Implementation of Graphical User Interfaces in Nuclear Applications

Kjell Arne Barmsnes, Terje Johnsen, Carl-Victor Sundling
Control Room Systems Development Division
OECD Halden Reactor Project
N-1750 HALDEN, NORWAY

ABSTRACT

As part of the continuous work on improving safety of nuclear installations, new requirements to computer systems and information presentation have emerged. This has contributed to an increased interest and acceptance in introducing software systems in nuclear applications, and in particular in the control room of nuclear power plants. During the recent years there has also been a shift from proprietary hardware and software towards more open off-the-shelf hardware and software solutions. As a consequence of this there is a demand for systems that support design and implementation of Graphical User Interfaces (GUIs) in this environment.

Picasso-3 is a User Interface Management System (UIMS) supporting object oriented definition of GUIs in a distributed computing environment. In Picasso-3, GUIs are defined using an interactive editor; however GUI components can be created, modified, and deleted at run-time. The graphics editor is designed in a way that allows immediate testing to verify that the user interface works as intended. Attributes of GUI components can be dynamically connected to application data in order to reflect some state of an application. The dynamic connection and the behaviour of the interface is described by the user interface designer using the Picasso-3 language, pTALK. Reusable interface components, or objects, can be created interactively using the graphics editor. Objects can be put in libraries for easy retrieval and reuse.

The Picasso-3 system is designed to operate in a network environment, and offers a high-level, powerful Application Programmer's Interface (API) for easy integration with user programs. The Picasso-3 system is based on industry standards such as C/C++, TCP/IP, and X-Windows, and is available on UNIX platforms and Windows NT.

The Picasso-3 system is currently being used in a number of different application areas within the nuclear industry, such as retrofitting of display systems in simulators and control rooms, user interfaces of simulators, visualization of data generated by analysis tools, education and training applications, and in human factors research. This paper will give some examples of nuclear applications where the Picasso-3 system has been used.

INTRODUCTION

The OECD Halden Reactor Project which carries out an international research and development programme in the man-machine systems area, sponsored by organisations in 19 countries, has for more than 20 years been engaged in the development of user interface systems.

Graphical presentation in computer programs (applications) has become more and more sophisticated and advanced throughout the last years. The Picasso-3 system is a flexible and powerful tool offering system developers an aid in developing and testing of graphical interfaces. Furthermore it is an execution environment for applications. It is the third generation of UIMS developed as part of the OECD Halden Reactor Project since 1984.

Main goals in developing the system have been to:

- enhance the quality of graphical user interfaces of applications
- make communication and integration with other systems easier
- reduce the development and maintenance costs throughout an application's lifetime

THE PICASSO-3 SYSTEM

The Picasso-3 system offers an interactive environment for definition and execution of high quality graphics. The user interface is specified using an interactive *Graphics Editor* (GED). GED has two modes of operation; edit-mode and test-mode. Edit-mode is for drawing operations and specification of dynamics and user interaction. Test-mode is for on-

line testing dynamics and user interface behaviour. Switching between the two modes is done by selecting from a menu in GED.

The integrated design/test environment makes Picasso-3 well suitable as a prototyping tool and leads to an evolutionary development process. Even though savings in this phase of a software's lifecycle can be considerable, the real potential for savings is found in the maintenance phase, when the software is installed and is used for its aimed purpose. Errors are found and corrected, but the main work for the developers is to improve and enhance the functionality.

Most of the changes in functionality involve presenting information that is already stored in the system in alternative ways. A way of reducing maintenance costs is to separate the user interface from the application code and information storage. This way, a change in the interface would result in minimal changes in an application's code. A major requirement to the Picasso-3 system is to offer a complete separation of the user interface from the application code.

The Picasso-3 system is available on several UNIX platforms as well as Windows NT. The API utilizes the TCP/IP communication protocol, and can be utilized by any C/C++ callable program.

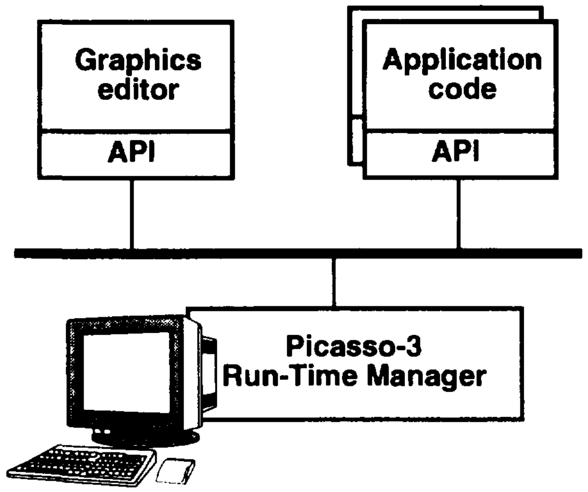

Figure 1. The end-user interacts with the user interface realised by the RTM. Application programs, communicating through the API library, can be distributed in the network. The GED is itself an ordinary Picasso-3 application utilizing the API.

Picasso-3 operates in a distributed computing environment. This means that the system is able to cooperate with other running tasks and support a client-server architecture. The central part of the system is the *Run Time Manager* (RTM), realising the user interface. Communication is carried out through a set of high-level interface routines that is linked with every program. These API routines give application programs access to all functionality in the RTM.

Dynamic Graphics

In order to create dynamic graphics, Picasso-3 offers functionality for connecting graphics to application data. The dynamic connections are specified in Picasso-3's user interface language, pTALK. pTALK is a C/C++ like language with additional constructs supporting graphic manipulation. The language is compiled at run-time, allowing source code to be added and executed on-line during the design as well as during the execution of the interface.

To connect the foreground colour of a polygon to an application variable would typically involve the following:

- create a function that returns a colour given an application variable

```
int statusCol( float status )
{
   if ( status > 0 )
      return Red;
   else
      return Green;
}
```

- connect the foreground colour attribute of the polygon to the application variable **'CompStatus'**

```
foregroundColour = 'statusCol( CompStatus )';
```

In this way any attribute of a predefined or user defined graphic shape can be assigned a dynamic expression.

As editing functionality is an integrated part of the Picasso-3 system, graphical objects can be manipulated at run-time. This can be used in applications to represent a dynamic data structure graphically. Utilizing standard API and pTALK functions the application programmer can add, modify, and remove graphics on-line.

User Interaction

An important part of a user interface is the interaction with the end user. In Picasso-3 the interaction is specified in dialogues that are integrated with the graphics elements themselves. Dialogues can be defined on different levels in the interface. User interfaces developed in Picasso-3 consists of three levels: from top to bottom, the application level, the picture level, and the element level. This hierarchical structure allows the user interface designer to create local as well as more global dialogues.

Figure 2. The user interface designer works in the interactive environment provided by the Picasso-3 Graphics Editor (GED). Attributes, functions and dialogues in the picture, or in selected objects in the picture, can be viewed or changed. Switching between design mode and test mode is done simply by pressing a push button. In test mode, the interface looks and feels exactly as it will do on the end user's screen. Graphics attributes can easily be changed by selecting colours, fonts, etc. from a graphics attribute window.

A dialogue consists of an event-action pair. The event can be a combination of predefined event specific pTALK functions. Examples of event functions are

```
buttonPressed( Left )
cursorMoved()
windowEntered()
shapeEntered()
shapeLeft()
```

The action part of a dialogue is a pTALK statement, and thus can contain calls to standard functions and user defined functions. Since both the event and action descriptions are user defined, highly specialised dialogues can easily be developed utilising the pTALK language.

Interface Components

In designing user interfaces it is important to be able to re-use interface components across related applications. In Picasso-3 this is provided through the use of *classes*. A Picasso-3 class is a collection of attributes, functions, dialogues, and graphics, and can be viewed as an independent user interface entity.

Once a class is defined it can be instantiated in a number of *instances*. An instance is a copy of its originating class and can be configured through its attributes.

Easy retrieval and re-use of classes is accomplished by grouping related classes in *libraries*. In addition to class definitions, a library can contain definitions of all graphic resources needed in a class. These resources include colours,

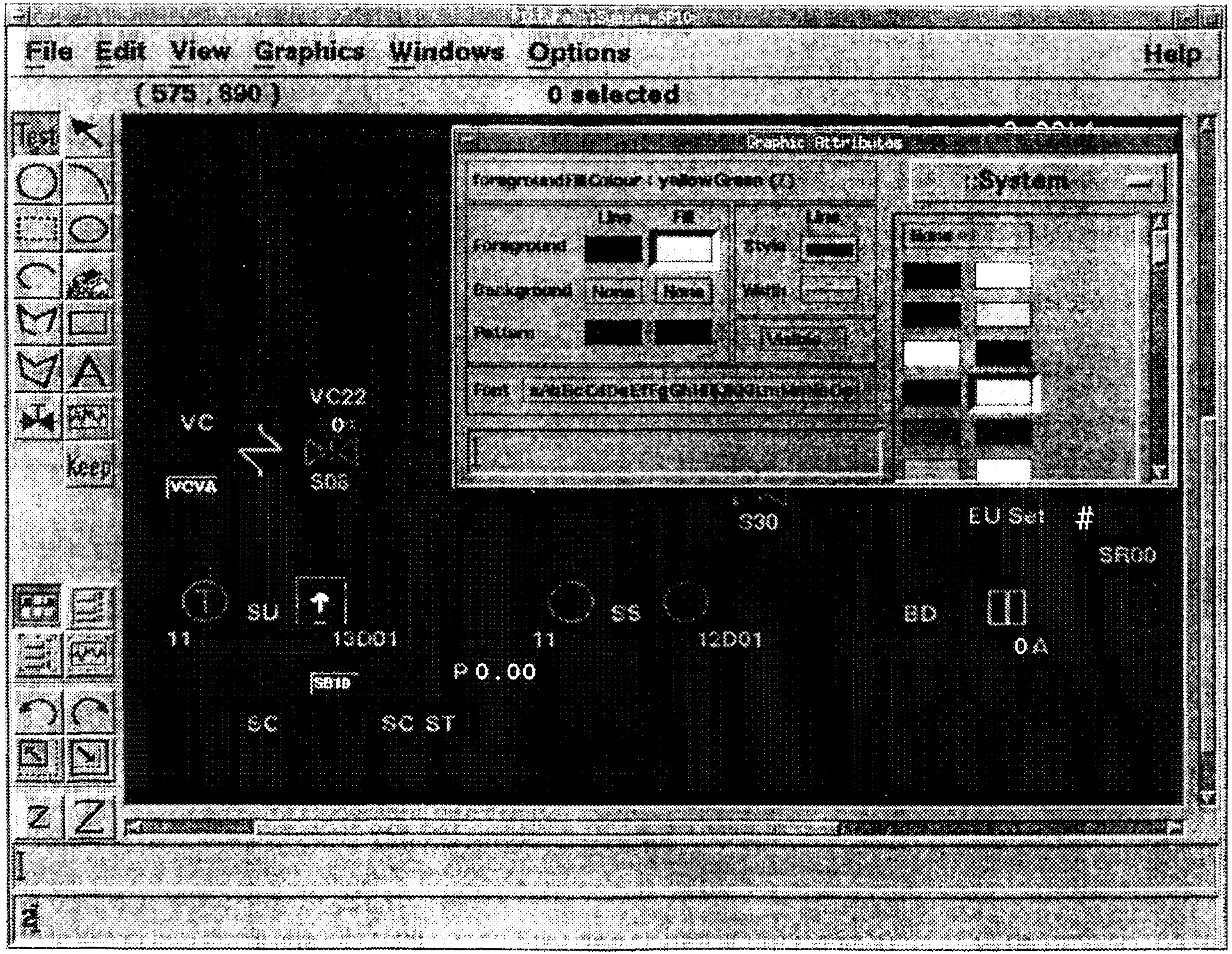

Figure 2. The user interface designer works in the interactive environment provided by the Picasso-3 Graphics Editor (GED). Attributes, functions and dialogues in the picture, or in selected objects in the picture, can be viewed or changed. Switching between design mode and test mode is done simply by pressing a push button. In test mode, the interface looks and feels exactly as it will do on the end user's screen. Graphics attributes can easily be changed by selecting colours, fonts, etc. from a graphics attribute window.

fonts, patterns, functions, and attributes. A library of classes and resources can thus be defined as an independent part of the user interface and can easily be re-used in a number of applications.

Application Programs

In Picasso-3 an application program is defined as an external program (C, C++, Fortran, etc.), that feeds the user interface with data. The data is typically used in the user interface to visualize some state of the application.

The application data is made available to the Picasso-3 system using the API. The API is a library of high-level C/C++ functions that is linked with the application program. The API is built on top of a communication system that supports communication between applications in a network. This allows Picasso-3 applications to be distributed in a network environment and run seamlessly on a number of computers.

A number of API functions are available for transferring data, calling functions, as well as drawing graphics in the Picasso-3 RTM. The API also provides a mechanism for declaring application functions to the RTM, allowing application functions to be called from pTALK.

Often the application containing the data (process computer, simulator, etc.) runs on a proprietary hardware and software system, making it difficult to interface with the data in a standard way, using a standard communication protocol. A solution is to put a small gateway program between the two systems, a program that will translate between the two protocols. The gateway program will typically include the two system's different API libraries.

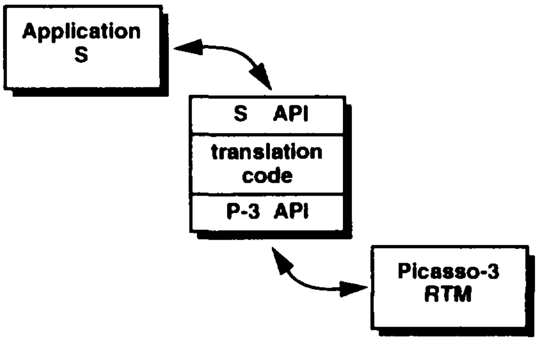

Figure 3. Communication between two applications with their own protocols. A gateway process is used to translate between the protocols. In this example, Picasso-3 is connected to another system S.

USER INTERFACES IN NUCLEAR APPLICATIONS

During the recent years the use of software systems in nuclear applications has become more common. This is particularly true for systems connected to the operation of nuclear power plants, such as display systems in a control room.

The requirements to graphical user interface systems in nuclear applications are not much different to those of other process control applications. However, in nuclear applications there are typically no process control actions initiated from the graphical user interface. All control actions to the process are carried out through conventional hardware controllers, and the graphical user interfaces are only used to present data from the process.

Graphics

In process control applications there is often a need to present the development of process parameters over time, what is referred to as historic trends. The Picasso-3 system has a built in trend log and display system. The trend log takes care of storing the actual historic parameter values. The historic trend data can be stored in memory or to disk. The trend log can be an integral part of a Picasso-3 RTM, or an RTM can be configured as a dedicated trend log server RTM itself. A dedicated trend log RTM is typically used in the following cases:

- when large amounts of parameters need to be logged a dedicated trend log RTM can run on a separate workstation to ensure high performance.
- when several display RTMs distributed in a network need synchronized trend data a dedicated trend log RTM can be used to provide one central trend log data source.

The trend display system provides the graphic presentation of the logged data, and can be configured to present trend data for several parameters in a diagram. A trend diagram is highly configurable and can be used to indicate state information of parameters as well as the actual parameter value.

User interfaces of process control applications typically contain many process pictures consisting of large amounts of process components. In these types of applications it is important to have some concept of graphic classes in order to efficiently develop and maintain the user interface.

The complexity of a process component can be encapsulated in a class, and the class can be designed to be easy to use. In order to be able to express complex user interface behaviour, it is important that the user interface system contains a mechanism that allows the user to configure the behaviour of the user interface. In Picasso-3 this is provided by the class concept and the pTALK language.

It is also important that the instances (the actual usage of a class in the user interface) of graphic classes are tightly coupled to the originating class. This should ensure that any modification of a class is reflected in all its instances, and thus making the maintenance of the user interface easier.

System Architecture

Two characteristics of process control applications are that the data source often resides on proprietary hardware and software, and that large amounts of data must be handled in a timely manner.

Proprietary hardware and software architectures often require tailor-made communication solution in order to transfer data to programs running on other architectures. This requires that the user interface system provides a flexible way of interfacing with other systems. In Picasso-3 this is realized through the API. The high-level communication functions of the API allows the system designer to easily interface the Picasso-3 system to other applications through a tailor-made communication gateway.

Even though the Picasso-3 API is a set of high-level communication functions, the API itself and the underlying communication system have been designed to handle large amounts of data in a timely manner.

EXAMPLES OF PICASSO-3 APPLICATIONS

The Picasso-3 system has been used in a variety of applications within the nuclear industry. Examples of application areas are

- retrofitting of display systems in simulators and control rooms
- user interfaces of simulators
- visualization of data generated by analysis tools
- education and training applications
- human factors research

In the following a brief description of some Picasso-3 applications within the nuclear industry will be given.

SCORPIO

SCORPIO [1] is a core surveillance system that has two parallel modes of operation: the Core Follow Mode and the Predictive Mode. The main motivation behind the development of SCORPIO is to make a practical tool for reactor operators which can increase the quality and quantity of information presented on core status and dynamic behaviour. This can first of all improve plant safety as undesired core conditions are detected and prevented. Secondly, more flexible and efficient plant operation is made possible. These improvements are obtained by better surveillance of core instrumentation and through detailed calculations of core behaviour using on-line simulators.

The VVER version of SCORPIO uses Picasso-3 as its user interface system. The Picasso-3 RTMs are connected to a communication gateway (supervisor) that takes care of the communication with the SCORPIO modules.

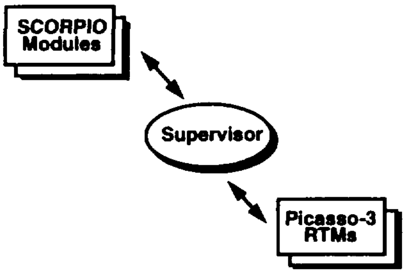

Figure 4. The Picasso-3 system communicates with the SCORPIO modules through a communication gateway called the supervisor. The supervisor takes care of data transfer between the two systems and also prioritizes the communication between the SCORPIO modules and Picasso-3.

The SCORPIO system is operated entirely from the Picasso-3 user interface. The supervisor is responsible for creating all process variables in the RTMs, and to transfer variable values to the correct SCORPIO module.

The supervisor also prioritizes the communication between the RTMs and the SCORPIO modules in case both operator RTMs and physicist RTMs are running. All communication to the operator RTMs has priority.

The predictive mode of SCORPIO utilises the advanced trend system functionality of Picasso-3 and allows combination of predictive and historic trend curves to be able to compare the prediction with actual parameter values.

NEWS

The NEWS (Nuclear Engineering Workstation Simulator) [2] system has been implemented by the U.S. Nuclear Regulatory Commission, Technical Training Division, USA. The user interface of NEWS has been created using

Picasso-3, and contains graphic process pictures for the following nuclear simulators:

- Babcock and Wilcox
- Combustion Engineering
- General Electric BWR 4
- General Electric BWR 6
- Westinghouse

NEWS is a training tool for the display of information that is not available on the full-scope simulator benchboard, and for showing physical phenomena in such a way as to add to the conceptualization of system interactions. NEWS is also being used to independently demonstrate the operation of control systems, component logics, and flow processes.

The system can operate in on-line mode connected to the simulators and in off-line mode with built-in logics. The off-line mode utilises the flexibility of the pTALK language, and contains complete descriptions of component logics and some control systems.

Simple models of some control systems have been developed in pTALK as part of the graphical interface. These models are typically run in off-line mode.

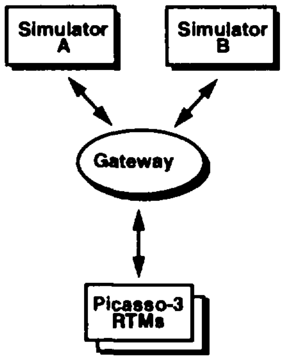

Figure 5. In NEWS, the Picasso-3 system is connected to different simulators through a communication gateway. The simulators are running on different hardware and the gateway translates between the different communication standards.

The NEWS user interface contains approximately 200 Picasso-3 pictures, and a majority of these have been implemented by instructors who typically are non-programmers.

Halden Man-Machine Laboratory (HAMMLAB)

The HAMMLAB facility comprises the NORS large-scale PWR simulator, the ISACS-1 [3] system, the associated experimental control room set-up, the experimenter's gallery/control center from where the man-machine experiments are conducted, and the basic HAMMLAB hardware/software structure.

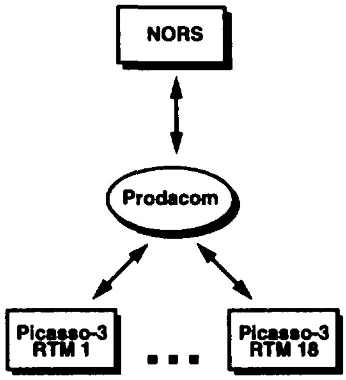

Figure 6. The user interface of HAMMLAB is implemented in Picasso-3 and consists of 18 workstations. The Prodacom gateway takes care of communication between the UNIX workstations and the Norsk Data mainframe on which the NORS simulator runs.

The user interface of the HAMMLAB facility is implemented in Picasso-3, and consists of approximately 250 pictures. The display system consists of 18 graphic workstations, some of which have fixed overview displays. The user interface contains process diagrams, dedicated alarm pictures, and configurable historic trend pictures.

The operation of the NORS simulator is carried out through the Picasso-3 interface and includes logging of all operator actions.

The HAMMLAB facility is used to conduct a number of human factors experiments and therefore requires a flexible user interface system that allows rapid changes of the interface to accommodate specific needs.

MELCOR

The Melcor program is a system for calculation of severe accidents in nuclear power plants. The output from this calculation is presented to the users as large documents with lots of numbers and x-y diagrams. It was left for real experts to interpret the results of the calculations. To improve the user interface of Melcor calculations, KEMA Nederland B.V decided to use Picasso-3 in a test case for visualizing the results graphically so that the results could be more useful and easily accessible for the users. The Melcor

Picasso-3 link was developed in a cooperation between the

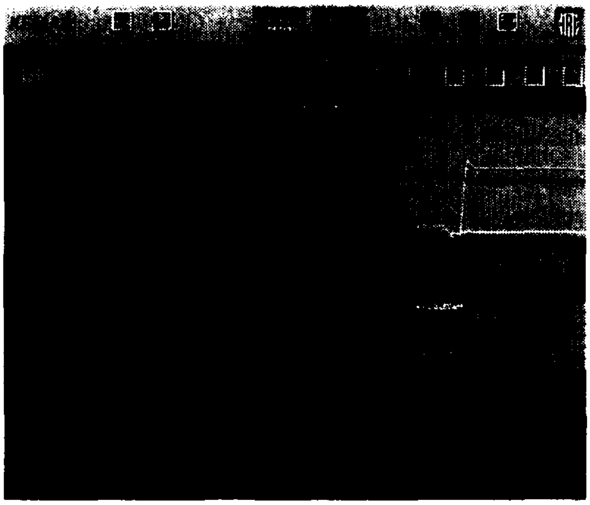

Figure 7. The figure shows a snapshot of water levels on the Melcor calculation. The colour of the water indicates the temperature according to the scales in the upper left corner. Water flow direction is indicated as arrows. Trend curves of water level plotted against time stamps are displayed in the upper right corner. A "play" pop-up window is used to select start time and step size of the calculations.

Halden Reactor Project and KEMA.

The first task that had to be carried out in order to visualize the Melcor code was to write a C++ application reading the Melcor plot file. This file is not generated in real time, so a direct connection is not possible. The format of this file is binary, and contains defined time steps with corresponding results. This program (MPROG) is linked with the Picasso-3 API. When reading the plot file, MPROG creates all necessary variables in Picasso-3 through the API.

The visualization of the Melcor plot file is carried out using Picasso-3. For the Vessel and Containment the following characteristics can be visualized as trend curves:

- Water level
- Water temperature
- Gas temperature
- Pressure
- Flow

All trend curves can be toggled in visibility and panning in time and dynamic setting of trend limits can be individually set. Another feature of the system is a graphical representation of core temperatures and distribution of selected core materials.

CONCLUSION

The Picasso-3 system is a flexible user interface management system that operates in a distributed computing environment. The system supports object oriented definition of graphical user interfaces. The powerful and flexible API allows interfacing to a wide range of existing applications. Picasso-3 is available on several UNIX platforms and Windows NT.

The system has been successfully used in several projects within the process control area such as retrofitting of display systems, simulator interfaces, training applications, and human factors research.

REFERENCES

1. K. Zalesky, J. Svarny, L Novak, J. Rosol, A. Hornæs, "SCORPIO-VVER Core Surveillance System", ENS Topical Meeting on I&C of VVER, Prague, April 21-24, 1997.
2. J.I. Griffin, L.F. Donatell, D.T. Nelson, G.O. Schneider, K.A. Barmsnes, "The Evolution of the Nuclear Engineering Workstation Simulator -- An Update", XI Annual Simulators Conference, San Diego, California, April 10-14, 1994.
3. N.T. Førdestrømmen, K. Haugset, "ISACS-1: Prototype of a Futuristic Integrated Cockpit Control Room", NEA/IAEA International Symposium on Nuclear Power Plant Instrumentation and Control, Tokyo, May 18-22, 1992