulators while preserving the substantial investment in legacy software. The desktop usage further leverages this investment by making it available to a many more potential trainees as well as engineering customers who could benefit from the use of the full-scope simulator.

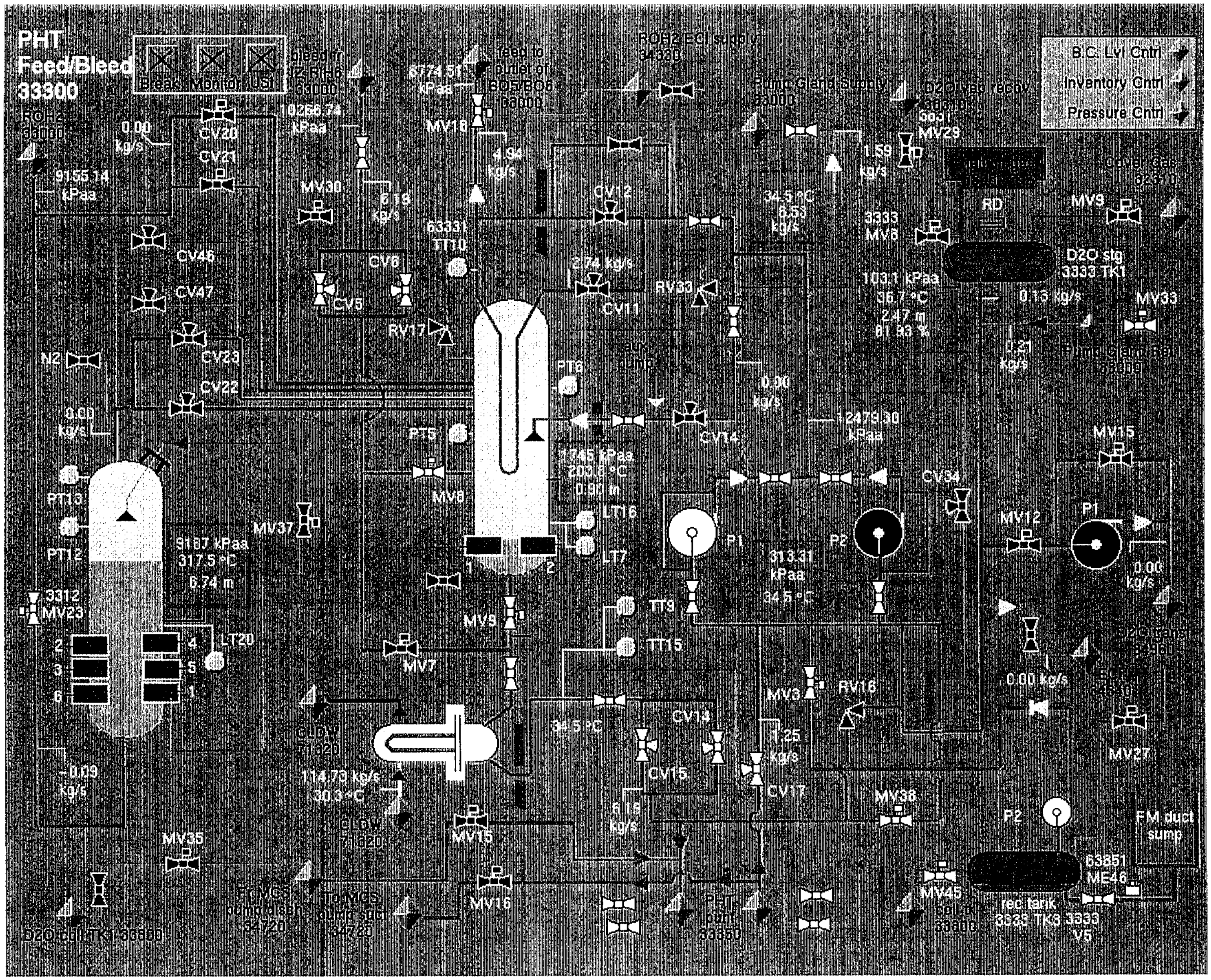
We have indicated future directions in which we plan on taking this technology. We hope to maximize development in this area which will benefit our customers the most as well as other users in Ontario Hydro.



Sample RighTSTEP virtual control panel - Bruce A PHT, lower section.



Sample RighTSTEP schematic showing zone powers.



Sample RighTSTEP schematic: Bruce A PHT Feed and Bleed System.