Methodological basis of plant DiD risk monitor system development and its prospective applications for NPPs

NAKAGAWA Takashi[1], TERASHITA Naotaka[2], and YOSHIKAWA Hidekazu[3]

1. Prime System Laboratory Co., Ltd., Suimeidai 4-4-13, Kawanishi-Shi, 666-0116 Japan (Nakagawa@prime-system.jp)
2. System P&A, Ryuge-Cho 1-4-2-2207, Yao-Shi, 581-0069 Japan (terashita.naotaka@system-panda.jp)
3. Symbio Community Forum, Co. RIAS, Tanaka-Ohi-Cho 49, Sakyo-Ku, Kyoto, 606-8202 Japan (yosikawa@kib.biglobe.ne.jp)

Abstract: A new risk monitor system is under development which can be applied not only to prevent severe accident in daily operation but also to serve as to mitigate the radiological hazard just after severe accident happens. The system simulates these applied situations, by generating interactions among all actors such as the plant and related people coping with the accident. In this paper, the plant DiD risk monitor has been developed utilizing Unified Modeling Language (UML) to model actors' knowledge and express the interactions. Case study to confirm functionality and ability of DiD risk monitor is also introduced by showing examples for Station blackout accident in conventional PWR plant in Japan. Prospective application areas are also discussed of the developed plant DiD risk monitor for the safety improvement of nuclear power plants.
Keyword: defense-in depth; risk monitor system; object oriented program; unified modeling language; state chart diagram; sequence diagram

1 Introduction

The authors of this paper have been developing a new risk monitor system, in order not only to prevent severe accident in daily operation but also even to serve as to mitigate the radiological hazard just after severe accident happens and long term management of post-severe accident consequences[1]. The conspicuous features of the proposed risk monitor to be compared with the existing risk monitors basically lie on the two points: (i)The range of risk is not limited to core melt accidents but includes all kinds of negative outcome events, *i.e.*, not only precursor troubles and incident but also any types of hazard states resulting from a severe accident, and (ii)The whole system of the proposed risk monitor system consists of plant Defense-in Depth (DiD) risk monitor and reliability monitor. The relation between the both monitors was discussed [2], although no detailed explanation of the reliability monitor will be made in this paper.

The method of how to configure the plant DiD risk monitor by functional modeling approach was first presented [3], with a preliminary study applying for passive safety system of AP1000. The similar preliminary study was conducted for active safety system of conventional PWR in Japan [4]. Software development method for the plant DiD risk monitor was presented [5], where the basic idea of configuring knowledge-based data utilizing State Chart Diagram was also proposed.

Following those preceding studies, brief summary of the authors' proposed risk monitor system for NPP is given in **2**. The software configuration of the plant DiD risk monitor is given in **3**. A case study to conform the functionality and ability of the plant DiD risk monitor is conducted in **4** for station black out accident of conventional PWR in Japan. Lastly, prospective application fields of the plant DiD Risk Monitor are reviewed in **5**.

2 Proposed risk monitor system for NPP

The authors' proposed risk monitor system is constituted by two layered systems as depicted in Fig.1. Basically it is composed by a Plant Defense in Depth (DiD) Risk Monitor and several Reliability Monitors. The Plant DiD Risk monitor simulates dynamic plant situation, which is generated by interactions among all actors such as the plant safety systems and the related human organization to cope with the situation, and evaluates plausible risk state for the situation. Several Reliability Monitors evaluates

Received date: May 10, 2017
(Revised date: May 24, 2017)

the reliability of individual subsystems to fulfill their expected functions successfully under the prescribed conditions, which are given by the Plant DiD Risk Monitor.

In Fig.1, various Knowledge Bases (KBs) which will be used for both Plant DiD Risk Monitor and Reliability Monitors are listed up in the block which is indicated as "KB for risk monitor". The plant DiD risk monitor will identify every potential risk state caused by any conceivable event in the plant system as a whole where not only internal events but also external events arising from common cause factors and human factors should be taken into account.

Reliability evaluation for a sub-system is conducted by the Reliability monitor by using a combination of failure mode and effect analysis (FMEA) and GO-FLOW [6]. Reliability is normally defined as the successful rate of a system's performance that will fulfill its expected function when it is requested to work successfully. In the safety design of nuclear power plant, reliability of safety functions is enhanced by principles of diversity, redundancy and physical separation.

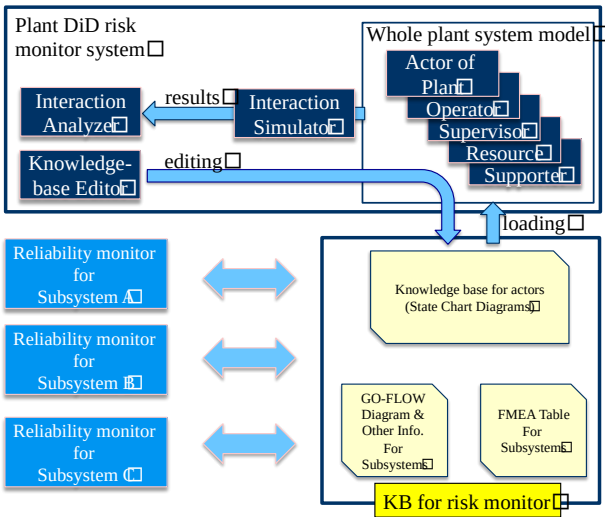

Fig. 1 Authors' proposed risk monitor system.

The method of how to apply a success tree oriented reliability analysis method GO-FLOW had been extensively studied for the reliability monitor of the real safety systems of PWR plants with the practical examples [7,8].

3 Development of plant DiD risk monitor

Since plant DiD risk monitor will be utilized to analyze and evaluate various risks caused by operation of nuclear power plant, it will be necessary to introduce a certain comprehensive framework to describe "types of analysis scenario". Table 1 shows a classification of operation modes for nuclear power plant operation which corresponds to types of analysis scenario. There are very many cases to consider in advance on different types of operation modes of plant process both in normal and in design-basis off-normal situations (A in Table 1) and "out of normal imagination" situations (B in Table 1).

Table 1 Classification of operation modes for nuclear power plant.

A. Design basis	Normal operation	Start-up, Steady state operation, Power change and shutdown
		Refueling and maintenance testing
	Off-normal	Anticipated transient, accident
	Design basis severe accident	
B. Imaginary emergency situation		

The both types of operation modes A and B as discussed in the Table 1 are also related with human factors, *i.e.*, the way of how to prepare operation procedures for the human operator and the operator training. It is said in human factors area that there are two types of human task: skill and rule based routine task and non-routine knowledge based task. It is also well known by human factors research that the operator's action becomes automated by proper training on the basis of acquired knowledge base on versatile behaviors of machines and plant systems. This situation is basically for the mode A in Table 1. However, there remain unfamiliar situations when operators have to cope with it by problem solving from scratch. This situation will be mode B in Table 1. The authors of this paper would like to start the issue by considering how to configure human-machine interaction model based on A in Table1.

The basic idea of the plant DiD risk monitor can be summarized by the following ways:(i)The whole plant system should be modeled by the combination of functional models for machines, operators, supervisors and other people coping with the plant operation. These models are called as “actors” in this paper. (ii)The dynamic behavior of the whole plant system can be simulated by the interaction among these actors. (iii)The actors have scenario data and will behave based on the scenario data. In order to create these scenario data easily and intuitively, the authors applied "State Chart Diagram"[11], which are extensively used in the field of systems engineering to model the behavior of the computer systems. (iv)The DiD risk monitor provides functions for investigating whether the simulated plant situation would be desirable or not, whether the interaction also would be suitable or not, and if it is inappropriate, what will be the causes of it.

The DiD risk monitor developed to realize the above idea consists of three subsystems: (i) Knowledge-based editor, (ii) Interaction simulator, and (iii) Interaction analyzer. Knowledge-based editor provides editing functions to create scenario data in the form of the "State Chart Diagram". The Interaction simulator drives all actors based on their scenario data and simulates the whole plant situation as a result of these behaviors of the actors. The Interaction analyzer shows how actors behave with one another and in what order in the form of "Sequence Diagram". The "Sequence diagram" is also widely used in the field of the system engineering to show the interactions such as event exchange and operation among system modules in time sequence.

These subsystems are developed as a plug-in of Integrated Development Environment "Eclipse"[9] with the use of Graphical Editing Framework "GEF"[10]. Those software modules and the libraries only depend on Java, an object oriented programing language which does not depend on any platforms, and therefore software system of DiD risk monitor can be installed on any Windows-PC or Macintosh-PC.

The details of those three subsystems, Knowledge-based editor, Interaction simulator, and Interaction analyzer will be described in the following subsections.

3.1 Knowledge-based editor

What is called "State Chart Diagram" is employed to express scenario data for each actor, and it is defined by Unified Modelling Language (UML) Ver.2.0 [11].

The UML has been successfully used for modelling the software modules in the software engineering, and especially, "State Chart Diagram" has been widely used to describe dynamic behaviours of software modules. By applying "State Chart Diagram" to model all knowledge-based information for the all kinds of actors, the following two merits of A and B are expected:

A. High capability to model dynamic behavior: "State Chart Diagram" can model dynamic behaviors of the modeling target in different levels of abstraction (from abstract/outline to concrete/detailed). The model is easy to understand intuitively by users.
B. Simple modeling of interaction: "State Chart Diagrams" can model behaviors of the target by using states and transitions between the states and event handler, which causes the state transition. The interactions among actors are simply described by sending events among "State Chart Diagrams" in order to handle the event and make state transition smoothly.

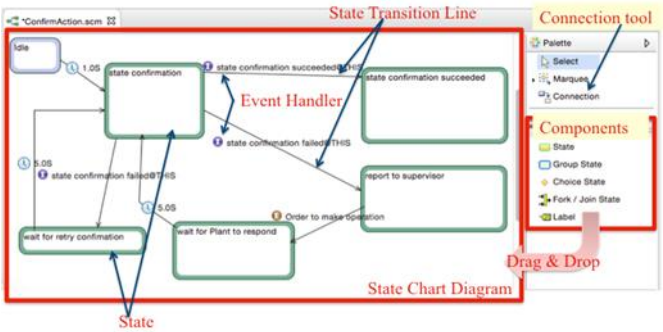

Fig.2 Snapshot of Knowledge-based editor.

Figure 2 shows a snapshot of the Knowledge-based editor. A canvas in the center of the screen shows "State Chart Diagram" which is operator's knowledge to confirm a machine status. The users can drag and drop a state, a label and so on, by selecting it from

the right side area named "Components" and dragging into the canvas. A “transition line” between the states can be drawn by Connection tool in the upper-right area named “Palette”.

The role of transition line is to connect from a source state to its target state by an arrow line. It also holds several event handlers to make this state transition. When a certain event is received by an event handler, then the handler for this event makes the state transition and execute the command sequence which is defined as the action of this event handler. The users can write these command sequences in Java style program. In plant DiD Risk Monitor, the following 4 types of events (A), (B), (C), and (D) and its handlers can be defined in "State Chart Diagram":

(A) Actor External Event - Actor External Events are transmitted among actors. For example, the plant actor sending an alarm as an "Actor External Event", other actors such as an operator and a supervisor, which have the corresponding event handler, receive and react the event of the occurred alarm. Therefore, the interaction among actors is simulated by sending and handling the "Actor External Events".

(B) Actor Internal Event - Actor Internal Events are used to communicate among the State Chart Diagrams within one actor.

(C) Primary Event - When states become active or inactive, the state generates a primary event such as "OnEntry" or "OnExit" on that time automatically. The corresponding event handler can be defined to execute some process in that timing.

(D) Timer Event - Timer event is generated after its pre-defined duration time.

The "State Chart Diagram" as shown in Fig.2 represents a task model of "Confirmation task" to confirm whether or not a certain machine state becomes a certain required status. Because operator's task in plant contains many confirmation tasks for various machines, these common tasks such as confirmation task should be modeled as software components and these components should be used repeatedly to make whole knowledge-based data efficiently.

To “componentize’ (to make the elemental task models as ‘component’ by the forms of "State Chart Diagram"), target dependent information *(i.e.* machine name, desirable status and so on) should be separated from "State Chart Diagram" model. To achieve such requirement, the employed "State Chart Diagram" model has the parameters area to handle these target dependent information. The machine name and its desirable status are given as variables through the parameter area. So, the "State Chart Diagram" can be used as software component, and therefore, its role seems like a subroutine of the computer programming.

All required basic tasks are provided by the form of components using the mechanism of the “componentization”. So general users can make scenario data easily and rapidly only by choosing the component and placing it on state. The componentization hides the detailed and complex information like a programming technique from the general user, and make the usage of the plant DiD Risk monitor easier.

All basic tasks are shown in Fig.3, while an example of creating the scenario data by using these components in Fig.4, respectively.

In Fig.4, the user places "State" on the canvas 4 times and connects them from left to right by transition lines. For selecting a suitable component from "Components" in the right side of the window, the user can drag and drop the selected component on the state respectively. For selecting the component on the state, the user can define required properties of the component such as required time and required number of members to complete the task modeled by the component.

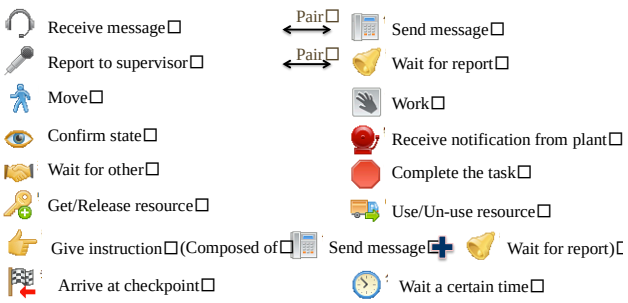

Fig.3 Component list for all basic tasks.

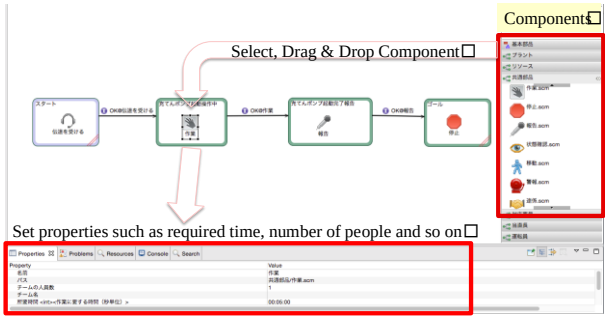

Fig. 4 Example of creating scenario data using Components.

Then the user should set event handlers for these transition lines. Figure 5 shows the direction of the setting event handlers. State-A holds a working component modeling working-task of the operator, the name of the task and required time to fulfill it are also set into the component as properties. All components for the basic task are developed to generate Actor Internal Event named "OK" when the task is completed. Therefore, the event handler for the Actor Internal Event should be placed on the transition line from State-A to State-B. Because Actor Internal Event handler should be expressed as "EventName"@"Generate Place Name of the Event" in this system, the internal event handler named OK@OpenValve should be place on the transition line.

When the interaction simulator drives the "State Chart Diagram", completing the working-task at State-A, "State Chart Diagram" will make transition to State-B by the Actor Internal event handler named "OK@OpenValve" and start to do the moving-task placed on State-B.

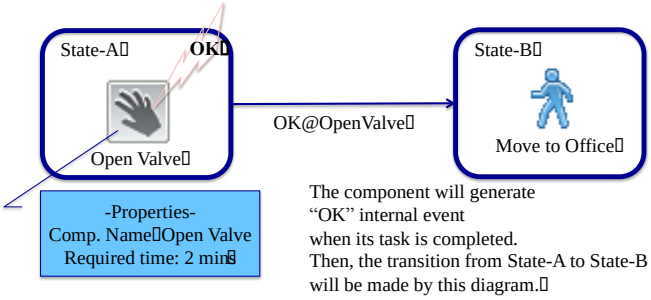

Fig.5 Directions of setting event handler.

3.2 Interaction simulator

Once users convert all knowledge-based information for a given accident scenario into a set of "State Chart Diagrams", the users can execute the interaction simulation among actors by activating the interaction simulator. The execution time of each basic task will be affected by the following five factors A to E:

A. Component for the basic task has a variable for required time. The variable will be used to simulate the execution time to complete the task.
B. Traveling time: When people move to other place, the traveling time will be added, which is calculated by the distance and the way of getting there (by walk or by car).
C. Required number of people: Each "State Chart Diagram" needs required number of people to complete it. The interaction simulator tries to get the required number of people to execute the "State Chart Diagram". If it failed, the execution will be postponed until it gets enough people.
D. Waiting time for other tasks: If the scenario includes sequential tasks, a person needs to wait accomplishment of leading tasks.
E. Required time to get equipment: If people fail to get equipment such as cars, tools and materials which is necessary to complete a task, he/she has to wait until the equipment becomes available.

3.3 Interaction analyzer

The results of the interaction simulation are evaluated by the interaction analyzer from the following aspects A, B, and C of the evaluation goal.

A. The result of the simulation is out of expectations -In the case that the simulation result is out of expectations or the simulation stops in the middle of the scenario, it is considered that the scenario data might have some errors or incorrectness. If these scenario data were entered by the actual operators or supervisors based on their understandings, their understandings might have errors or incorrectness. By correcting the problems in their understandings and the scenario data, then simulating them again, this iterative process will make their understandings for the emergency situation, their roles and the scenario deeper and wider.
B. Steps cannot complete in the limitation time - Some steps in an emergency procedure generally have limitation time to do them. If such a step is done over the limitation time by the interaction simulation, the investigation for the cause should be conducted. The cause might be the potential

problems of the scenario, lack of people and/or lack of resources.

C. The simulation result is not desirable – This is the case that the simulation is executed on the given scenario, but the result is not desirable. For example, when it becomes core melt accident, radiological hazard as the result, the investigation should be conducted to find turning points which can exit the current scenario and the countermeasures should be considered to exit the scenario at the turning points such as finding and conducting alternatives of reactor cooling and keeping integrity of the containment vessel.

To support these goals of the investigation, the interaction analyzer shows the result of the interaction simulation as in form of the "Sequence Diagram" as shown in Fig. 6. The "Sequence diagram" is also defined by UML and widely used in the field of the system engineering to show the interactions such as event exchange and operation among system modules in time sequence.

All actors are laid out horizontally and the interaction among actors such as event exchange and the execution of the task are arranged in time sequence. The arrow line between actors shows sending and receiving Actor External Events. Thin arrow lines are for sending the event. Thick arrow lines are for receiving and handling the event. One red box shows executions of one basic task. Plural red boxes arranged vertically shows the execution plural steps defined in a "State Chart Diagram".

The green labels placed on the left side of Fig.6 mean that a certain step can be executed before its limitation time.

Purple lines in red boxes show the traveling process. Yellow lines mean receiving the instruction from other actors. After receiving the instruction, required number of the people is tried to get. If it succeeds, the next task is executed immediately and the red box of the task is placed just below the yellow line. If it does not succeed, a green line is drawn to show the task was postponed. Blue lines mean waiting time for others.

If user finds some problem such as long postponed task (green line), long waiting time (blue line), no reaction (no thick arrow line) and so on, by right-clicking a line, a red box or an arrow line, the corresponding state or event handler of the "State Chart Diagram" is displayed immediately by the Knowledge-based editor so that the user can check the scenario data and correct it rapidly.

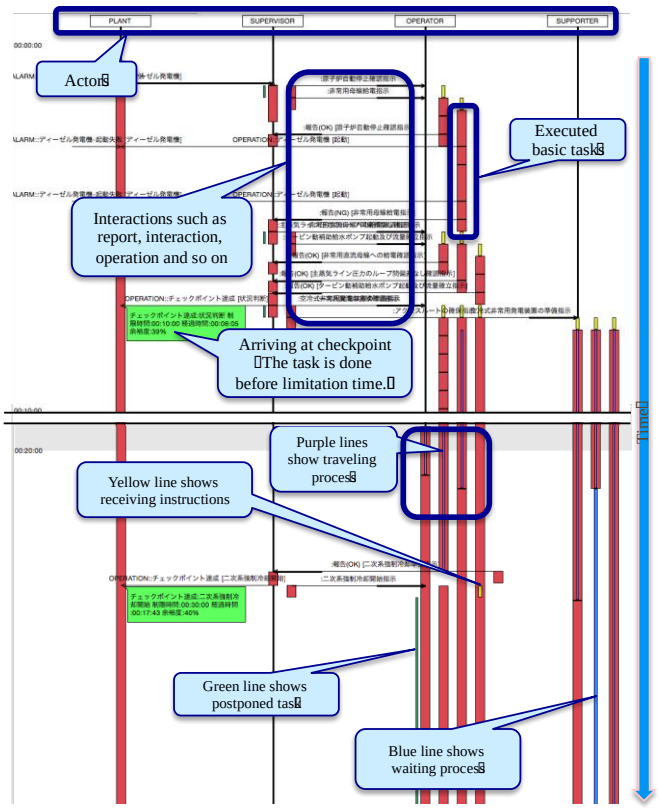

Fig.6 Example of sequence diagram.

4 A case study for successful severe accident management

A case study is conducted by the developed plant DiD risk monitor in order to confirm basic functions such as modelling the knowledge of actors, simulating the interaction among the actors and analysing the simulation results. The conducted case study is for the successful management of a conventional PWR in Japan to cope with all station blackout accident by scenario-based procedure.

The outline of the scenario is (i) station blackout accident occurs because emergency diesel generator is failed to start, and (ii) plant personnel try to cope with the accident by setting up alternative generators and fire engine tracks to pump seawater to activate

core-cooling functions. This scenario is seriously considered in Japan to reflect the lessons from Fukushima Daiichi NPP accidents in March, 2011.

The base case of scenario study is that the emergency team formed by 2 supervisors, 8 operators and 17 supporters will try to manage the accident. This scenario also contains following checkpoints and limitation times; judge the accident in 10 min, start forced cooling of secondary system in 30 min, supply electricity from alternative generator in 1 hour, start alternative water injection into core by charging pump in 2 hour 20 min, hot shut down status in 4 hour, available to supply seawater to auxiliary feed water tank in 11 hour, available to supply seawater to cv recirculation unit and high pressure injection system in 51 hour. The case study is conducted by the following four steps;

(1) Input scenario data in the form of "State Chart Diagram": The total time to input this scenario data was 1 - 2 hours. If the detail of the procedure is clear, input work is very simple and easy because the required components for the basic tasks are provided. The input work also made user's understandings for the procedure clear. Knowledge-based editor supported to find careless mistakes such as name mismatch between component and event handler, lack of event handler, lack of component and so on.
(2) Run through the whole scenario: It was required 1-2 hours to improve the scenario data which was able to run through from start of the accident to end of the scenario. In this improvement phase, the user could find problems, in which the interaction had stopped unexpectedly from "Sequence Diagram", identify and correct the cause of them in "State Chart Diagram" easily and rapidly. Because the interaction simulation could run in 100 times speed, the process of simulating, analysing and editing can be repeated many times.
(3) Focus on the limitation time and long waiting time: Finding the late execution of the checkpoint task, long postponed task, long waiting time and long traveling time from "Sequence Diagram", the user could investigate the causes from the "State Chart Diagram". Then the user could change the scenario data, assigned number of person and resources and confirm the result by running the interaction simulation. Through the analysis, the user could find potential problems of the scenario or better assignment of person /resources.
(4) Consider failure of machine and trouble in work: The interaction simulation could conduct the conditions such as a certain machine is failure and/or a trouble occurred in a certain work and it consumes long time to complete. Simulating and analysing the interactions under the conditions could introduce more resilient procedures or person against the accident,

The results of the above case study are shown in Table 3 for the steps (3) and (4).

The case 1in Table 3 is original assignment condition, which consists of 2 supervisors, 8 operators and 17 supporters. The checkpoint for starting alternative water injection into core by charging pump is failed to do within the limitation time of 2 h 20 min, it is done at 2 h 44 min 24 sec in the case 1. The cause of the delay is considered that, although the 8 operators are in the main control room at the beginning of this scenario, 7 operators move to the field and only one operator remains, because the one operator can do their tasks one by one, the operator must postpone the delayed task until completion of the previous tasks such as staring forced cooling of secondary system.

To avoid the problem, one supervisor is shifted to an operator in the case 2. The simulation for the case 2 is conducted, which is under the assignment of 1 supervisor and 9 operators and 17 supporters. In this case, all the checkpoints are done before their limitation time, but checkpoint task of judge the accident is delayed compared with case 1. The cause is that, because there are many tasks immediately after the accident by one supervisor in this case, so the completion to judge the accident should be delayed.

In the case 3, only the task for starting alternative water injection into core by charging pump is assigned to supervisor by modifying its procedure, people assignment condition is same with case 1. The

result of the case 3 shows that all checkpoints are successfully finished before the limitation time and there are no delays compared with other cases.

Those case study investigations mentioned above were performed easily and rapidly by the use of plant DiD risk monitor. The process for the investigation was very effective to understand the procedure, person assignment, and potential problems among actors.

Table 3 Results of case study.

Checkpoints & Limitation time	Case 1	Case 2	Case 3
	Supervisors: 2	Supervisors: 1	Supervisor: 2
	Operators: 8	Operator: 9	Operator: 8
Judge the accident (0:10:00)	0:06:10	0:07:29	0:06:07
Start forced cooling of 2nd system (0:30:00)	0:18:04	0:19:43	0:18:00
Supply electricity from alternative generator (1:00:00)	0:37:12	0:38:51	0:37:09
Start alternative water injection into core (2:20:00)	2:44:24	2:00:17	1:38:11
Hot shut down status (4:00:00)	2:44:24	2:33:34	2:33:41
Able to supply seawater to aux feed water tank (11:00:00)	5:00:18	5:01:58	5:00:17
Able to supply seawater to CV recirculation unit (51:00:00)	6:28:39	6:30:19	6:28:38

5 Discussions on further application of plant DiD risk monitor

There are two possibilities by the presented plant DiD risk monitor as discussed in 5.1 and 5.2 in this chapter.

5.1 Tabletop exercise for leadership training

As presented in 4, the plant emergency situation can be simulated by modelling the interaction of plant staffs and plant states as the related actors by the plant DiD risk monitor. The operator actors of plant personnel will act in accordance with the emergency procedure of the plant system. However, the plant DiD risk monitor can accept additional models of human models, *i.e.*, many actors of various organization outside of the plant such as a firefighting team, a police station and so on. This expanded risk monitor system can be depicted as shown in Fig.7.

In this case, example framework the plant DiD risk monitor can simulate co-operation between the different organizations related to the event such as for nuclear emergency drill when radioactive hazard is assumed to affect the surrounding areas outside of nuclear power plant so that the neighboring citizen have to evacuate orderly in accordance with the guidance by police officers, fire fighting team, *etc.*. The procedures of co-operations are confirmed by simulations under various conditions obstructing their actions and they can be sophisticated by using simulation results. The feature of this function can be applied to generating scenarios for a tabletop exercise [12].

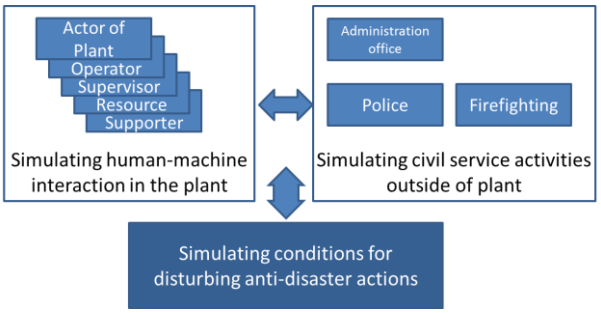

Fig.7 Modelling of relationship between other organizations.

Various information which includes unexpected ones under emergency situations is given to a trainee in the tabletop exercise. The trainee acting as a responsible person is expected to act as calm and cool-headed even in emergency situations with high stress. For this purpose, the appropriate training scenarios of the tabletop exercise will be generated by the following ways: (A)Construct a basic scenario based on the procedure or manual against an emergency. (B)Append other emergency situations on the basic scenario. (C) Simulate the scenario and find and append obstructing points of trainee's actions. According to this process, the emergency situations are not improved even if the trainee performs his correct actions. Then the trainee is forced to his high stress condition.

Furthermore, the plant DiD risk monitor can be used as a supporting tool of the actual tabletop exercise. The training is advanced by trainers who present event**s'** information of the scenario to trainees. The trainers can advance the scenario by generating external events and internal events by pressing buttons through "Event Control Panel" as shown in Fig.8.

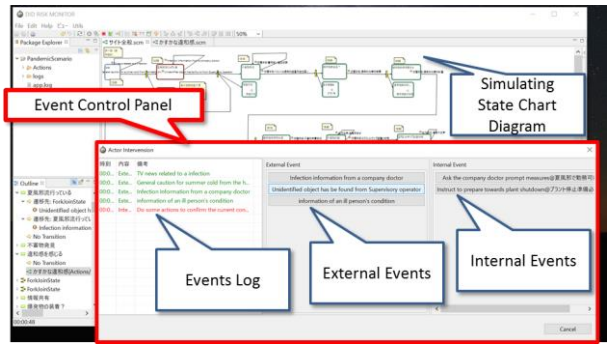

Fig.8 Event control panel of the DiD risk monitor.

The external events represent events given to the trainee as information coming from outside. (For example, Explosion occurred, Request from headquarters of company/government, and so on). The internal events are expected actions of the trainee to prevent the progression of the emergency, or to minimize damages of the plant, employees, resident in the neighborhood and so on. The trainers press the internal events according to the trainee's action. The information of active events with time are recorded as log data in the DiD risk monitor and can be effectively used at a review meeting after the exercise.

5.2 Connecting with a process simulator

Thus far, the presented plant DiD risk monitor system assumes the human-machine interaction simulation by using discrete state transition model. However, application of the system will be further expanded by connecting the risk monitor system with a continuous simulation system.

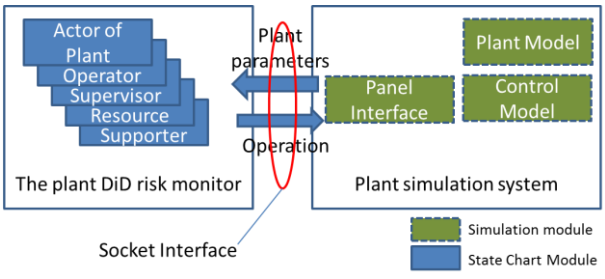

Fig.9 Socket interface between risk monitor and real process.

As is illustrated in Fig.9, a modification for communicating with another application software or simulation system will be realized by using a socket interface [13]. Parameters of actor's "State Chart Module" can be read or overwritten through the socket interface. This allows, for example, constructing an effective test bed for operator training simulators. The tuning work of simulators is a work of trial and error mode by engineers and operators and takes long time. However, in case that the plant DiD risk monitor equipped with plant operating procedures is connected to the plant simulator, the DiD risk monitor acts as operators of the simulator. Then the simulator can be tuned up without actual operators and with a high-speed simulation mode of the simulator. On the other hand, when a completed simulator exists, operating procedures can be verified by use of this test bed.

6 Conclusion

The progress of the author's developmental study on a new risk monitor system was introduced, which can be applied not only to severe accident prevention in daily operation but also to serve as to mitigate the radiological hazard just after severe accident happens and long term management of post-severe accident consequences. Then, the fundamental method was summarized on how to configure "Knowledge-based data" in the plant DiD risk monitor by "State Chart Diagram" and how to describe the interaction among actors by "Sequence Diagram", these diagrams are extensively used in the field of systems engineering.

In this paper, the authors show that by applying the componentize mechanism of the "State Chart Diagram", whole knowledge-based information essential to simulate human-machine interactions can be modeled easily and efficiently. And the "Sequence Diagram" can describe the interactions among actors and express the problems of the interactions well. The investigation method for the integrations is also proposed by searching the problems on the "Sequence Diagram", finding and correcting the cause of the problems from "State Chart Diagram" and conducting the interaction simulation for the corrected data iteratively. The authors confirmed the method could be done easily and rapidly, and the process for the investigation was very effective to understand the procedure, person assignment, and potential problems among actors.

In addition, the plant DiD risk monitor is modified to apply it to other fields of application. The application to generating tabletop exercise scenarios and the

application to communicate with plant simulator as discussed in 5 are under development.

List of acronyms

DiD: Defense in Depth
FMEA: Failure Mode and Effect Analysis
KB: Knowledge Base
NPP: Nuclear Power Plant
PWR: Pressurized Water Reactor
UML: Unified Modeling Language

References

[1] YOSHIKAWA, H., *et al* : Configuration of risk monitor system by plant defense-in-depth risk monitor and reliability monitor, Nuclear Safety and Simulation, Vol. 3, No 2, pp.140~152. .2012.

[2] YOSHIKAWA, H., *et al.:* A new functional modeling framework of risk monitor system, Nuclear Safety and Simulation, Vol. 4, No. 3, pp.192~202, 2013.

[3] YOSHIKAWA, H., *et al*.: Integrated functional modeling method for configuring NPP plant DiD risk monitor and its application for AP1000, (ICONE22-30987) Proceedings of the 22nd International Conference on Nuclear Engineering (ICONE22), July 7-11, 2014, Prague, Czech. .2014.

[4] YOSHIKAWA, H., *et al*.: Integrated functional modeling method for configuring NPP plant DiD risk monitor and its application for conventional PWR, Proc. ISOFIC/ISSNP2014, Aug.24-26, 2014, Jeju Island, Korea, 2014.

[5] YOSHIKAWA, H., and NKAGAWA, T.: Software system development of NPP plant DiD risk monitor -basic design of software configuration-, (ICONE23-1312) Proceedings of the 23th International Conference of Nuclear Engineering (ICONE23), May 17-21, 2015, Chiba, Japan, 2014.

[6] MATSUOKA, T.: System Reliability Analysis Method GO-FLOW for probabilistic Safety Assessment, CRC Sogo Kenkyusho, 1996. (In Japanese).

[7] HASHIM, M. *et al*.: Dynamical reliability analysis for ECCS of pressurized water reactor considering the large break LOCA by GO-FLOW methodology, Nuclear Safety and Simulation, Vol. 3, No. 1, pp. 81~90. 2012.

[8] HASHIM, M., *et al.*: Addressing the fundamental issues in reliability evaluation of passive safety of AP1000 for a comparison with active safety of PWR, Nuclear Safety and Simulation, Vol.4, No.2, pp.147-159, 2013.

[9] Eclipse foundation: Eclipse IDE for Java Developer Version: Luna (4.4.1), http://www.eclipse.org, 2014.

[10] Eclipse foundation: GEF (Graphical Editing Framework) Release 3.9.101, https:// eclipse.org/gef, 2014.

[11] Object Management Group: UML Version 2.4 Specification, http://www.omg.org/spec/UML/2.4, November 2011.

[12] Guidance for a Large Tabletop Exercise for a Nuclear Power Plant, NUREG-1514, 1995.

[13] IEEE Std. 1003.1-2008 Standard for Information Technology- Portable Operating System Interface (POSIX(R)