1

MULTIMEDIA CAPABILITIES IN DISTRIBUTED REAL-TIME APPLICATIONS

Subhabrata Sen, Oscar González, Krithi Ramamritham, John A. Stankovic, Chia Shen* and Morikazu Takegaki**

Department of Computer Science
University of Massachusetts Amherst, MA 01003

** A Mitsubishi Electric Research Lab*
*** Mitsubishi Electric Corporation*

ABSTRACT

In this paper, we consider the problems and benefits of developing the infrastructure needed to handle the timing requirements of a real-time industrial control application and the processing of multimedia streams. In particular, this paper explores various application scenarios which can benefit from a real-time control system integrated with multimedia. We propose a high-level architecture based on open systems and identify key research issues at the system and database levels that must be addressed in order to incorporate multimedia capabilities into a distributed industrial control application. A real-time multimedia database is a key component in this platform and it can aid both in real-time control and in the decision making process at the management level, potentially resulting in reduced costs and improved operation of an industrial plant.

1 INTRODUCTION

Distributed real-time industrial control applications can benefit significantly from the incorporation of multimedia information into the control environment. Potential benefits of such systems include enabling more intuitive visualization of the state of the control process, resulting in improved quality and/or safety of the process, better decision making at both the management and supervisory operator levels, as well as distance collaborative planning, design and effective trouble shooting. Thus, the processing and analysis of different types of multimedia information provides a wider range of capabilities to different entities (e.g., operation, maintenance, training, and administration) within a company than is possible now. Recent advances in computer technology, high speed networks, high capacity storage and innovative sensors make it possible to realize such enhanced control systems. The goals of this chapter are to examine real-time control applications that can exploit multimedia capabilities so as to determine their requirements, to identify key research issues at the system and database levels, and to propose a generic architecture for these applications.

1.1 Inadequacies of Current Practices

When we study ways to add multimedia capabilities to today's control systems, we quickly realize that modifications or extensions to a large number of current distributed plant control systems is not feasible. These systems have been built mostly with proprietary network components and host systems, and real-time constraints are partially met through physical isolation of sub-networks. In order to guarantee the plant control timing constraints, the workstations supporting the human interface or plant information management often do not reside on the same physical network as the network in the controlled plant site. Recently, limited multimedia (particularly video) capabilities have been incorporated in some industrial plants by the addition of a separate physical network for video transmission from industrial TV cameras.

Figure 1 portrays a typical example of current industrial control systems [7, 6]. The network architecture contains two layers, a control layer and an information layer. The network bus in the control layer is FDDI-compliant, connecting together proprietary subsystems such as programmable logic controllers (PLCs), and operator stations (OPs). A proprietary protocol, called *cyclic transmission*, is used to achieve guaranteed end-to-end delays in the control layer among the subsystems. Under this protocol, each subsystem in the control layer has a fixed amount of time within a periodic cycle to broadcast information. The information layer employs standard general purpose components for low cost. This layer contains an Ethernet LAN and a number of Engineer Workstations (EWS). The EWSs handle plant information management, plant maintenance, and report generation.

Some of the problems with such a two layer system architecture are:

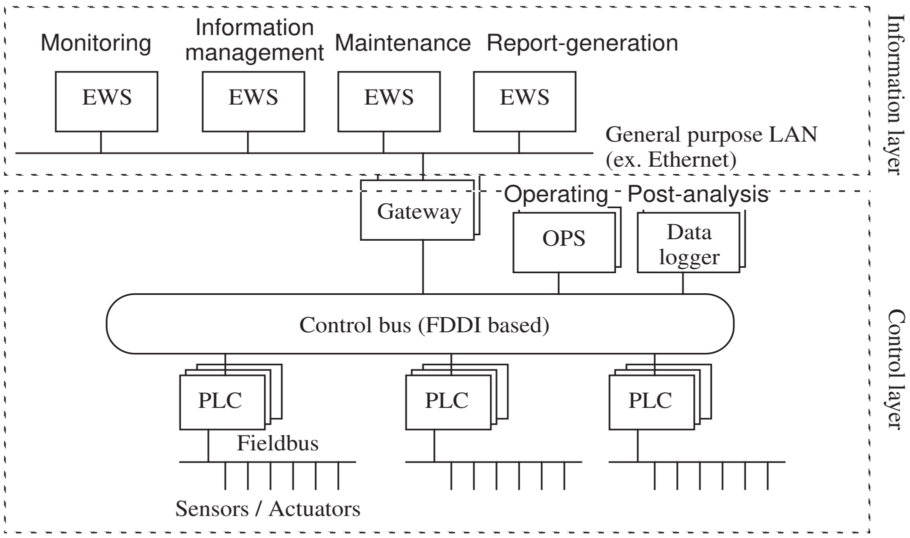

Figure 1 Architecture of Current Industrial Control Systems

- Fixed Quality of Service (QoS): The QoS is fixed by the cyclic transmission period and cannot be altered according to dynamic application requirements.
- Hierarchically layered network: Gateways are employed to connect the information layer and the control layer. The gateways can be a system bottleneck for performance and reliability.

Additional inadequacies of such an architecture are pointed out in [7]. Consequently, these distributed plant control systems are inflexible, expensive, and difficult to modify.

The integration of multimedia and traditional real-time applications has been attempted as part of the Mercuri project [3] where data from remote video cameras is transferred through an ATM network and displayed using X windows, but they provide only best effort services. The Presto project [4] deals with providing session-based QoS guarantees to continuous multimedia database applications, but does not address the coexistence of control data with continuous media data. The explosive growth of the Internet has spawned applications like Vosaic [14] and Realaudio [13] which deliver multimedia information to the remote user. The integration of different media is typically done or defined offline and the data delivery is geared towards best effort or probabilistic types of service guarantees. These and other similar developments in the commercial arena are unlikely to address the problems arising in real-time control environments due to the existence of *critical deadlines*, predictable end-end performance requirements, and good performance in crisis and overload situations.

1.2 Desired Features in an Integrated Multimedia Real-Time System

The above discussion indicates that what we need is an integrated platform supporting hard real-time control tasks, multimedia data, and a distributed data repository, built over a common set of computational resources. This approach will help produce a scalable and cost-effective solution for realizing the benefits discussed earlier. The platform must have the ability to seamlessly manage large quantities of multimedia information, and allow *intermedia integration* – the coupling of various media sources with real-time information – while providing *end-to-end system guarantees* for the application's timing constraints. Due to its technical advantages (namely, high bandwidth, end-to-end individual QoS, virtual connections and flexible topology) an ATM network is a suitable candidate for the main backbone of this architecture. The ATM network will carry sensor data to the decision support system and data repositories, and carry command and control data to the sensors and actuators.

To effectively support real-time interactions between different entities in a company, a distributed real-time database with multimedia capabilities is vital. This database can act as a repository for the different types of information collected at various levels of the control hierarchy. It will allow the retrieval of historical information as well as real-time state information (traditional sensor-based or multimedia) under timing constraints. When a query is made to this database, it may interact with several tools to aid in the decision making process at various levels of management. The presence of multimedia data implies that the database must have support for content extraction and content organization as well as for querying multimedia content. These are challenges in their own right, but accomplishing them in a time-constrained environment demands new approaches to these problems.

The rest of this chapter is organized as follows. Section 2 describes various application scenarios that will benefit from the incorporation of multimedia information into a real-time control environment. Section 3 proposes an architecture for an integrated real-time multimedia control platform. In addition, it describes a real-time multimedia database and identifies key research issues. Section 4 summarizes the ideas presented in this chapter.

2 USES OF MULTIMEDIA IN A REAL-TIME CONTROL ENVIRONMENT

This section contains various application scenarios that will benefit from the incorporation of multimedia information into the control system.

2.1 Operator level

The operator of modern industrial plants is currently overwhelmed with the amount of information presented to him. As more information is integrated with control systems, the question of how a human can keep up with the flow of results becomes critical [2]. Expert systems for alarm processing have been implemented to aid the operator when an overflow of alarms occurs. Recently, experimental visualization techniques have been developed [2] with the purpose of enabling a human to understand the output of the control system in a more intuitive way.

In industrial plants, video and audio allow the operator to *confirm* the state of the process [8]. For example, a sensor reading a low fluid level at a storage tank can be confirmed with an image of the tank. Similarly, sensors or equipment can easily be identified as faulty using images. Also, video images can help an operator determine whether it is *safe* to start the operation of a particular equipment in areas where maintenance or operations are being conducted.

2.2 Management level

One way to improve the decision making process at the management level is to allow a high degree of interactivity between the manager's analysis tools and the real-time information collected by the control system. For example, the real-time information can be utilized to drive and maintain a three dimensional model of the plant. This will allow a management team to develop and evaluate alternative plant designs before conducting additions, repairs or modifications to the plant, possibly in real-time. For each alternative, the designer can select different equipment options and combine it with its historical performance profile to evaluate the operational and maintenance costs of that particular alternative. Also, evaluation of the equipment and actions taken by human operators under crisis conditions can help in post-mortem analysis. Information from various multimedia sources can help a review team evaluate the performance of the control system and the operator during a crisis situation.

2.3 Maintenance

The integration of new high-tech sensors and diagnostic equipment in modern control systems will provide a maintenance engineer with the ability to monitor equipment conditions directly. In addition, further analysis of the information can prevent costly equipment failures, allow more timely maintenance decisions and reduce operational and maintenance costs [1].

In order to automate the maintenance process, a real-time multimedia database is essential. The control system collects the information from the diagnostic equipment and sends it to a higher level in the control hierarchy for proper interpretation and analysis of the sensed state of the plant components by qualified personnel. In addition to the process data, the information collected at the plant can include ultrasound, thermography, X-rays, and video.

Electric companies are in the process of evaluating and implementing predictable maintenance programs for fossil power plants. In one such program, periodic and non-periodic diagnostic information from 40 systems is combined with other data to determine maintenance requirements on the basis of equipment conditions. One particular example focuses on preventing unexpected failures in large electrical motors. Thermographic images collected using infrared cameras are used to detect hot spots that may indicate clogged air filters, abnormally high current flows, or elevated bearing temperatures [1].

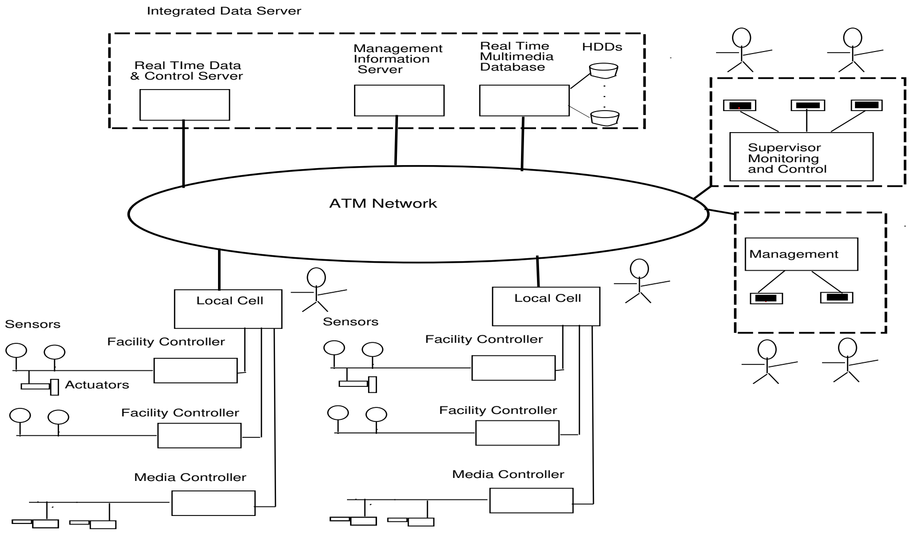

Figure 2 System Architecture

3 DESCRIPTION OF FUNCTIONAL UNITS IN THE DISTRIBUTED ARCHITECTURE

This section presents a distributed system architecture for handling the integrated multimedia and real-time control processing requirements of a wide range of current and future real-time applications, such as distributed industrial control systems. The major components of the architecture are a number of Local Cells (LC) – distributed throughout the plant, an Integrated Data Server (IDS), a Supervisory Monitoring and Control Module, and a Management Module. Each Local Cell (LC) covers the sensors and actuators in a certain part of the plant. The IDS is made up of the following subsystems, each of which is implemented in a *distributed* manner: Real Time Data & Control Server (RTDCS), Management Information Server (MIS) and a Real-Time Multimedia Database (RTMMDB). The IDS serves as a global repository of plant information and has capabilities for QoS negotiation and admission control which are used by the other modules to achieve their functionality. The different parts of the architecture are connected via a high-speed ATM network. Figure 2 shows a graphical representation of the architecture. The following sections describe the functionality of each of the components in more detail.

3.1 Local Cell

A Local Cell has information gathering devices (video cameras, ultrasonic equipment, infrared cameras, sensors, etc.) which it controls via some Facility Controllers (FCs) and Media Controllers (MCs). The information from these sources constitute the LC's world-view. The LC acquires video and other sensor information, and displays the individual video streams on the screen for the local operator(s). The Local Cell could either use the video as an auxiliary information source or as an integral part of the sensor data. Depending on the application, sensor processing and fusion may be done, and updated local world models may be maintained in databases local to a cell. The LC sends out video streams to the IDS. It could also require and request information from the IDS. Control information may flow from the supervisory module to the LC.

Facility Controller

Each Facility Controller is a relatively simple controller and has at its disposal, a set of sensors and actuators with which it interacts based on control commands coming, for example, from the IDS or due to built-in internal logic. A Programmable Logic Control (PLC) or fuzzy logic controller falls under this category. Speed, reliability and timely responses under tight temporal constraints (in the order of milliseconds) remain the overriding performance requirements from these controllers. These, in turn, place limits on the sophistication of the algorithms running here. The FC collects and examines the sensor data and performs tight-loop reactive control using the actuators at its disposal. It also forwards its collected *state of the world* to its parent Local Cell.

Media Controller

This unit controls the operation of a number of cameras of different types (visual, infra-red etc.). It switches the cameras on and off based on control commands coming from higher layers or due to built-in internal logic. At any instant, input from only a subset of the operating cameras may be used. The frames are sent to the parent LC (perhaps in compressed format) for storage and/or simple visualization.

3.2 Integrated Data Server

The Integrated Data Server (IDS) obtains sensor and multimedia (video, thermographical images, X-rays and ultrasound images) information from multiple Local Cells (LC), collates them, updates its *global view* and stores the relevant information. Some of the capabilities in the IDS may also be present in the Local Cells. The IDS can be thought of as consisting of three interacting subsystems: Real Time Data & Control Server (RTDCS), Management Information Server (MIS) and a Real Time Multimedia Database (RTMMDB). These are described in the following subsections. In a real industrial environment, the functionality of each of these subsystems will be distributed across different entities in the network.

Real Time Data & Control Server (RTDCS)

The remote operators interact with the RTDCS to get real-time control information and to provide real-time feedback to the Local Cells. The RTDCS collates real-time data coming from the Local Cells and provides information to the Supervisory Monitoring and Control operators. It also mediates QoS negotiation and end-end admission control between the different information source and receiver entities. It performs intermedia integration either dynamically, on request, or when storing related data in the real time multimedia database.

The RTDCS has to support a wide range of hard and soft real-time types of processing. It is responsible for gathering information streams from multiple LCs, and using the information to update its global view of the system state. This information will also be stored in the RT Multimedia database. The incoming state information can be utilized for maintaining a 3D global model of the factory which will give remote operators and management personnel a more realistic feel for the state of the industrial plant. For example, consider online annotation operations on live images as part of information filtering whereby, e.g., some important features may be highlighted and tracked as their coordinates change with time. Information from the detected image and other more traditional sensors will allow updating the global 3D model. If this model is to be used for online decision making, the workload associated with the updates have to be handled in real-time.

The RTDCS must be capable of handling multiple incoming live video streams (depending on the requirements, the processing of the streams may be done online or offline). It is possible for video feeds to be started dynamically (a camera may be turned on). This implies that the workload imposed by online processing may not be uniform and admission control and scheduling support must be able to deal with this situation.

Broadly speaking, three types of data and processing need to be supported in RTDCS:

1. Processing low volume traditional sensor-based (e.g., temperature) information arriving from the local acquisition units (LCs). The worst case processing requirements per period are known for normal situations. The number of such sensor streams is normally fixed for a given operating mode and a given mode may last a long time.
2. Processing multimedia streams coming from infrared, ultrasonic and visible wavelength cameras. The data workload arrives periodically (e.g., 15 frames a sec.) and is high volume even for the compressed case. Data arriving per period may be fixed (uncompressed) or varying (compressed). The corresponding computation requirements are characterized by variable and intensive processing requirements per period. For example, if the current image is very different from the last one, much more processing needs to be done to detect a particular object. The number of multimedia streams which are active at any time will vary.
3. Processing reactive tasks with real-time deadlines initiated in response to incoming state information. These tasks are dynamically activated and the resulting instantiation may be a one-time activity or a periodic one. For example, if an out of range sensor value is detected or an unexpected image is seen in a video stream, or some abnormal system state is derived from sensor fusion of *standard* sensors and audio/video sensors, the system will react to it by initiating some activity to handle the cause of the abnormality.

Given the large number of sensors and video cameras likely to be used in a plant, more often than not, an operator only needs data from a small subset of the sensors or cameras. This calls for dynamic resource allocation, i.e, admission control and scheduling. The admission control services should be capable of evaluating if there are enough system resources (OS and network) to accommodate the requested QoS, given the current system commitments and the importance of the new session. Based on this schedulability analysis, the required resources are reserved for the session which is then admitted, or if the QoS cannot be guaranteed, a scaled back QoS may be negotiated. The admission control mechanism will prevent over-allocation of system resources and allow predictable performance through controlled sharing of resources.

Possible global resource allocation policies include:

1. *Admission control on a per operator basis.* Reserve enough resources per operator so that the normal operating mode workload can be supported. Based on the system specifications, it may be possible to define the normal operating mode for any area - and specify the corresponding system resource requirements. Part of the system resources could be used as a common pool - operators requiring extra resources (say to operate one more camera) would compete with each other for this pool.
2. *Admission control on a per session basis.* The advantage is that operators use only those resources which are needed. The disadvantage is that one operator can monopolize all the resources.
3. *Statically partition the entire system resources among the operator domains.* If enough resources are available, it may be possible to account for the worst case resource requirements for each operator station. This will result in a highly underutilized and costly design. Otherwise, we can still have a tentative partitioning of resources among the different operators with a particular operator given priority within his allocated resources.

A quality-of-service (QoS) guarantee is a promise to provide a certain level of "performance" to an application. Quality of service may be defined in a number of ways even within the same system. Hence, it is important that the OS be flexible enough to accept different types of QoS requests. Further, while there has been a lot of discussion about quality of service there has been too little work in OS mechanisms and algorithms that can actually provide the necessary variations of this service. Solutions need to be developed that enable real-time control and multimedia applications to satisfy predictable end-end performance according to user provided QoS parameters. To do this there is a need for an integrated QoS framework encompassing the application, the OS, the I/O system (e.g., mass storage), the database system, and the communication network.

Appropriate QoS degradation policies are required to allow graceful QoS degradation from the application point of view. The notion of priority in a global context will help in determining how resources may be preempted from existing activities to start an important new activity.

Important research issues in the design of the RTDCS are:

- What are the system level mechanisms needed to support the implementation of each policy?

- How does the system seamlessly support the simultaneous operation of two or more policies? Different resource allocation and scheduling policies may be more suitable for different data types. The mechanisms implementing the different policies must be aware of and cooperate with each other for access to the shared system resources.
- Which admission control and resource management policies work better for each of the workload types?
- How should the system resources be partitioned between the different workload types?
- How should the system be structured to allow graceful degradation from the application point of view?

In designing system support for multimedia real time control applications, some issues which need to be addressed are:

1. *Service Specification:* A well defined application specific service interface is needed, through which the user specifies desired performance requirements (might be a range of choices) for a particular application. An application specific translation service should be able to map these to the specification of system level services (OS and network) necessary to provide the desired performance quality. The application-to-system service interface should be expressive enough to support the accurate specification of a wide range of QoS requirements ranging from deterministic to statistical to predictive to best-effort. A challenging research problem is designing a generic specification language which is powerful and flexible enough to accommodate the different QoS requirements of real-time industrial control applications. Also the translation operation between high-level to low-level QoS requirements and vice versa is nonunique, nontrivial and needs exploration. How to specify and deal with the negotiation aspects of quality of service is another open issue.

2. *Dynamic and Adaptive QoS Management and Maintenance:* Once a session is admitted based on some admission control policy, it must continue to receive its contracted QoS throughout its lifetime. A continuous, systemwide monitoring function is needed to ensure that applications are receiving the guaranteed service. The monitoring will also enable quick reaction to unexpected events (e.g., to misbehaving applications and overloaded situations). We may need to override previously admitted sessions or compute tradeoffs to maximize the likelihood of success of the overall

mission. This might be accomplished by a renegotiation process or by automatic reactions because of tight time constraints. This needs to be part of the runtime platform and interact with any renegotiation that might take place as part of crisis management.

3. *Scalable scheduling techniques for handling the real-time multimedia data:* The scheduling algorithms should perform well in the extremely dynamic operating environment of an industrial workplace which is characterized by large and varying number of tasks and varying task arrival rates. Changing operating modes may trigger changes in the admission control and resource allocation policies and the scheduling algorithms must gracefully adapt to the changed requirements.

4. *Reducing context switching and locking overheads through appropriate process modeling:* For example, given the processing requirement of 30 frames a second, consider the arrival process P as a producer and the image processing entity C, as the consumer of frames. Then, C could be modeled as a periodic process with different period lengths which satisfy both the arrival rate and the acceptable end-end latency, e.g., 33 ms, with each periodic instance processing one frame, or, with period 1 second, with each instance processing 30 frames. As the period length increases, more frames are processed by each periodic instance and so the the context switch overhead per frame decreases, and the ability to mask input jitter increases. This also decreases the number of times locks on shared resources are acquired and released, but increases the granularity of the locks. Also the buffer requirement per periodic task instance goes up as the period length increases.

5. *Handling high bandwidth requirements of multimedia streams:* In practical situations, a large number of high volume multimedia streams may be coming into the RTDCS over the high speed network. In soft real-time environments, compression is used to reduce the bandwidth requirements per stream. However, this approach will require decompression at the RTDCS, and will introduce additional delays (compression, decompression) into the data path. The overheads due to concurrent decompression of multiple streams, as well as techniques for reducing the rate variability of compressed multimedia data to allow more sessions to be supported simultaneously need to be investigated.

Some of the issues presented here are being investigated as part of ongoing research at A Mitsubishi Electric Research Lab and Mitsubishi Electric Corporation on the MidART middleware architecture for networked real-time industrial systems. MidART [7] provides a set of real time network services and industrial application specific but network transparent programming abstractions, and application programming interface. For example, the *selective real-time channels* abstraction has been proposed for supporting on-demand transmission of plant data and utilizes network resources efficiently.

Management Information Server

Management personnel will interact with the MIS for all their information needs. Users in this domain will typically require soft real-time guarantees. The managers will want history and status information, and will typically deal with older data. Occasional QoS degradations will be tolerable here as the operations involved do not have hard deadlines and are not essential to the correct working of the plant. Functionally, the Management Information Server (MIS) is to the management domain what the RTDCS is to the supervisory maintenance and control domain. The MIS will have a sophisticated query processor, a search and retrieval engine to interact with the RTMMDB and other data repositories, and tools for filtering and analyzing the requested data. While the MIS is logically centralized, it can be physically distributed.

Database with Real Time and Multimedia Capabilities

The incorporation of multimedia data into control systems presents new challenges in the storage and movement of this time-dependent information [11]. That is one reason why the Real-Time Multimedia Database (RTMMDB) is an essential component of the IDS. The RTMMDB is responsible for the storage, transformation and retrieval of multimedia and traditional data collected throughout the plant. It interacts with both the RTDCS and the MIS in order to service their respective clients. To achieve its functionality the RTMMDB can be distributed or centralized, and it consists of a query engine, a global model of the system state, a retrieval engine and a composer.

- The query engine should support real-time query processing and deal with queries such as: *get me the state information of generator X for the last 5 minutes at station A*, or, *search for all occurrences of a particular event in the last week.* Query processing involves the following: search, retrieval, collation and actual transmission. There may be some time bound on the delay between the time the query is submitted and the time the results are sent to the user. If the information being sent contains stream-like data such as audio and video, the transmission phase will also have timing constraints on it. The composer acts as a filter. It composes the retrieved information from multiple stored/live streams into a presentation format compatible with the requested query. The system scheduler must handle the dynamically arriving processing workloads associated with the queries.

- The database collects various incoming data streams, and uses the information to update the 3D model of the global system state. If this model is to be used for online decision making as in the RTDCS, the transaction workload associated with the updates to the system state will have to be handled in real-time.
- Other functions in the database include indexing (text, image, audio, video), and intermedia integration.

Broadly, two types of information may be requested. One type is intended to provide state information in real-time. System support is required for low latency transmission of large volume multimedia data, on-the-fly raw data and image processing, filtering, and authoring services to extract the relevant information and present it in a suitable format to the user. Possible clients for this service are the remote operators who cannot be physically present on the factory floor, but still need a realistic feel of the workplace environment, e.g., in applications such as telepresence, telerobotics, and teleoperations.

The second type of information supported by the database is *stored* historical information (either event-based or continuous) which will aid management to make long term decisions, supervising operators to analyze, in detail, situations leading up to abnormal fault-like situations, and possible erroneous operator decisions. The requested information may be available either as a single stored entity or distributed across multiple logical information entities. In the latter case, the requested information will have to be collated together, temporally correlating the different entities and then sent to the user. Users in this class will also be more tolerant to less than deterministic types of QoS guarantees as occasional failures here are not critical to the working of the factory itself.

Some open research questions in the context of the database are:

- How should the Multimedia Database be structured?
- How do we satisfy complex ordering requirements (e.g., multimedia synchronization, temporal correlation of images/data from different sources) for orchestrated presentations?
- How and when should information from different media be integrated/synchronized with each other and with real-time data?

- How do we maintain consistency given that the data sources are distributed?

All of these questions are closely connected because they deal with the organization of the physical storage and techniques used for retrieval and presentation. Data organization is an important issue for multimedia data given the large quantities of data involved and their influence on efficient query processing and information retrieval.

If we store information from each source (e.g., images, text, audio, video) separately, the advantage is that media-specific organization and storage can be used and the information needed can be integrated dynamically on-the-fly in response to user requests. This avoids redundant storage. Alternatively, when the data comes to the IDS, the integration is done automatically and what is stored is the integrated information. The drawback of this approach is that it presupposes a certain set of queries for which it builds the appropriate integrated data sets. This may not be true in practice as it is not possible to accurately anticipate user needs beforehand. It is possible that a particular user may want only a subset of some stored data set, and the system resources can also support transmission of that subset to the user.

Thus, the first approach, of dynamically integrating needed data is preferable. In this case, clearly, there is a need for a global time base and a means to timestamp the data so that data originating from different sources can be correlated. The granularity of the timestamps will determine how close related data can be correlated. Furthermore, given the inherent limitations of retrieval algorithms for multimedia (e.g., retrieve images of pipes that had a *similar* rupture), processing techniques are necessary to combine data that is associated with different levels of confidence.

Finally, transmitting temporal data in a distributed system so that it remains fresh requires efficient protocols for data to be delivered between nodes. The data servers and network servers must interact so as to make efficient use of buffer, bandwidth, and computing resources to satisfy specific QoS requirements.

3.3 Supervisory Monitoring and Control Module

This module contains the plant command center operator interface. It comprises one or more human supervisor operators, each responsible for a certain portion of the work area, that is to a subset of the LCs. Each has access to a local PC or workstation and is connected via the ATM network to the IDS and also to the different LCs in its designated area. It interacts with the IDS to set up its data and control connections to the LCs. The supervisory operators may be in a central location or physically distributed throughout the plant.

3.4 Management Module

The Management module provides a wide variety of services appropriate for personnel who undertake, for example, long term planning, post-mortem analysis, and maintenance procedures. Typically, users in this class will access non real-time information such as state of the plant in a certain interval in the past, log of decisions taken to handle specific normal or extraordinary plant situations in the past, plant performance statistics, as well as general information available from other plants or via the World Wide Web. They can be served by the MIS which will cater to this type of stored offline information.

4 SUMMARY

Research in real-time multimedia databases is only recently being undertaken. In doing this work, it is necessary to understand how and where such databases might be used. In the previous sections, we described an industrial application that can benefit from this research. In addition, we presented a high level architecture and discussed the system level support needed for an efficient integration of multimedia capabilities into the control environment. This exercise enables us to enumerate the many research issues that need to be addressed to make this a reality.

5 ACKNOWLEDGMENTS

This research was supported in part by a contract from the Community Management Staff's Massive Digital Data Systems Program and Mitsubishi Electric Corporation.

REFERENCES

[1] J. Douglas, "The Maintenance Revolution," *EPRI Journal*, Vol. 20, No. 3, May/June 1995, pp. 6-15.

[2] J. Douglas, "Visualizing Complex Systems," *EPRI Journal*, Vol. 19, No. 8, December 1994, pp. 18-23.

[3] A. Guha, A. Pavan, J. Liu, A. Rastogi and T. Steeves, "Supporting Real-Time and Multimedia Applications on the Mercuri Testbed," *IEEE Journal on Selected Areas In Communications*, Vol. 13, No. 4, May 1995, pp. 749-763.

[4] J. Huang and D. Du, "Resource Management for Continuous Multimedia Database Applications," *Proceedings of the* 15^{th} *IEEE Real Time Systems Symposium*, Puerto Rico, December 1994.

[5] H. Kaneko, J. A. Stankovic, S. Sen and K. Ramamritham, "Integrated Scheduling of Multimedia and Hard Real-Time Tasks," (to appear) *IEEE Real Time Systems Symposium*, December 1996.

[6] G. Meinert, "Openness for Automation Networks," *InTech*, October 1995, pp. 29-32.

[7] I. Mizunuma, C. Shen and M. Takegaki, "Middleware for Distributed Industrial Real-Time Systems on ATM Networks," (to appear) *IEEE Real Time Systems Symposium*, December 1996.

[8] K. Sato, A. Tsukada, F. Matsuda, K. Kawasaki and M. Ozaki, "Multimedia Systems for Industrial Surveillance," (to appear) *Multimedia Computing and Networking 1997*, February 1997.

[9] C. Shen, "On ATM Support for Distributed Real-Time Applications," *Proceedings of Real-Time Technology and Applications Symposium*, June 1996.

[10] J.A. Stankovic, "Continuous and Multimedia OS Support In Real-Time Control Applications," *Fifth Workshop on Hot Topics in Operating Systems (HotOS-V)*, May 1995, pp. 8-11.

[11] W. Sterling, F. Cariño and C. Boss, "Multimedia Databases and Servers," *AT&T Technical Journal*, Vol. 74, No. 5, September/October 1995, pp. 54-67.

[12] S. Teger, "Multimedia-From Vision to Reality," *AT&T Technical Journal*, Vol. 74, No. 5, September/October 1995, pp. 4-13.

[13] Realaudio is a product of Progressive Networks, 1996. At http://www.realaudio.com/.

[14] Vosaic is a product of Vosaic Corp., 1996. At http://www.Vosaic.com/.