

# A toolkit for computerized operating procedure of complex industrial systems with IVI-COM technology

ZHOU Yangping<sup>1</sup>, YE Jingliang<sup>2</sup>, DONG Yujie<sup>3</sup>, HUANG Xiaojing<sup>4</sup>, and YOSHIKAWA Hidekazu<sup>5</sup>

1. Institute of Nuclear and New Energy Technology, Tsinghua University, Energy Science Building, Haidian District, Beijing 100084, China (zhouyp@mail.tsinghua.edu.cn)

2. China Ship Study and Design Centre, Wuhan 430064, Hubei, China (yejinliang222@163.com)

3. Institute of Nuclear and New Energy Technology, Tsinghua University, Energy Science Building, Haidian District, Beijing 100084, China (dongyj@mail.tsinghua.edu.cn)

4. Institute of Nuclear and New Energy Technology, Tsinghua University, Energy Science Building, Haidian District, Beijing 100084, China (huangxj@mail.tsinghua.edu.cn)

5. Kyoto University/Harbin Engineering University, Nantong Street, Nangang District, Harbin 150001, China (yosikawa@kib.biglobe.ne.jp)

**Abstract:** A human interface toolkit is proposed to help the user develop computerized operating procedure of complex industrial system such as Nuclear Power Plants (NPPs). Coupled with a friendly graphical interface, this integrated tool includes a database, a procedure editor and a procedure executor. A three layer hierarchy is adopted to express the complexity of operating procedure, which includes mission, process and node. There are 10 kinds of node: entrance, exit, hint, manual input, detector, actuator, data treatment, branch, judgment and plug-in. The computerized operating procedure will sense and actuate the actual industrial systems with the interface based on IVI-COM (Interchangeable Virtual Instrumentation-Component Object Model) technology. A prototype system of this human interface toolkit has been applied to develop a simple computerized operating procedure for a simulated NPP.

**Keyword:** computerized operating procedure; complex industrial system; IVI-COM; human interface toolkit

## 1 Introduction

Owing to the development and maturation of computer and related technologies, digitalization is apparent in many fields of complex industrial systems such as Nuclear Power Plants (NPPs) [1]. It is believed that the application of these operating procedure systems can improve the safety and reliability of complex industrial system and reduce the worker's work load [2-5]. However, the design, development and maintenance of operating procedure systems such as digital operating procedure under both operational states and accident conditions require not only a profound understanding of design, operation and structure of NPP but also expertise on information technology.

Considering the aforementioned reasons, a human interface toolkit is proposed to help the user develop the computerized operating procedure of complex

industrial system. With a friendly graphical interface, this integrated tool includes a database, a procedure editor and a procedure executor.

In this database, a three-layer hierarchy is adopted to express the complexity of the operating procedure, which includes mission, process and node. There are 10 kinds of node: entrance, exit, hint, manual input, detector, actuator, data treatment, branch, judgment and plug-in. With the procedure editor, user can easily develop and maintain the procedure and the finished procedure will be stored in the database. The procedure executor can then load the procedure from the database for operating procedure and thus act as a computerized operating procedure. The computerized operating procedure will sense and actuate the actual industrial systems with the interface based on IVI-COM (Interchangeable Virtual Instrumentation-Component Object Model) technology [6] embedded in detector and actuator nodes. With the help of various nodes, processes and missions, the developed digital system can access

information from plant, make interaction with operator, call additional application, and so on.

According to the design mentioned above, a prototype system of this human interface toolkit has been developed with Visual C++, DirectX SDK, MSXML Parser SDK, Microsoft Agent SDK and SQL server. The prototype system is applied to develop a simple computerized operating procedure for a simulated NPP. Several operating conditions are tested on the prototype system and the developed computerized operating procedure in order to validate and improve the proposed work.

## 2 Hierarchy for operating procedure

A three-layer hierarchy is adopted to express the complexity of operating procedure, which includes mission, process and node. A mission denotes a specific task which the operator wants to finish, such as drop of the power during normal operation, startup of plant, *etc.* Usually, a mission can be divided into several independent processes. A certain process consists of a chain of actions, termed here as nodes. At present, ten kinds of node are applied to express the various actions in the operating process:

- Entrance: Startup point of whole mission or entrance from upstream process to present process.
- Exit: End point of whole mission or exit from present process to downstream process.
- Hint: Can provide information to operator with text or video.
- Manual input: Operator can manually input a value to the procedure system.
- Detector: Provide input interface with a real detector in an industrial system based on IVI-COM technology.
- Actuator: Provide input interface with a real actuator in an industrial system based on IVI-COM technology. Here, an actuator indicates an automatic mechanism by which a signal acts to control the equipment of the industrial system, such as valve, pump and so on.
- Data treatment: Treat the value in procedure system with numeric calculation, string treatment, base conversion, *etc.*
- Branch: Automatically determine the destination node according to the judgment conditions.

- Judgment: Operator can make a logical (YES/NO) judgment according to the condition.
- Plug-in: Provide interface with external program which can start the external program if necessary.

There are two kinds of connectors in this hierarchy, process connector and node connector. The process connector connects the downstream process with the upstream process, while the node connector connects the downstream node with the upstream node. In addition, the process connector is affiliated with an exit in an upstream process and an entrance in a downstream process at node level.

Figure 1 shows a simple example of this hierarchy. There are four processes in mission 1: processes 1, 2, 3, and 4. There are four nodes in process 1 (*i.e.*, entrance 1, hint 1, detector 1 and exit 1) and four nodes in process 2 (entrance 2, hint 2, detector 2 and exit 2). The process connector between process 1 and process 2 is affiliated with exit 1 in process 1 and entrance 2 in process 2 at node level. Sometimes there is more than one process connector between two processes but only one node connector between two same nodes.

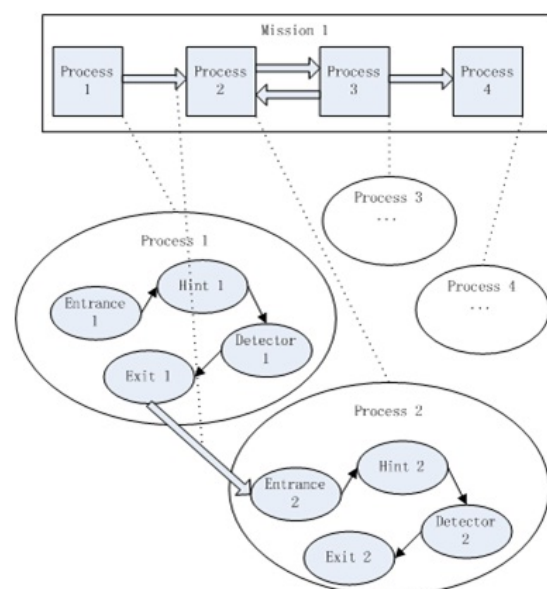


Fig. 1 Hierarchy for operating procedure

## 3 Design of procedure development toolkit

The correlations among target system, operating procedure and procedure development toolkit are

shown as Fig. 2. The procedure development toolkit has a procedure editor and a procedure executor. The work of procedure development toolkit consists of two stages, editing stage and executing stage.

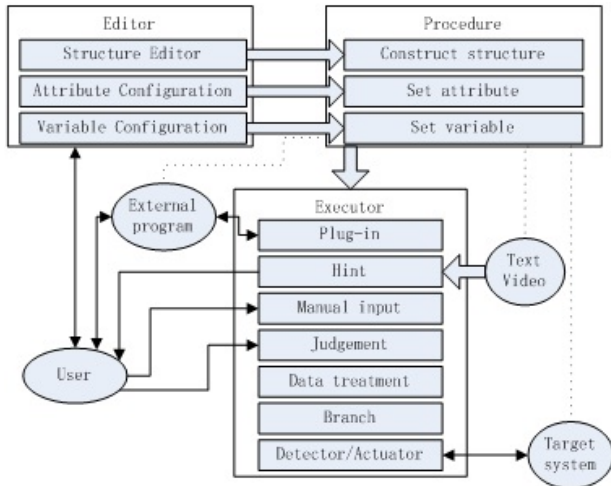


Fig. 2 Target system, operating procedure and procedure development toolkit

During editing stage, the operating procedure will be constructed and maintained by user with the help of the editor. The external multimedia files such as text and video will be affiliated with the hint node in the procedure to provide information to users. In order to extend the ability of developed procedure system, a plug-in function is provided in the plug-in node which is embodied with an interface for loading the external program and transferring data between the developed system and external program. The attribute of detector and actuator node can be set with the editor to configure IVI-COM interface among procedure system, actual detector and actuator. A set of variables can be established with the help of the editor, and the name, type and value can be set by the variable configuration tool. In addition, the variable can be affiliated with the various nodes. With a friendly graphical interface, users can establish the digital procedure whose information including mission, process, node, connector and variable is stored in a database system.

During executing stage, the executor loads the procedure from the database and acts as a computerized operating procedure. When a mission of the procedure is activated, the executor will start from an entrance node which is a startup point of the

mission. The working point of the procedure will then go to next node which is the downstream node of this entrance node. The executor will act differently according to the type of the node. If the node is a plug-in, the affiliated external program will be activated, which can load needed data and provide support to operator. The hint node will provide information to operator by using affiliated text and video. Through manual input node, user can input the value to the variable of system. When arriving at judgment node, user can make a YES/NO answer to the question. Data treatment node can treat the value of variable and branch node can make a jump to a certain node according the value of a variable. Detector node will get value from the real detector in the target system and actuator node will actuate the real actuator in the target system for system control. The detector node and the actuator node work based on IVI technology.

## 4 IVI architecture

IVI is an integral component of a National Instruments test system. The IVI Architecture is shown in Fig.3. IVI is located above the VISA I/O layer in the program hierarchy and is integrated into the application development environments provided by National Instruments. The IVI architecture breaks the traditional instrument driver into two parts – an instrument-specific driver and a class driver. The instrument-specific driver functions the way traditional instrument drivers have in the past, but with an underlying architecture that is optimized for performance and includes instrument simulation. The class instrument driver contains generic functions for controlling an instrument category and calls the corresponding instrument-specific driver functions at run time. The test program can be written with either the class driver or the specific driver.

The IVI Foundation has defined the IVI architecture to work with two interface technologies, one based on the ANSI C standard (IVI-C) and one on Microsoft COM (Component Object Model) technology (IVI-COM). The two architecture types are designed to be interoperable.

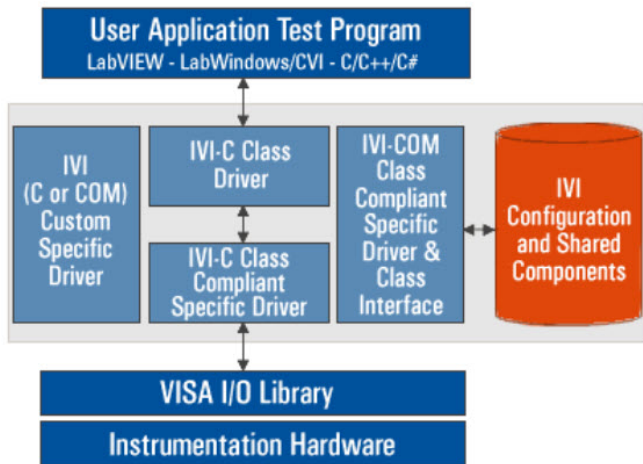


Fig. 3 IVI architecture

## 5 Prototype system of integrated tool

According to the design mentioned above, a prototype system of this integrated tool has been developed with Visual C++, DirectX SDK, MSXML Parser SDK, Microsoft Agent SDK and SQL server. The prototype system is applied to develop a simple computerized operating procedure for a simulated NPP. Several operating conditions are tested on the prototype system and the developed computerized operating

procedure in order to validate and improve the proposed work.

Figure 4 shows a snapshot when the procedure editor of prototype system is working to establish the demo operating procedure. The explorer can manage the mission, processes and nodes in a tree list mode. User can easily find, choose or delete the object in the procedure with various operations such as expand, fold, *etc.* With the help of mission editor, user can append, delete and choose the symbols of process and process connector with simple mouse operation. User can conveniently append, delete and choose the symbols of different nodes and node connectors with the process editor. At the bottom of the window, configuration assistant can assist user to configure the attribute of mission, process, node, process connector and node connector when the relevant object is chosen in explorer, mission editor or process editor. In addition, the variable can be added, deleted and configured in configuration assistant.

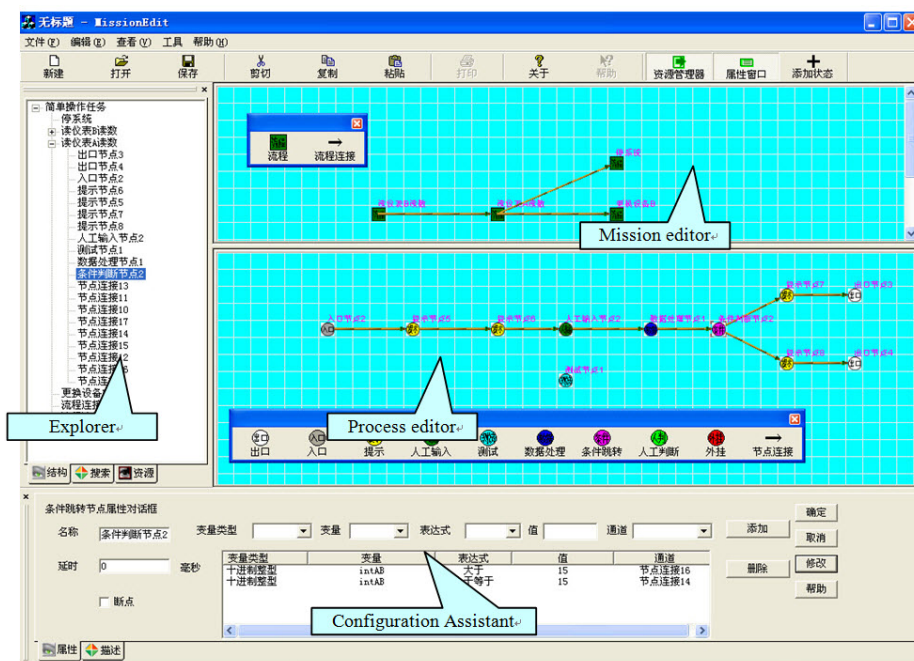


Fig.4 Editor of procedure toolkit

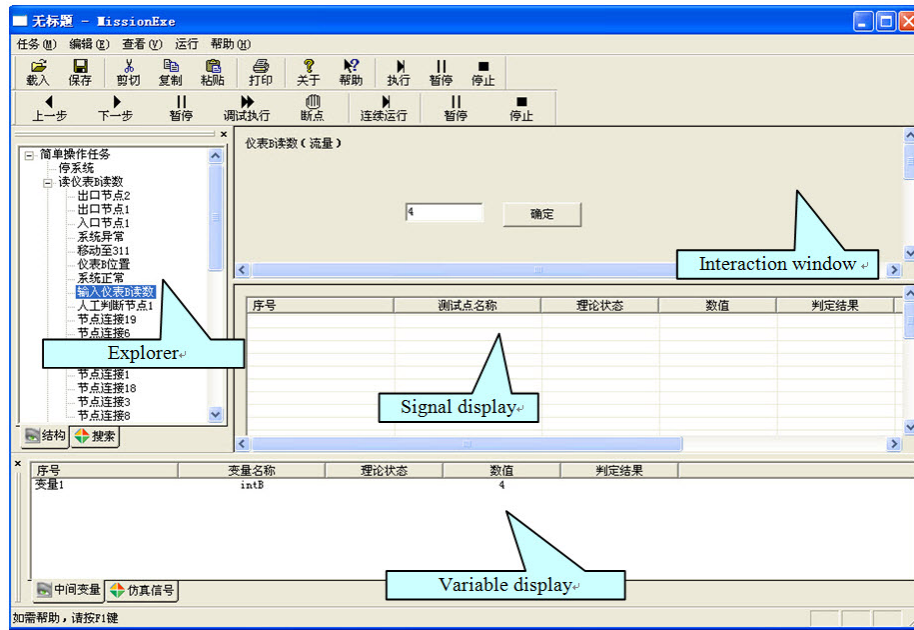


Fig. 5 Executor of procedure toolkit

Figure 5 shows the scene when the procedure executor of prototype system loads the procedure and acts as a computerized operating procedure. The executor can work in three modes: debug mode, run mode (with breakpoint) and run mode (without breakpoint). In the debug mode, the executor will stop at each object to help the developer of procedure confirm and validate the procedure. In the run mode with breakpoint, the executor will stop at breakpoint and interaction node. The executor will only stop at interaction node, such as manual input, judgment, *etc.*, in which the interaction between user and procedure executor is required. In the executor, the explorer can be used to manage the mission, processes and nodes in a tree list mode. The present node which the executor runs on will be marked in the explorer. In the interaction window, user can make interaction with the executor. For example, user can input a value to a variable when a manual input node is executed. The signal display can display the state of the signal which is derived from detector and related node. The variable display can be used to monitor the state of variable.

## 6 Conclusions and perspectives

With an integrated graphical interface, a procedure development toolkit is proposed for the development and maintenance of application for Man-Machine Interaction (MMI) such as operation procedure system, *etc.* In terms of this integrated tool, operating

procedure for different target systems can be constructed, maintained and implemented conveniently.

In this study, a prototype system of this integrated tool has been developed by using Visual C++, DirectX SDK, MSXML Parser SDK, Microsoft Agent SDK and SQL server. A demo operation procedure system for a Nuclear Power Plant simulated by RELAP5/MOD2 has been developed by using this prototype system. Several operating conditions are tested on the prototype system and the developed computerized operating procedure in order to validate and improve the proposed work.

The proposed integrated tool is now far away from the actual application to the industrial system. In future, the proposed integrated tool will be improved according to the following considerations:

- Possible support to smooth and rapid migration from paper based operating procedure to digital operating procedure.
- Improvement of the human interface especially for the executor by considering the actual requirement of operator when operating the industrial system.
- Interface for connecting other software and hardware in the actual control room efficiently and reliably.

- Improvement of reliability and efficiency of the integrated tool.

## Acknowledgement

The work of this paper is sponsored by Chinese Major National S&T (ZX06901).

## References

- [1] ZHANG L.: Design of Digital Operating Procedure for FuQing NPP. In: Technical Meeting on the Impact of Digital I&C Technologies on the Operation and Licensing of NPPs, Beijing, China: 2008.
- [2] LIU F., Zhang Z., Peng M.: Development of Emergency Operation Procedure System for Nuclear Power Plants. Chinese Nuclear Power Engineering, 2008, 29(4): 104-108.
- [3] PARK J., JEONG K., JUNG W.: Identifying cognitive complexity factors affecting the complexity of procedural steps in emergency operating procedures of a nuclear power plant. Reliability Engineering and System Safety, 2005, 89: 121-136.
- [4] NIWA Y., HOLLNAGEL E.: Integrated computerisation of operating procedures. Nuclear Engineering and Design, 2002, 213: 289-301.
- [5] LEE S.J., SEONG P.H.: Development of automated operating procedure system using fuzzy colored petri nets for nuclear power plants. Annals of Nuclear Energy, 2004, 31: 849-869.
- [6] IVI information, <http://www.ni.com/ivi/>.