MODELING, IDENTIFICATION AND CONTROL, 1993, VOL. 14, NO. 2, 85–91
doi:10.4173/mic.1993.2.3

ISACS, an integrated surveillance and control system

JON KVALEM†, ROLF-EINAR GRINI† and KJELL HAUGSET†

Keywords: *System integration, control room systems, expert systems, process databases.*

In this paper we report about application requirements and design principles used in the development of an integrated surveillance and control system, intended to be used by the operator of a nuclear power plant. The system, for its own nature, is an example of a class of applications normally referred to as critical applications, and involve the use of both expert system and database technologies in a real-time process control environment.

1. Introduction

Information analysis and presentation are increasingly important activities in modern control rooms of industrial processes and power plants. The introduction of different Computerized Operator Support Systems (COSSs) in the control room, calls for a new approach in handling and presenting information to the operator. To deal with these problems, the project on the Integrated Surveillance and Control System (ISACS) was initiated at the Halden Reactor Project (Haugset *et al.* 1990a, b). The ISACS concept is of general nature in industrial process and power plants, but our laboratory implementation is specialized towards nuclear power plants. In ISACS, information from several COSSs is collected, grouped, and presented in a uniform way. All visual information is presented to the operators on colour graphic CRTs.

ISACS consists of a number of modules, and of these the expert system called Intelligent Coordinator (IC) and the Common Database play central roles (Yamane and Grini 1991, Kristiansen, Kvalem, Kvalem and Swanberg 1991). The IC receives input from the different COSSs, from the process, and from the operator. Sitting with the extract of available information, it can offer a concentrated overview of the status of the plant. In addition, it has access to detailed information which can be brought up on request.

The database management system of ISACS was required to be flexible in its handling of data. Due to the heterogeneous types of data which is included in the database, e.g. process related data, data from operator support systems, man–machine interface data etc., and to ease the integration of the different parts of ISACS, the use of a commercial system, with the necessary flexibility, seemed to be the solution with the most favourable cost–benefit ratio.

A demonstration version of ISACS, including the Intelligent Coordinator and the Common Database, is developed, preliminary tested in the Halden Man–Machine Laboratory and found satisfactory for a previously specified scenario. The nucleus of

Received 20 September 1992.
† Institutt for energiteknikk, OECD Halden Reactor Project, N-1751 Halden, Norway.
An early version of this paper was presented at the 2nd International Conference on Database and Expert Systems Applications DEXA '91, August 21–23 1991, Berlin.

the Halden Man–Machine Laboratory is a full-scale nuclear power plant simulator, which acts as the process in the laboratory implementation of ISACS.

As an example of other work related to the ISACS project, see Hollnagel (1990). Introduction of a control room concept like ISACS will strongly influence the operators role, as new types of information are made available to him. New support will be given to the operator both in the phase of process status identification, such as diagnosis of disturbances, in action planning, and in implementation of control actions. Experiments are being conducted to measure positive and negative effects of this 'soft automation' on operator performance.

2. System structure

The multi-computer system which forms the basis for the implementation of the ISACS system, is a multi-vendor network-based system integrating conventional and knowledge processing capabilities.

It was required that ISACS was developed in such a way that optimal flexibility and modularity were maintained. It should be possible to:

remove parts of ISACS for integration into other environments than the Halden Man–Machine Laboratory
remove one or several COSSs from an ISACS environment and integrate with other systems
easily include new COSSs into ISACS
easily expand ISACS with new functions.

To meet the requirements specified above, it is essential that:

all communication between tasks in ISACS is message based
the interfaces between the different subsystems within ISACS are strictly defined
data are available in accessible databases.

The overall structure of the ISACS system, including data flow, is shown in figure 1. The real or simulated process data are continuously being updated in the Process Database, and hence made available to COSSs and the MMI. Some COSSs are running on a continuous basis, e.g. alarm filtering and fault detection systems, while some are triggered at events, e.g. diagnosis and prognosis systems. All COSSs access the Process Database for information and their produced information is delivered to the Common Database, where it can be accessed by the Intelligent Coordinator and the MMI. All information produced by the Intelligent Coordinator for presentation in displays is directed to the MMI via the Common Database.

3. ISACS databases

The use of commercial database systems in process control environments, as in ISACS, is not very often seen. Usually, commercial relational database systems are used in administrative application areas, while in the technical application area one, to a large extent, relies upon home-made solutions.

The wish for a system which has a uniform interface to the applications and a system which is, as far as possible, data independent, motivated an examination of relational database systems. Three items were given special focus when choosing the relational database system for use in ISACS. Those items were: performance, distributed processing capabilities and the existence of a well-defined interface to

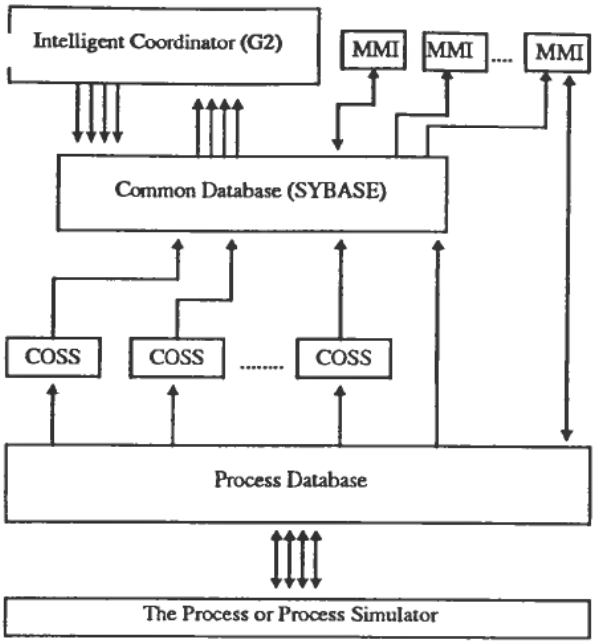

Figure 1. Overall structures of ISACS.

applications written in conventional programming languages, such as C or FORTRAN. A careful study of relational database management systems resulted in the use of SYBASE for implementing the ISACS Common Database (SYBASE Inc., 1989).

The nature of a relational database management system implies that handling cyclic updates of large amounts of measured data, as is the case in process control applications, is hard. Due to these limitations, ISACS requires a process database which is capable of handling cyclic updates of process signals and is capable of providing applications with updated values of process signals. In this way the process database will contain all process related signals, while the relational system will contain event-based information and all non time-critical information.

3.1. *The Common database*

The Common Database in ISACS is the main repository of data to be shared by the different tasks. To avoid a tight coupling between tasks, a solution was chosen to let tasks deliver data to one central data pool, the Common Database, while other tasks are allowed to read the required data from this data pool. The data will typically be:

a small selection of process data
alarms from the different alarm systems
diagnosis and prognosis data
man–machine interface data
operator actions.

The different COSSs transfer all relevant data to the Common Database, from which the Intelligent Coordinator can access the data which it finds interesting at all times. In addition, the COSSs' data which are relevant for presentation in displays, can be accessed from the Common Database by the display systems.

Access to the Common Database by applications, e.g. COSSs, the Intelligent Coordinator and the MMI systems, is performed using the SYBASE functional library Open Client. As opposed to the embedded philosophy, the Open Client functions are called by applications and requires no pre-compilation. SQL-statements are given as parameters to these functions (SQL 1986).

To avoid the interpretation of SQL procedures for each data-base access, the use of Stored Procedures, i.e. pre-compiled SQL procedures stored in the data dictionary, is extensively used in ISACS. By simply passing parameters to procedures ready for execution, performance is substantially enhanced. Specific SYBASE facilities for making high speed copying of bulks of data into the database is also being used to enhance performance further.

3.2. *The Process Database*

The Process Database in ISACS is an in-house made memory-resident database based on the FORTRAN COMMON block principle. The process simulator delivers updated signals at a 1·5 seconds fixed interval. The number of signals which are being updated every simulator cycle varies between a small number and several thousands, depending upon the changes in process state, since only changed signals are being updated.

The access to the Process Database takes place using a specially designed access library. This library provides applications with functions for both fetching and placing data from/into the Process Database. All process signals are identified by a tag name, and a specially developed hashing algorithm is used for converting the tag names to unique database indexes. In this way process signals are accessed by direct reference.

A small selection of key process signals are, at pre-defined intervals, transferred from the Process Database into the Common Database. In this way they are made available to the Intelligent Coordinator and can be used for coordination and plant status identification purposes.

4. The Intelligent Coordinator

The Intelligent Coordinator (IC) continuously supervizes messages coming in from COSSs. The contents of the messages are used to analyse current plant status which is represented by several specific parameters defined for ISACS. The IC activates passive COSSs when necessary, to get further information such as diagnosis, prognosis, or procedure recommendation. Another important aspect of the IC's role, is to interpret operator actions. When requested by the operator, the IC must produce the requested information, coordinate the additional assessments and report the results.

When a COSS reports a planned or unexpected plant transient, the IC defines it as an event represented by an object in the knowledge base. All new information from the same or any other COSS, related to the same transient, will be tied to the already existing event. COSSs related to alarm, diagnosis and prognosis systems provides a process area coding which follows the message from a COSS. This is used by the IC to decide whether to group information from different COSSs in the same event or not. Another type of events is function oriented events which are based on information from critical function and success path monitoring systems. Information related to a certain success path will be grouped in the same event as information related to the corresponding critical function which is threatened or violated. If a message from a COSS cannot be related to an existing event, a new event is created. This way there may

be several events present at the time, each representing a transient in the plant. The IC prioritizes between the events to point out to the operator which are the most serious and should be concentrated on. A plant state will be defined, and recommendations for the operator about what to do to mitigate a failure will be given. Information created inside the IC will, in addition to data from the COSSs, be grouped and sent to the MMI.

The current prototype implementation of the IC is based on the COSSs currently available in ISACS, and on the MMI display specifications. Consequently, parts of the IC will have to be re-coded if the selection of COSSs is changed, or if another set of information should be displayed for the operator. The modular implementation of the knowledge base will ease such changes. The basic philosophy behind the event concept will be preserved.

G2 has been chosen as the implementation and development tool for the IC (Gensym Corporation 1989). This is an expert system shell, specially constructed for real-time applications. It provides facilities for including the time aspect in the reasoning. This was very important when choosing G2, because of the dynamic character of the application. The implementation is object-oriented. Rules are written in a special G2 language, which also includes the use of procedures. An extensive use of WHENEVER ⟨ ⟩ DO ⟨ ⟩ type rules, results in an event based reasoning where rules are triggered only under certain circumstances. This is saving computer time compared to interval based triggering. The implementation of the IC is effected inside one knowledge base. The rules are organized in workspaces, where each workspace is related to one task or contains a collection of related items. The workspaces are organized in a hierarchy with one top-level workspace and subworkspaces, see figure 2. There are up to four levels in the hierarchy and a total number of approximately 60 workspaces. The hierarchy gives a well organized knowledge base in which it is easy to navigate through the workspaces. The maintenance of the hierarchy is made easier through the division into subworkspaces, as each subworkspace contains the knowledge associated with a specific function in the IC.

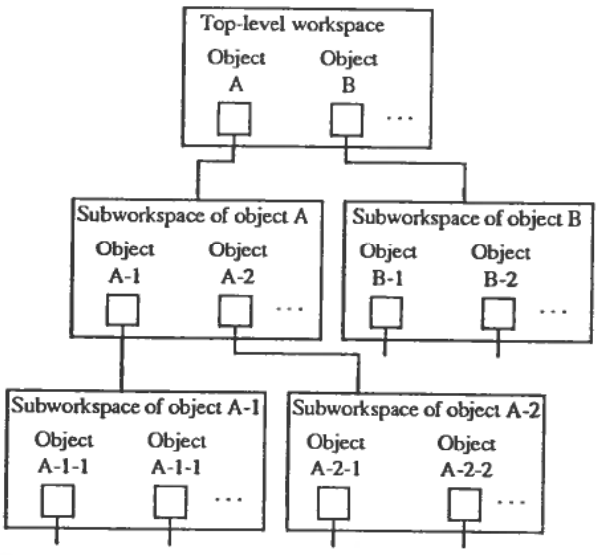

Figure 2. Workspace hierarchy of the Intelligent Coordinator.

The interface towards the Common Database, is defined through sensor objects in G2. A sensor represents a variable to be transferred or received. Attributes in the sensor object defines a location in the Common Database. The values of the attributes may be static, which leads to one-to-one correspondence between a sensor and a location in the Common Database. However, the attribute may vary, which enables use of the same sensor to send values to different locations. The integration of the G2 Standard Interface and the SYBASE Open Client provides for an efficient way of communicating betwen the Intelligent Coordinator and the Common Database.

5. The Man–Machine Interface

ISACS is intended to act as a single, integrated interface for the operator for all operational situations. As such, all information from the process, and all commands to the process will be passed through ISACS. Therefore, the design of the ISACS MMI is critical, as the operator's ability to interpret and control the process is entirely dependent on the MMI and its underlaying software.

The ISACS-1 MMI consists of a total of thirteen colour graphic screens. Five of them are showing ISACS driven overview information, while the rest are available to the operator for interaction with the process and the individual COSSs. Operator interaction with ISACS are taking place by using a context sensitive dynamic keyboard.

6. Conclusions and status

A prototype, ISACS-1, has been made of the Integrated Surveillance and Control System, in which two major parts are the expert system named the Intelligent Coordinator, implemented in G2, and the Common Database, implemented in SYBASE. The choice of G2 has proved to be successful, with its capabilities for reasoning with time as a parameter. Communication between G2 and SYBASE has been realized and works satisfactorily through the use of the G2 Standard Interface package. One shortcoming, though, is that GSI does not permit external systems to set values in G2. This forces us to make G2 do time consuming polling in SYBASE looking for changes in the data.

ISACS belongs to a class of computer based systems with very little proven experience from the point of view of how to use traditional database technology. This is mainly due to the fact that in process control environments where the demand for real-time response is essential, home-made database solutions have been used in almost all situations. The reason for introducing a relational database in an environment like ISACS, was the need for a flexible and modern tool for handling the heterogeneous types of data which are present in the system. The preliminary experience, although thorough system testing has just been initiated, is rather good (Fält, Jensen, Kvalem and Kvalem 1991).

References

Fäit, C. O., Jensen, L. S., Kvalem, I. and Kvalem, J. (1991). Database Technology in Industrial Process Environments. OECD Halden Reactor Project Technical Report, HWR-291.

Gensym (1989). G2 User's Manuals, 1989, Gensym Corporation, USA.

Haugset, K., Berg, Ø., Bologna, S., Evjen, O., Nelson, W. R., and Yamane, N. (1990a). ISACS-1 motivation, general description. OECD Halden Reactor Project Technical Report HWR-265.

Haugset, K., Berg, Ø., Fordestrommen, N. T., Kvalem, J., and Nelson, W. R. (1990b). ISACS-1, The prototype of an advanced control room. *IAEA International Symposium in Balancing Automation and Human Action in Nuclear Power Plants.* Munich, 9–13 July 1990.

HOLLNAGEL, E. (1990). The design of integrated man–machine systems and the amplification of intelligence. Paper presented at the *International Conference on Supercomputing in Nuclear Applications*, March 12–16 1990, Mito City, Ibaraki, Japan.

KRISTIANSEN, L. I., KVALEM, I., KVALEM, J. and SWANBERG, B. (1991). ISACS-1 System Description. OECD Halden Reactor Project Technical Report HWR-290.

SQL (1986). American National Standard, ANSI X3.135.

SYBASE User's Manuals (1989). SYBASE Inc., USA.

YAMANE, N. and GRINI, R. E. (1991). The Intelligent Coordinator Module of the Integrated Surveillance and Control System (ISACS-1). OECD Halden Reactor Project Technical Report, HWR-289.