AN OBJECT MODEL FOR INTEGRATING PRODUCTION MANAGEMENT IN NEWSPAPERS

Nils Enlund*, Stig Nordqvist**, Jari Alasuvanto***, Reijo Sulonen***

Keywords: Process management, integrated newspaper production, production management systems.

Abstract: Process management and production monitoring in newspaper production is currently, when at all present, handled by separate function based local monitoring systems, e.g., ad tracking, press control, and mailroom control systems. There is a need for interchange of production status information between such local control systems as well as for standardized data acquisition methods for company-wide production management. This paper discusses a general, structured model for the exchange of object and production status information between separate multivendor and -platform production management systems that will support integrated production management in newspapers. A hierarchical view is taken that allows for production monitoring on different levels of granularity. A proposed message format for the interchange of object and status information is described.

THE NEED FOR INTEGRATED PRODUCTION MANAGEMENT

As the newspaper industry moves toward integrated production processes, the need for tools to monitor and manage the process becomes increasingly obvious. The economic situation also increases the interest in efficient monitoring and management of the entire newspaper production.

* Helsinki University of Technology, Media Technology Group, Espoo, Finland, MediaCom Oy, Porvoo, Finland, and Royal Institute of Technology, Division of Graphic Arts Technology, Stockholm, Sweden (nilse@gt.kth.se).

** Royal Institute of Technology, Division of Graphic Arts Technology, Stockholm, Sweden (stign@gt.kth.se).

*** Helsinki University of Technology, Media Technology Group, Espoo, Finland (jaa@cs.hut.fi and shs@cs.hut.fi).

The management of events and objects in the newspaper production process is today, when present at all, handled by function based, local production management systems [Karttunen, 1994]. We speak of a local production management or monitoring system, when the scope of the system is limited to a functionally distinct part of the entire newspaper production process. Local solutions exist today for mailroom management, press control, page output monitoring and printing plate tracking. In the prepress area, a few ad tracking and document management systems have been introduced in the marketplace, whereas production management within the editorial department has only recently been addressed by systems manufacturers [Enlund, 1992].

There is a definite need in newspapers for extending production monitoring and management to cover the entire process, from news gathering and ad marketing to the delivery of printed copies to the readers. Monitoring information must be exchanged between local production management systems in order to be able to follow, e.g., the progress of a page from make-up through imaging and platemaking to press mounting. In addition to this, status information from local production management systems must be gathered in a global production management system. Such a global system can provide an overall view of the entire production and supply management information for planning, process improvement and strategic planning [Nordqvist and Enlund, 1993].

In this paper we will describe an approach to creating a framework and a methodology for exchanging production monitoring information between heterogeneous local production management systems and between local and global production management systems. Our objective is to support the integration of local production management solutions from different manufacturers and to facilitate the construction of global company-wide production monitoring systems in a mixed systems environment.

Production management can take many forms and can be applied on various levels of ambition. In an earlier report [Nordqvist and Enlund, 1993], we suggested a four tier hierarchical model of production management systems. This taxonomy model includes the levels of production monitoring, scheduling, closed-loop control, and production management. In this structure, monitoring (often also referred to as tracking) is the basic, technically and conceptually least complex level. The proposed model also includes the communications infrastructure and the functions of simulating the production process as a decision support tool [Nordqvist et al., 1994].

In this paper we will address only the simplest level of production management: production monitoring. As will be pointed out later, the extension of the concepts presented in this paper into higher levels of production management will require much additional work. However, since the knowledge of the production process status acquired through process monitoring is the necessary basis for all other forms of production management, it is logical to first investigate the problems in production monitoring.

Figure 1 shows an example of a hypothetical production process with several subprocesses — production phases — each being monitored by a local production management system (PMS). The black arrows indicate exchange of monitoring information. There may well be a hierarchy of management systems — for instance, locally monitored prepress functions may be coordinated by an intermediate prepress management system.

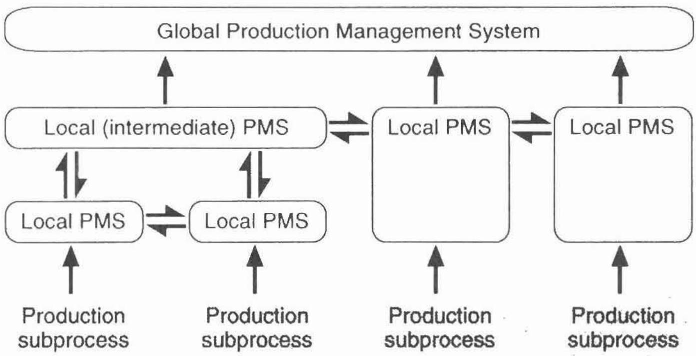

Figure 1: The exchange of monitoring information between local and global production management systems in a general case.

Several local production management systems (some of them integrated with the production systems) are available today from different manufacturers and many new solutions are under development. Each of these systems takes its own approach to solving a locally defined management problem. There is no reason for an attempt to standardize these approaches — indeed, a standard may hamper innovation in this rapidly developing field. Instead, the ways and methods the local production management systems use for communicating with external systems should be defined in a structured manner.

A model for the interchange of production management information must be defined with respect to three aspects:

- Semantics — the contents of the information that must be exchanged.
- Syntax — the description language and structure for encoding the semantics of production management interchange.
- Delivery mechanism — the technology and methods for delivering the messages from sender to receiver.

SEMANTICS

Objects and granularity

Production monitoring is concerned with **objects** and their **states**. The state of an object is modified through **processes**, or activities, to which **resources** are assigned. The changing of the state of an object we call an **event**.

Project monitoring consists of the registering of events. By associating the events with objects and their states, the production monitoring system can follow the progress of the production run.

An object is a hierarchical construct that can be observed and its state monitored on various levels of granularity. We have chosen to use a broad definition of the concept of objects. An object can be a physical object but also an instance of a process that produces a physical object. On the topmost level, the entire manufacturing process of an issue of a newspaper can be considered as one object. Production is the process that transforms the state of this object from an initial state, e.g., {created} to a final state, e.g., {completed}.

On the next, more detailed level, we may observe the production of a newspaper issue as consisting of, e.g., four main functions: prepress, press, mailroom and distribution. Looking closer, we can identify the different production steps in the prepress area: editorial, advertising, composing room, repro and platemaking. And, going even further, we can identify the different steps necessary for producing advertisements.

These views represent different levels of granularity. We observe processes and objects with various degrees of detail.

Granularity is a central issue in production monitoring. Local production management solutions work internally and communicate with other systems on different levels of granularity. An ad tracking system is concerned with bookings, texts, images, layouts and completed ads. A plate monitoring system is concerned with pages, paste-ups, films, printing plates, register punches, benders and plate transports. A global production tracking system might be concerned with plates, presses, copies, bundles and trucks. They all work on different levels of granularity. Still, they need to exchange event information, both horizontally and vertically.

A method for exchanging event information must be flexible and general enough to accommodate event message exchange on different levels of granularity. The same format must be usable for ad tracking within the prepress area, for page and plate monitoring between prepress and press, and for global monitoring of the entire newspaper production process.

Object classes

In order to describe the objects to be monitored in the newspaper production process, we define a set of generic abstract object classes. The actual objects monitored during a production run are instances of these object classes.

The object classes can and should be defined hierarchically in each production environment. The definition must be made at the level of granularity on which monitoring is required and according to the system structure of the particular newspaper company.

On a high level of abstraction, considering only major functions and material, the traditional newspaper production process may be described as in figure 2.

On this level, the main internal object classes subject to production monitoring are Plate, Copy, and Bundle. We have chosen not to include externally generated objects nor raw materials in our discussion, but our methodology can easily be extended to include them in a monitoring system.

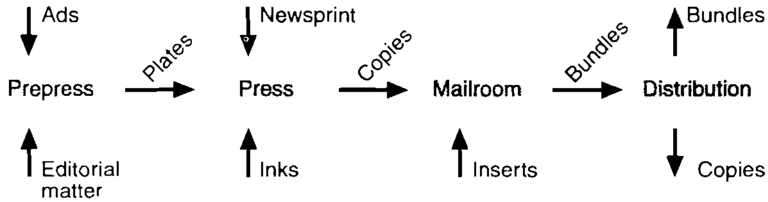

Figure 2: A high level view of the newspaper production process.

A closer study of the prepress production process reveals a number of distinct functions, or subprocesses, passing information and other material between them. There is also a need for exchanging production status concerning these objects between the corresponding local, function-based production tracking systems. Figure 3 shows an example of a general typical prepress production process. In such a process, the object classes subject to production monitoring might be Advertisement, Story, Image, Processed image, Page, and Page film.

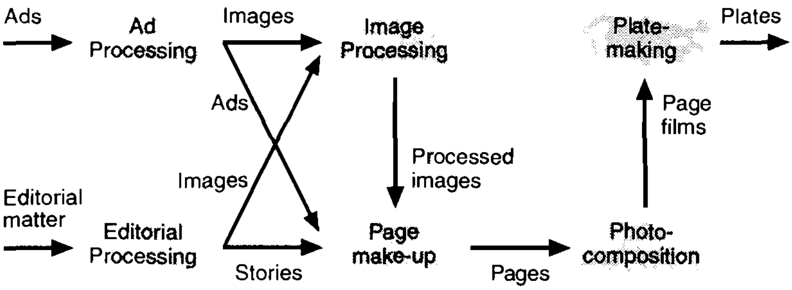

Figure 3: A closer look at a simplified prepress production process.

In the press area, object classes may be defined, for instance, as Plate, Imposition, Press run, Production sequence, and Production set. In the mailroom, object classes may be Copy, Preprint copy, Label, Bundle, Order, Load, and Route.

It is obvious, that the types of objects that are to be monitored are dependent on the work flow, functional organisation and technical solutions in each newspaper company [Alasuvanto et al., 1993].

The general rule for defining the object classes on which event information has to be exchanged between local production monitoring systems is: status information must be exchanged concerning objects that are passed between production functions being monitored by separate local production management systems.

Object states and events

There is a set of possible states associated with each object class. The general object class independent states are {created}, {in process}, {suspended}, {terminated} and {completed}. These are common to all object classes. There may also be object class dependent intermediate states.

Within a hierarchical object structure, states can be propagated upward in the structure. A Page object cannot attain the state {completed} until all objects on lower hierachical levels, i.e., the elements of the page, all are in the state {completed}. This characteristic can be used as a verification method in production monitoring — when an object is reported to be {completed}, we can safely assume that all lower level objects are also {completed}. Upward propagation of states is also important if we take a quality view of objects. An object is good only if all its components are good. If we later detect that for some reason one component is not good (a factual error in a news story), it is necessary to reprocess all related objects on hierarchically higher levels (new page originals and printing plates).

There is generally no downward propagation of states. The termination of, e.g., a page does not automatically mean that all its components are terminated — elements may be moved to other pages.

An event signifies the transition of an object from one state to another. If there exists a defined order of allowed states for each object type, an event can be identified by the resulting state of the object concerned and by its previous state. Figure 4 shows the state diagram of the basic object states.

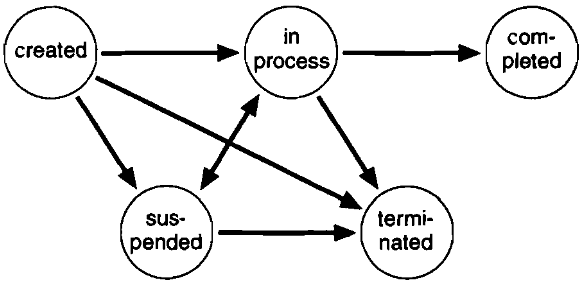

Figure 4: The basic object states and the allowed state transitions.

In order to keep the number of possible states and transitions as low as possible, we propose a solution where more complex state transitions are represented by several events. For instance, the renaming of an object may be represented by the termination of one object and the simultaneous creation of another. In the same manner, the completion of one object may generate a new object, thereby, e.g., transforming an Image into a Processed image. The relationship between such objects must be handled by strict object naming rules.

SYNTAX

Information concerning the events defined above must, in a production monitoring situation, be communicated between various production management systems from different manufacturers. A standard exchange format for communicating event information has to be designed.

Any method for interchanging production monitoring information between local production management systems in the newspaper production process should be kept as simple as possible. The format must be flexible enough to accommodate all the necessary monitoring information on various levels of granularity. At the same time, it must not restrict the designers of local and global production management solutions in any way. Simplicity is the key: the essential information and only the essential information must be included [Enlund, 1994].

We propose a simple **event message envelope**. This envelope identifies the information contained in it as production management event information. The format of the envelope can have the following structure:

<event message header><event message><event message trailer>

The header is a simple code identifying the following message as an event message. It may also contain information about the length of the event message and the staring points of each message component within the event message. The trailer is a simple code indicating the end of the message.

The actual **event message**, enclosed within the envelope, contains the following information:

<originator> Data identifying the system or subsystem generating the message. There is a need for developing a company-wide system identification method that assigns unique codes or numbers to each system. The originator naming method must be able to express hierarchical relationships between originator systems.

<object-id> A unique code, identifying the object that the message concerns. The identification must also cover newspaper issue and edition. A well designed object naming method can also include structural information. Object hierarchy relationships can be indicated by using the concept of an "object path name", where the name of an object includes the names of all parent objects.

<object class> An identifier that indicates the class of the object that the message concerns. The different object classes must be specified on different levels of granularity.

<status change> The reason for issuing an event message is that a change in the status of an object has taken place. For instance, a certain page has been output on an image setter, or a certain printing plate is ready. A status change is best identified by indicating the new status that the object has attained — each local or global production management system can keep track of the status history of the objects.

<time> The exact time for a status change must be included in the event message. There must also be a way to synchronize clocks between the different subsystems.

The above event information constitutes a minimum set of production management data to be passed between production management systems. Using this type of messages, all essential production event information can be expressed. For instance, the addition of two pages in a newspaper issue can be expressed by identifying the new pages with the new status {created}. Or, the completion of a printing plate can be indicated by page-id and the new status {completed}.

Let us illustrate the event message format with an imaginary example. An event message indicating that the first magenta printing plate for page 24 of the May 5, 1994 issue of edition 2 of the Baltimore *Sun* has been made ready at 10:05 pm could look something like this:

Start event message
    Production Platemaker 2
    BS 940505 ed 2 page 24 magenta 1
    Printing plate
    Completed
    940504-220546
End event message

The use of hierarchical naming methods in event messages, both for message originators and objects, makes the automatic filtering out of irrelevant information straightforward. A monitoring system can, by simple analysis of originator and object names, select only those event messages that are of interest.

MESSAGE DELIVERY MECHANISMS

A simple but reliable mechanism for delivering event messages from one production management system to another, or from one local system to a global one, has to be designed. There are three basic requirements on a suitable message delivery mechanism:

- Simplicity. The mechanism must be easy to implement and must not require any significant processing overhead.
- Openness. Each system should need to have as little knowledge of the other systems as possible, preferably none at all, while still being able to communicate with them. The less the various systems know about each other, the easier it is to add and replace modules in the total production system.
- Reliability. Each system must receive all messages of relevance.

Let us call a system that generates a tracking message indicating a significant process event a "sender". Any system, local or global, that can register and use this tracking message is a "receiver".

There are three basic methods that could be used in delivering these messages in a heterogeneous systems environment:

- **Broadcasting.** The sender outputs the event messages on a local area network, connecting all systems. It is the responsibility of all receivers to continuously monitor the network and register all relevant event messages. This makes the communication method open and extensible: the sender needs to know nothing about the rest of the production tracking system nor about the receivers.

  Broadcasting is a communication method that is extremely simple to implement. The main problem is, however, the inherent lack of reliability of the method. The sender can never know whether anyone is listening and the receiver cannot know if it has received all relevant messages. If the receiver, for some reason, has been unable to continuously monitor the transmissions on the network, it has no way of catching up.

- **Handshaking.** The problem of reliability of communication can be solved by establishing a handshaking protocol between sender and receiver. The cost of this reliability is high. A handshaking method can only work if all systems know of the existence of all other systems, and if each sender knows the destination(s) of each event message. An implementation of a handshaking communication method would reduce the simplicity, openness and flexibility of the solution. Any change in the production management systems would require the reprogramming of all senders and receivers to reflect this change.

- **Data base.** A communication methods that combines simplicity and openness with a reasonable degree of reliability is that of using a common data base as a storage for event messages. The sender transmits the event messages to the data base where they are recorded in a message log. Each receiver polls the data base for new messages and retrieves the relevant ones.

  This method introduces a slight delay in the event message traffic, but this is of minor consequence since the real-time requirements of newspaper production monitoring are rather limited. The method is open and extensible, since the systems only communicate through a common data base and need no additional knowledge of each other. It also ensures a certain level of reliability, since a receiver can access also older tracking information in the data base. Actually, it gives each receiver the option of retrieving and analysing the event information in the common data base in any way it sees fit — each system manufacturer can design his own analysis and presentation methods.

  The data base can be very simple. A flat file data base, logging the tracking messages in order of arrival and permitting sequential access and retrieval is sufficient. However, in order to facilitate standardization of access, an SQL based data base can be used. Most system manufacturers build their products around SQL data bases, thereby making the integration of a production monitoring SQL data base trivial.

We see the data base method as the most promising approach at this moment.

BEYOND MONITORING: INTEGRATED PRODUCTION CONTROL AND MANAGEMENT

This paper has outlined a method for designing a protocol for the interchange of production monitoring information between production management systems for the newspaper production process. An industry-wide implementation of the event message format described above would make it possible to implement production monitoring covering the entire newspaper production process. This would be a great achievement, but it would still only constitute a beginning of the process of introducing in the newspaper industry the same type of production management that is commonplace in other industries.

Production monitoring is not enough. Monitoring will tell us only what has already happened. If we know what is supposed to happen and when, i.e., if we have encoded a production schedule or a set of acceptable futures, monitoring will also inform us when something has gone astray. It will not, however, help us avoid or correct the problem during the production run. In order to obtain a smooth, efficient newspaper production process, we will have to move beyond monitoring into the realms of closed-loop, adaptive production control and production management [Hodges, 1994].

Production control involves the active rescheduling of worksteps and the reassignment of resources in cases when there is a deviation from a predefined production schedule. That is, when something goes wrong, a production control system will attempt to correct the problem in a way that minimises delays and costs. In the case of major rescheduling operations, it will request approval and advice from an operator.

For active production control to be possible, the control system requires detailed information on the product to be produced, the worksteps involved, the resources available and the default production schedules. In a heterogeneous environment, involving production systems from different manufacturers, this information — as well as information on changes in products and schedules — has to be propagated between the control systems in a structured, standardized manner. The definition of these mechanisms should be the objective of future initiatives and efforts.

Production management involves the collection and analysis of data on production runs and production costs, covering the entire newspaper production process. This information will be used by production and general management to support process improvement and strategic planning [Fuchs, 1994]. Also in this area, structured and standardized information exchange mechanisms must be designed. We plan to continue our work in this direction.

ACKNOWLEDGEMENTS

Much of the work reported in this paper has been discussed within the IFRA Production Tracking Working Group, of which one of the authors, Nils Enlund, is chairman. These discussions have contributed significantly to the paper. The members of the IFRA Production Tracking Working Group represent newspapers as well as manufacturers of prepress, press and mailroom control systems. The objective of the Working Group is to devise recommendations for the interchange of production tracking information between systems from different manufacturers. A recommendation is forthcoming by the end of 1994.

LITERATURE CITED

Alasuvanto, J., Enlund, N., Mäntylä, R., Riipinen, M., Sulonen, R.
1993 "Workflows in Newspaper Prepress Departments", Groupware '93 Proceedings, New York, pp. 279-295.

Alasuvanto, J., Enlund, N., Mäntylä, R., Sulonen, R., Vuorikoski, M.
1994 "Process Management for Newspaper Prepress — A Systems View", paperpresented at the TAGA '94 Conference, Baltimore

Enlund, N.
1992 "Production Management in Newspapers", Proceedings of the IBM International Customer Executive Conference for the Media Industry, La Hulpe, 9 p

Enlund, N.
1994 "Tracking Newspaper Production", IFRA Newspaper Techniques, February 1994, pp. 18–22.

Fuchs, B.
1994 "Lean Production in der Zeitungsindustrie", IFRA Special Report 6.15, Darmstadt, 40 p.

Hodges, C.
1994 "Implementation of a Production Management System", Proceedings of the Seybold Newspaper Conference 1994, London, pp. 46–52.

Karttunen, S.
1993 "Production and Production Management Systems (PMS) for Publishing and Printing", Graphic Arts in Finland, Vol. 22, Nr. 3, pp. 3–10.

Nordqvist, S., Enlund, N.
1993 "Global Production Management for Integrating Prepress, Press and Postpress Systems in Newspapers", paper presented at the 22nd IARIGAI Conference, Munich, 15 p.

Nordqvist, S., Stenberg, J., Karttunen, S.
1994 "Modelling of Production Management Solutions", paper presented at the TAGA '94 Conference, Baltimore, 16 p.