

Compact Nuclear Simulator 의 성능향상을 위한 그래픽 시스템 재설계

Re-design of Graphics System for Performance Upgrade in Compact Nuclear Simulator

차 경호, 권 기준, 송 순자, 박 재창

한국원자력연구소

대전광역시 유성구 덕진동 150

요 약

국내 원자력 산업 요원의 양성 및 개발도상국 원자력 요원의 훈련을 목적으로 활용되어온 Compact Nuclear Simulator(CNS)는 원자력발전소에서 일어나는 대부분의 계통에 대한 현상을 모의화한 CNS 모의코드를 Micro-VAX II 컴퓨터에 탑재하여 이용되어 왔다. 그러나, 최근 CNS 의 모의 호스트 컴퓨터(simulation host computer), 그래픽 서브 시스템, 그리고 제어 판넬의 인터페이스(Hard-wired) 등이 노후화되어 시뮬레이터로서의 기능을 상실하였다. 이로 인하여 CNS 구성 요소들을 교체 및 성능 개선하고, VMS 에서 운용되던 발전소 모의 프로그램은 HP-UX 10.2 로 이식하였다. 그래픽 시스템은 기존의 그래픽 하드웨어의 교체에 따라 컴퓨터-기반 그래픽 인터페이스로 다시 설계되었다. 한편, HP-UX 10.2 에서 Picasso-3 를 이용하여 정적, 동적 그래픽 디스플레이를 개발 중이다.

Abstract

This paper describes re-design of the graphic subsystem in Compact Nuclear Simulator(CNS). CNS, which has been utilized with the aim of education and training the personnel from both domestic nuclear industry and developing country at Nuclear Training Center in KAERI(Korea Atomic Energy Research Institute), often causes erroneous functions because of its old-fashioned equipment. The graphic subsystem is also old-fashioned and causes hardware errors. Therefore, it is also being upgraded as a computer-based one with high performance. The electronic equipment is replaced with computer hardware while the graphic software is re-designed with Picasso-3 system. Picasso-3 enables trend curves, axial curves, P-T graphs, various diagrams, and menus to be designed as graphical interfaces on distributed computing environments. Currently, the dynamic interfaces are being interfaced and tested with the CNS simulation program.

I. 서론

CNS는 크게 나누어 하드웨어와 소프트웨어로 구성된다. 하드웨어는 주제어실의 제어반을 축소한 operator console, 실제 발전소와 유사한 상황을 모의화 하는데 필요한 모든 데이터를 발생시키는 모의 호스트 컴퓨터(simulation host computer), 모의 호스트 컴퓨터와 console 사이의 입출력 정보를 전달하는 interface 카드, 컬러 CRT에 표시되는 full graphic display 시스템, 훈련원에게 각종 명령을 내리고 훈련원의 훈련 내용을 감시할 수 있는 강사 조작반으로 구성되어 있다. 이들 구성 요소들이 노후화되어 시뮬레이터로서의 기능을 상실하게 되었다. 이에 따라 CNS의 모의 호스트 컴퓨터(Micro VAX II/VMS)를 HP-UX 10.2를 운영체제를 탑재한 HP Visualine으로 교체하고 모의 프로그램은 모두 HP Visualine으로 이식된다. 통신 인터페이스 카드들은 PLC를 이용하여 TCP/IP 프로토콜을 이용하여 네트워크를 구성한다. 그리고, 그래픽 시스템은 HP Visualine/HP-UX 10.2 UNIX 워크스테이션과 3대의 X-터미널을 하드웨어로 플랫폼으로 하고, Picasso-3 [2,3,4]를 Client/Server형 소프트웨어 플랫폼으로 하여 그래픽 디스플레이 개발 환경을 이용한다. 아울러, 기존 CNS에서 제공하지 못하던 Process & Instrumentation Diagram (P&ID)를 그래픽 시스템의 기능으로 설계한다.

II. CNS 그래픽 시스템 성능향상

2.1 기존 CNS의 그래픽 시스템

기존의 CNS를 위한 그래픽 시스템은 그림 1과 같이 3개의 동일한 디스플레이 채널로 구성되어 있고, 각 채널은 1 Mbyte의 내부 메모리를 가진 CPU 카드, 각 1 개씩의 디스크 제어 카드, 그래픽 디스플레이 제어 카드, 통신 카드, 그리고 19" 컬러 모니터로 이루어져 있다. 또한 3개의 채널이 공유하는 각 1 개씩의 floppy disk drive(FDD)와 hard disk drive(HDD), hard copy unit로 구성되어 있다. 각 디스플레이 채널은 2개의 버스 시스템으로 구성되어 있으며, 그 중 1개의 버스는 HDD와 통신하기 위한 것이고 다른 1개의 버스는 모의 호스트 컴퓨터인 Micro VAX II 컴퓨터와 통신하기 위한 것이다. 또한 그래픽 채널의 한 채널에는 FDD가 연결되어 있어 backup이나 update된 소프트웨어의 설치에 사용된다. 리모콘에 의한 제어는 모의시스템 수행 시간 스케일 선택(1-5), 각종 diagram 및 KAERI LOGO(11-15), 각종 axial curve(21-25), trend curve(31-43), trend curve의 시간 스케일(50-53), 화면 저장용 utility(60-73), 모니터 #1의 화면을 플로터에 출력하는 명령(97), 그리고 모든 모니터를 reset하는 명령(99)으로 구성되어 있다. 기타 비워둔 코드는 추후 첨가를 위한 여분이다. 한편, 모의 호스트 컴퓨터와 그래픽 시스템과의 통신은 매 200 msec마다 Micro VAX는 각 모니터에 어떤 화면이 표시되어 있으며, 어떠한 명령이 유용한지를 검색하여 각각 8 Kbyte 용량의 RAM이 있는 통신 카드를 통해 데이터를 전송한다. Micro VAX가 데이터를 전송시 busy flag가 active되고 디스플레이 CPU가 디스플레이 루프를 끝마칠 때마다 이 busy flag를 검색하여 inactive이면 데이터를 읽어 들인다.

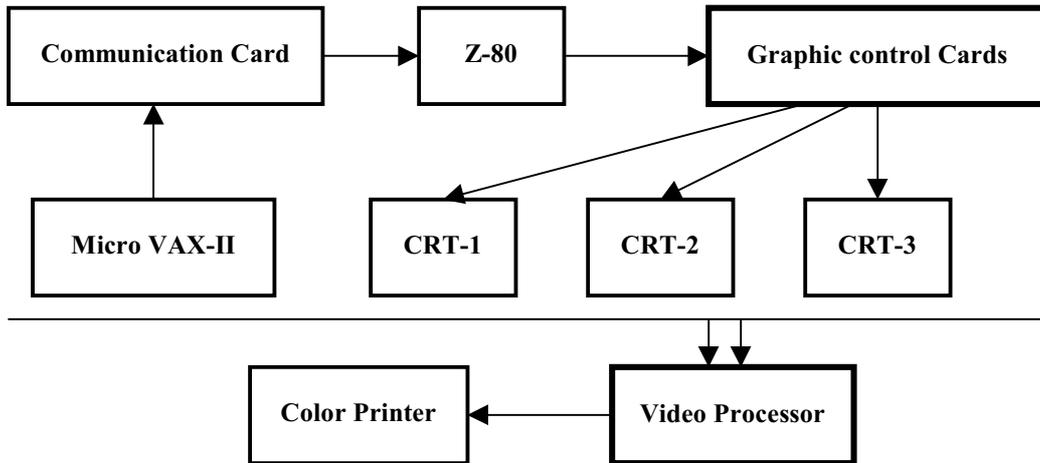


그림 1. CNS 그래픽 시스템의 개략도

2.2 Picasso-3 시스템

OECD Halden Reactor Project 에서 원자력 산업에 이용할 목적으로 개발되어온 사용자 인터페이스 관리 시스템(User Interface Management System: UIMS)인 Picasso-3 를 그래픽 응용을 개발할 수 있는 잇점으로 인해 선정하였으며, Picasso-3 시스템의 개략적인 구조는 그림 2 와 같다.

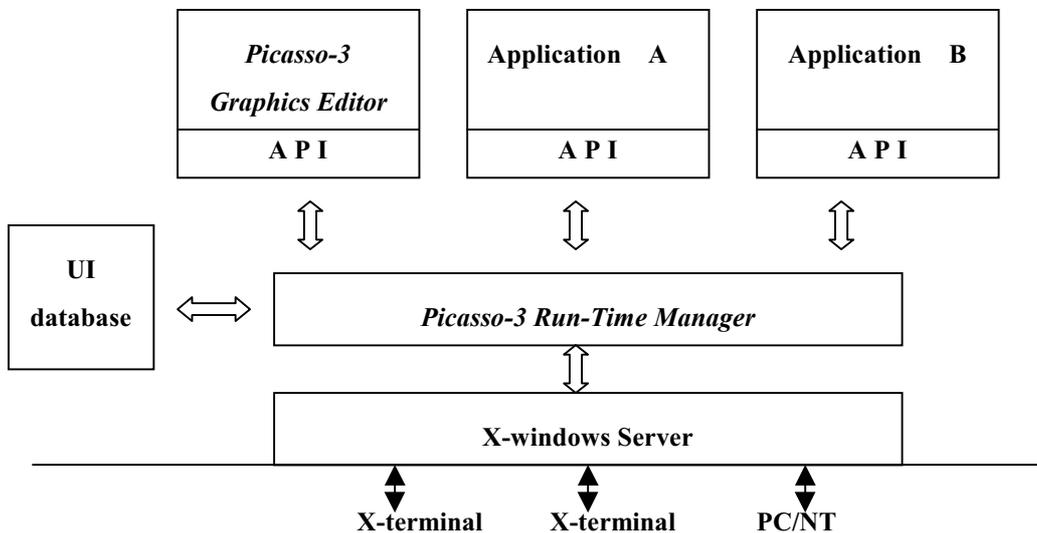


그림 2. Picasso-3 시스템

CNS 그래픽 시스템의 개발에 이용된 Picasso-3 는 분산 컴퓨팅 환경(distributed computing environment)에서 그래픽 사용자 인터페이스(Graphical User Interface: GUI)를 개발하기 위해 재사용(reusable)이 가능한 객체지향(object-oriented) GUI 요소(components or objects)를 대화형

(interactive)으로 생성, 수정, 라이브러리가 가능한 프로그래밍을 지원한다[2,3,4]. Picasso-3 는 컴퓨터 모의 도구(computer simulation tools), 전산화된 운전 지원 시스템(Computerized Operator Support System: COSS), 사고 관리 및 통신망 관리 시스템(accident management and network management system), 그리고 전문가 시스템의 GUI 개발에 이용될 수 있다[5].

2.3 Picasso-3 를 이용한 CNS 그래픽 시스템의 재설계

GUI 개발 도구인 Picasso-3 를 이용한 CNS 그래픽 시스템의 재설계는 기존 CNS 그래픽 시스템의 기능을 Picasso-3 환경에서 구현하는 것과, P&ID 를 추가하는 것으로 진행된다. 그리고, P-T 그래프, trend 그래프, P&ID 가 각각 3 개의 X-터미널을 통해 모니터 된다. 리모콘으로 선택하던 디스플레이는 메뉴 방식으로 선택하도록 설계된다. Picasso-3 를 이용한 CNS 그래픽 시스템의 재설계에는 다음과 같은 Picasso-3 기능이 이용되었다.

Application - Picasso-3 에서 Application 은 UI 구조의 최상위 계층 레벨로서, CNS 그래픽 시스템의 재설계에서는 기존 리모콘에 의해 선택 가능한 디스플레이(1-99)와 P&ID 가 각각 Application 으로 정의된다.

Picture - X- 윈도우 스크린 상에서 이동, 크기, 아이콘, 표시가 가능한 application 을 구성하는 UI element 로써, 하나의 application 은 1 개 이상의 Picture 들로 구성된다. CNS 그래픽 시스템의 재설계에서는 diagram, axial curve, trend curve 가 picture 로써 정의되고 있다.

Shape - Picture 를 구성하는 graphic element 로써, line, circle, rectangle, text 등이 있다. 그리고, 2 개 이상의 graphic element 들을 이용하여 더 복잡한 shape 을 정의할 수 있다.

III. Picasso-3 를 이용한 CNS 의 GUI 구현

Picasso-3 는 객체 지향 프로그래밍 언어인 C++로 개발되었으며, 객체 지향 개념의 특징인 클래스(class), 정보 은닉(encapsulation), 그리고 상속(inheritance)이 가능하다. 이러한 특징을 GUI 개발에도 적용되어 GUI 개발에서의 생산성을 높일 수 있다.

3.1 Applications

Application "CNS_GUI"에서는 variable 및 data structure 들을 "database" statement 로 아래와 같이 정의한다.

```
database = "CNS_GUI.pdat";
```

"CNS_GUI.pdat" 파일에는 CNS 모의 프로그램으로부터 공유 데이터를 읽어오기 위한 변수, trend log 및 시간 변수, 그리고 structure 등 global variable 및 data structure 가 정의된다. Application 의 Picture 는 "Graphics Editor(GED)"에 의해 대화식으로 정의될 수도 있고

("filename.ppic"), text editor 를 이용하여 작성("filename.Tdoc")하여 "pcc" interpreter 를 사용하여 "filename.ppic" 파일을 생성할 수도 있다. 예를들어, application CNS_GUI 에서 main menu 에 해당하는 picture 를 screen 에 나타내기 위해서는 다음의 "displayPicture()" statement 를 사용한다. 여기서, "mainWin"은 screen 의 window 를 정의한다.

```
displayPicture("mainWin", "mainPicture.ppic");
```

한편, interaction 을 위한 "dialogue"에는 event 와 action 들이 정의되는데, main menu 에서 menu 에 의해 선택한 window 를 나타내는 "Next" class 의 dialogue 를 다음과 같이 정의하였다.

```
dialogue  
{  
  event = `buttonPressed(1)`;  
  action = statement `{  
    if (strlen (ExecuteString) > 0) execute (ExecuteString);  
  };
```

Picasso-3 에서는 parameter 의 trend 를 정의하여 사용할 수 있도록 "trendLog"를 제공한다. 다음 예는 Application CNS_GUI 에서 사용된 trend curve 를 위해 정의된 "myLogger"의 attribute 들을 나타낸다.

```
trendLog myLogger  
{  
  diskLogLimIntvl = 60;  
  timeTickIntvl = 1;  
  timeMaster = 2;  
  timeVariable = "timeVar";  
  trendVariableFile = "CNSTrendVars.ptrv";  
  diskFlushIntvl = 0;  
  diskBufferSize = 30;  
  diskForceWrite = 120;  
}
```

Application CNS_GUI 의 다른 statements 들은 "SomeShapes" class 로써 polygon, circle, line 으로 complex shape 이 정의되었으며, type "window", 사용되는 class library 를 정의하는 "use" statements, 그리고 type "colour", type "lineStyle" 등이 정의되었다.

3.2 Class Library

Picasso-3 에서는 그래픽 에디터를 통해 그래픽 클래스를 정의하거나, Picasso-3 Syntax 에 따라 states 를 attributes 및 graphic shapes 으로, behavior 를 functions 및 dialogues 로 정의할 수 있다. CNS 의 그래픽 클래스는 Trend, XYPlot 에서부터 polygon 에 이르는 클래스 라이브러리가 정의되어 이용되었으며 library "TrendLib"의 "trend"를 나타내면 다음과 같다.

trend T

```
trLog = "myLogger"; ..... // attributes definition
trendPlot TP1 { variable = `Value`; // "value" of trend parameter
               ..... // other attributes of trendPlot TP1 }
trendPlot TP2 { // the 2nd trend curve }
trendPlot TP3 { // the 3rd trend curve }
trendLabels TL1 { // attributes definition including "offset"}
trendRuler TR1 { // attribues definition including "depthPosition" }
trendGrid TG1 { // attributes for defining grids on the defined window }
dialogue { event = `cursorMoved()`;
           action = statement `{
               TR1.moveRuler(); theWindow().update(); }; }
           // others
} // end of trend T
```

이러한 "TrendLib" library 외에도 "arrow", "textField", "slider", "scrollBar", "checkButton", "pushButton", "scale" class 들이 정의된 "MotifLib" library 와 "horizontalLabel", "verticalLabel", "XYPlot", "Bargraph", "bar3D", "Bargraph3D", "Bargraph3DDepth", "PieSlice" class 들이 정의된 "PlotsLib" library 가 이용되었다. 그리고, P&ID 를 위한 그래픽 클래스들은 밸브, 탱크, 증기 발생기, 원자로 냉각재 펌프, 원자로 등 발전소 기기들에 대해 정의된 "PIDsLib" library 가 이용된다. 현재, 고리 2&3 호기에 대한 "overview", "1 차계통"에 대한 P&ID 가 완성되어 리모콘 메뉴의 각종 diagram(11-15)에서 이러한 P&ID 를 선택하도록 하며, 3 개의 그래픽 터미널 중 하나에는 "overview" 화면이 나타나도록 배치한다.

3.3 API

Picasso-3 application "CNS_GUI"와 "Run-Time Manager"간의 인터페이스로써, CNS 모의 프로그램의 공유 데이터를 1 초마다 읽어서 "CNS_GUI"의 변수값을 update 하기 위한 프로세스 "CNS_API"가 C 언어와 Picasso-3 의 "function"들을 이용하여 정의되었다. Picasso-3 를 이용한 CNS GUI 의 구현 결과를 그림 3 에 예시하였다.

IV. 결론

프로세스 모니터링을 위한 그래픽 디스플레이 개발에 유용한 trend graphs, P&IDs, 메뉴 등을 위한 대부분의 그래픽 인터페이스 구성요소의 정의에서부터 유지 보수에 이르는 작업이 Picasso-3 를 이용함으로써 효율적이다. 그리고, 객체 지향 프로그래밍에 의해 클래스 정의를 재사용할 수 있는 장점, Picasso-3 의 높은 이식성, 그리고 PC/NT Picasso-3 를 활용한다

면 다양한 GUI 개발이 가능하며, Picasso-3 소프트웨어가 OECD Halden Reactor Project 의 회원국이나 기관에는 무료로 보급되므로 별도의 GUI 개발용 소프트웨어를 구입하지 않고도 GUI 개발에서의 생산성과 경제성을 얻을 수 있다.

알 립

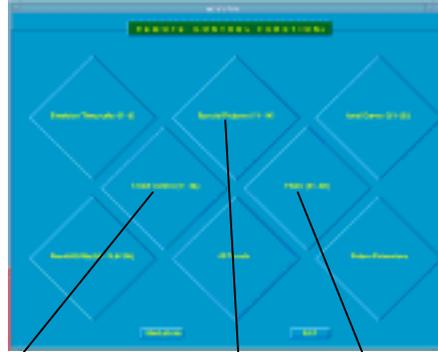
본 연구는 과학기술부의 원자력 기반연구 사업으로 수행되었음.

참고문헌

1. 박 재창 외 10 인, "CNS 성능향상 및 노후 장비 교체," 연차 보고서 KAERI/RR-1794, 1997.
2. Kjella Ame Barmsnes, et al., "PICASSO: a user interface management system for real-time applications," Presented at Workshop on User Interfaces for Expert Systems), London, March 11-12, 1992.
3. Kjella Ame Barmsnes, et al., "Developing graphics applications in an interactive environment," Presented at 1994 SCS Simulation Conference, April 11-13, 1994, San Diego, California, 1994.
4. Oystein Jakobsen, Terje Johnsen, Hans Olav Randem, "The Picasso-3 User Interface Management System," Presented at the OECD Halden Reactor Project EHPG Meeting, Storefjell, 8th-11th March, 1993.
5. <http://www.ife.no/projects/picasso/UGM0398/presentations/>.
6. Picasso-3 Reference Manual. URL=<http://www.ife.no/projects/picasso/documentation/>.
7. Picasso-3 User's Guide. URL=<http://www.ife.no/projects/picasso/documentation/>.
8. Picasso-3 Tutorial. URL=<http://www.ife.no/projects/picasso/documentation/>.
9. 권기춘, 계측제어시험검증설비 사용자 지침서, 기술보고서(KAERI/TR-729/96), 1996.
- 10.오영환·신성용 공역, 컴퓨터 그래픽스, 흥릉과학출판사, 1990.

그림 3. Picasso-3 를 이용한 CNS GUI 구현 예

Application Level:



Picture Level:

