Event Driven Digital Publishing Workflows

Alex JONSSON, Lic. Tech.

Royal Institute of Technology, Media Technology & Graphic Arts
Drottning Kristinas väg 47. S-100 44 Stockholm, Sweden
alexj@gt.kth.se, http://www.gt.kth.se/~alexj/

ABSTRACT: By utilizing technology derived from critical real-time information environments, such as the stock exchange or security systems, in a more general information sharing contexts, the quality of the information flows change and thus the organization and the business model itself. By receiving and sending custom information on occurrence, in user-selected media channels, rather than by traditional pull or polling methods, and being able to react upon the information, preferably using the same channel, the organization becomes faster and more adaptive to change.

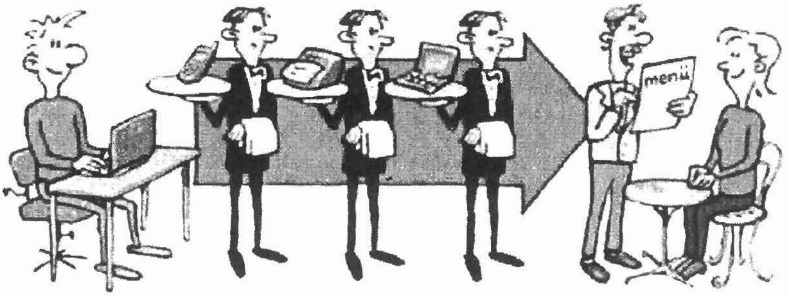

Figure 1. Conceptual approach to an event driven publishing workflow. The consumer subscribes to information, which is distributed continuously on occurrence in accordance with the subscriber's preset preferences.

1. Background

»Any product or service can be copied...a company's only advantage is to learn, unlearn and relearn faster than its competitors« [TAPSCOT 99]. The information flows within a company or organization is a key issue to meet future demands from the market and society, adapt to change and prosper in the industry. By producing faster information flows with less friction, the entire organization can act, react and learn faster than before.

A preliminary study conducted during 1999 at the Swedish Employers' organization, SAF, [Winsnes 1999] showed that staff members of the Information Department spend less time retrieving, evaluating and compiling information than the time spent identifying and distributing information to the appropriate receiving parties. If each recipient can be equipped with the appropriate tools to take responsibility for their own information flows themselves, information could be distributed with less friction. Distribution of information using a publish – subscription model, will then flow in accordance to each subscribers preferences and personal taste.

It all boils down to investigating how each user retrieves her information on a daily basis and converting these preferences into profiles. The profile is created once, stored in a user database and is fine-tuned continuously as the user becomes a more efficient information consumer. For example, one user might prefer to have some types of information sent on occurrence to her personal e-mail account, but other types sent to her cellular phone using SMS (Short Message Service) and why not a summary of each day's highlights, sent to a fax machine nearby, every evening. Another user in the system may prefer to have e.g. financial information sent to a Java applet on the desktop and all other subjects channeled to the e-mail account or printed on demand onto paper. There are probably as many preferred profiles as there are users in the system. See figure 1.

2. Objective

This report is based upon the assumption that if the information flow within an organization can be routed at a higher pace, by minimizing downtime and delays, the whole organization may become more efficient and thus successful within its industry. Another assumption is that each individual in the system takes responsibility for their own information flow, both for sending and receiving data. This report will describe a conceptual model and framework for how such a system can be put in practice.

3. Concepts and approach

The role of producing information is called a »publisher«. Each entity of information is referred to as a »message«. Messages are then sorted into information categories, referred to as »subjects«. Around each subject, publishers are sorted into "trusted groups", one for each main subject. Information is received and consumed by means of subscription, on a subject level; this role is called the »subscriber«. In effect, a user in the system can, with the proper authority, obtain the role of publisher or subscriber at any time, individually set for each subject. The key lies within the privileges set in the user's publishing and subscription profiles; how they are logically set up, managed and utilized over time. This conceptual approach will be referred to as CIS (Coordinated Information Services) within this report. Subject-based addressing, to directs messages to their destinations so application processes can communicate without IP addresses or connections, features a set of rules that define a uniform name space for messages and their destinations. The subscriber independently looks for subject names of interest, regardless of source, in accordance to their subscription profile [Tibco00].

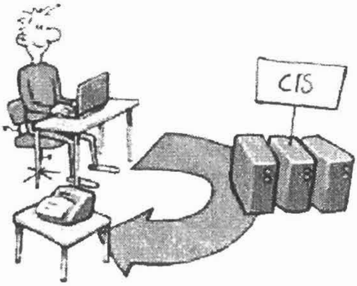

Figure 2. The publisher is unaware of the subscriber's preferences in the publishing profile; it is facilitated by the system.

3.1 System effects

These are several effects of the chosen conceptual approach, of which these listed here are considered the most important. These constraints and their effects will all be explained in further detail in chapter 4.

The publisher…
…keeps no record of the subscribers preferred channels
…does not have access the subscribers contact information
…belongs to one or more trusted groups for creating information

The message…
…is enveloped with metadata, provided by the publisher.
…is distributed and stored in a document description language, such as XML, SGML as well as related formats for multimedia information formats.

The subscriber...
...can well be unaware of the publisher's identity
... selects all preferred media channels herself on a contextual, subject-based level.
...can act upon the information received, preferably in the same channel, by obtaining the role of a publisher.

4. Method

The chosen approach is to find methods which make it possible to »separate« the publisher from the subscriber. Our choice is to use a four-tier model, with two intermediate layers separating the two parties, creating the framework for the trusted groups necessary for event driven publishing workflows with dynamic properties. By assigning a trusted group of information providers for each category of information, information can be authenticated on a message level. For example, a co-worker can send an inquiry by e-mail to the trusted group regarding company policy concerning car rental. The reply, in this case received by fax according to the co-worker subscription profile, states the requested policy. The subscriber does not in fact need to know which individual that actually sent the reply to the inquiry, just that it was received from within the trusted group in question. When a new subscriber enters the system, she is assigned a default profile that corresponds to the job description at hand. This profile can then augmented or diminished in accordance with her preferences. See system overview in figure 3.

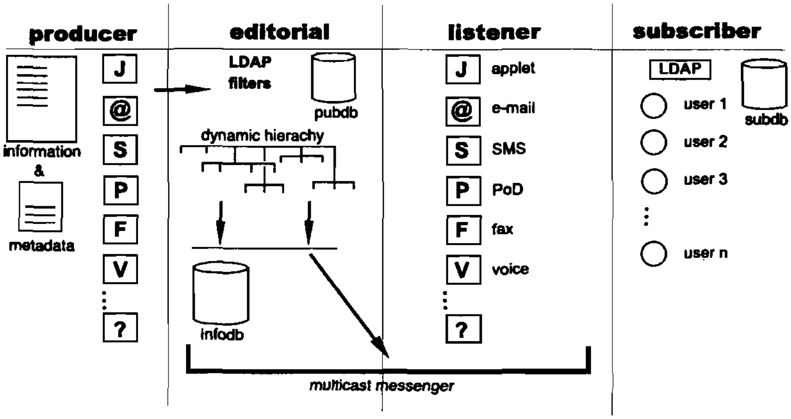

Figure 3. The four–tier publish–subscribe model of the CIS system.

4.1 CIS producer tier

During the process of content authoring, it is most important that authoring is made simple and facilitated so that tags and other code is hidden from the producer. Each message is stored in a database with a unique id, a reference in the publishing system. The reference can then be used as an alternative to re-authoring, allowing re-publishing and less risk of introducing errors in the system, as argued by [Saarela99]. By adding metadata as part of the authoring process, the message description can be made richer and more automated than if metadata is added as a post-process, by other individuals. Some may argue that author-independent is a more objective way of categorizing messages, but this will defer the information flow and is considered foreign to the approach, since the main challenge is to create faster information flows.

If messages can be described at the source [figure 4], much is gained and also a prerequisite to render the middleman obsolete. This has proven to be found a frightening concept, since many job description on a management level imply various levels of information filter functions within a company's infrastructure. By redefining the role of the information filtering jobs into becoming guides of the information flows, handling the subscribers profiles will add to the flexibility of the organization as a whole.

By letting go of the information control, the flow of information will become more powerful. Just as each user can have a preferred set of channels for subscription of information, she has a preferred set of channels for publishing.

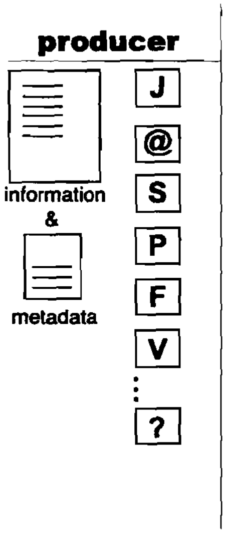

Figure 4. Messages with metadata are published via the preferred channel

Metadata in this case is typically a set of predefined fields. Here is a typical example of metadata DTD described in an SGML format [Goldfarb 1990]:

```
<! ELEMENT mess_id - - CDATA>
<! ELEMENT subject - - CDATA>
<! ELEMENT author - - CDATA>
<! ELEMENT date - - (day,month,year) CDATA>
<! ELEMENT keywords - - CDATA>
<! ELEMENT description - - CDATA>
<! ELEMENT day - - CDATA>
<! ELEMENT month - - CDATA>
<! ELEMENT year - - CDATA>
<! ATTLIST subject mess_id CDATA #IMPLIED>
```

The system can provide most of the metadata automatically, retrieved from the publishers' database and the LDAP server. When publishing from a cellular telephone, with a maximum number of characters of 160, there is very little authoring space for entering metadata and must hence be obtained from previously stored data. Reading from printed matter requires an OCR (Optical Character Recognition) input device. Metadata will then be entered by other means, typically using a web interface or a standalone application.

The subject field will be entered using a list of set subjects, although the responsible member of each trusted group has privileges to set new subjects. These will show up in the publishing system as child subjects to the main subject of the trusted group. Subscribers of this group will per default get this new subject included in their profiles, using the same subscription channel as the parent subject, until altered by the subscriber in her profile. This is a positive push publishing approach, where the subscriber must undergo the active task of unsubscribing, rather than subscribing. A bit confusing concept at first since the result could be a massive overflow of information, instead of the right information to the right individual at the right moment. But, the act of subscribing is a typical client-pull process and needs only to be performed once for each subject category. The task of adding new subjects is much like adding new categories to a library, a delicate task that requires thought and consideration. Each user with this privilege must act in a conservative manner, and should be regarded as corporate policy decision.

4.2 CIS editorial tier

By converting each message into an SGML or XML format, the reusability of the information and the value is restored. The reuse of one-time authored information is a key factor in the first two tiers, to minimize the need for authoring resources, re-authoring and influences of human error. The metadata can then be inserted into the actual document structure, to maintain the information value of the entire message.

In the example below, a typical XML description of the SGML DTD stated in the previous subchapter [W3C]:

```
<message mess_id="12345678" subject="Security PM">
   <date>
   <month>12</month><day>10</day><year>2000</year>
   </date>
   <keywords>alarm,fire,siren,drill</keywords>
   <description>Manual for fire drills proceedings
   </description>
   <author>Alex Jonsson</author>
   <contents>In case of a fire … act as planned
   </contents>
</message>
```

The editorial tier consists of five logic parts as described in figure 3:

- A parser that converts each message into an XML document
- An LDAP server that stores the contact information of each user
- A publishers' database that handles authentication for publishing and defined the trusted groups on a subject level
- An SQL interface for entering messages into a message database
- An interface for encapsulating the message, along with its metadata and authentication properties, into a format suitable for the middleware.

All subjects are actually placed on the same level, and the hierarchical structure is created using a look-up table in the message database. The nature of a relational database makes the hierarchical model useful for viewing purposes, when accessing the information at a later date or performing searches. The usage and interface design of such information retrieval systems is known territory and will not be discussed further herein.

The middleware, tying the second »Editorial« and third tier »Listener« together can be one of several. A natural choice in accordance to the conceptual approach described in chapter 2, is to use a technology that has the least impact on the network, allows implementation of the publish–subcriber model and that can transport payload contents over a network regardless of its format and data size. A good example ca be found at Tibco, a company that delivers software solutions mainly for financial applications, who has developed a middleware engine called TIB/Rendezvous.

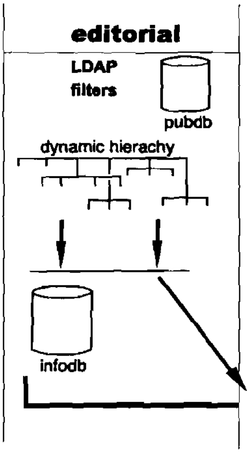

Figure 5. The editorial tier. Incoming messages are filtered sent to the listeners.

This piece of software is basically a middleware messaging tool for distributed applications that allows privileges to be set for reading and writing on a message level, using the metadata together with the privilege data from the publishers' database. When applied to the message, it describes the privileges coupled with the message payload and applies it to the message's envelope. It is then multicasted to those listeners that subscribe to a specific port on a multicast address. Another approach to facilitating message distribution would be to use a IP queuing application, where publishing requests would be placed in a buffer prior to publishing on the network, thereby obtaining a similar result. But, this choice of technology does not rhyme well with the chosen conceptual approach and is neither location transparent; hence it is discarded for the task at hand.

As shown in figure 5, there are two concurrent events symbolized by the two vertical arrows. They show that each message is both stored in the information database »infodb« and sent to the listeners of the third tier. The choice not to route all information through the message database is both a system performance consideration but also a deliberate design choice, true to the publish-subscribe model. This is also done with a reservation for future usage, allowing the introduction of system components or interfaces to external systems that require split-second publishing performance where mid-flight storage in a relational database simply would not be fast enough. There are time-critical applications where each idle second decreases the information value of a message sent, e.g. stock exchange information, system errors from computers or fire alerts.

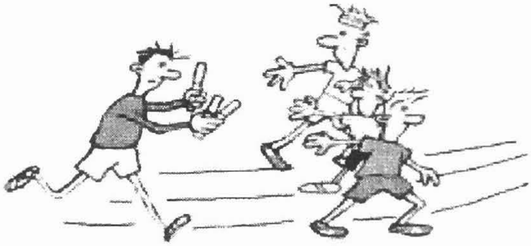

Figure 6. A middleware technology that supports concurrent events through mulitcasting, is far politer than sending all data to every computer. It is faster and has greater scalability in a system with many subscribers.

4.3. Listener tier

A listener is basically an open socket attached to a piece of conversion software that converts the encapsulated XML document into a format suitable for each specific listener. The specification for each listener depends on what off-the-shelf software is available on the market [figure 7] For instance, a fax listener consists typically of a desktop computer connected to a fax modem. The routing to a specific subscriber depends on the information in the lookup table in the subscribers' database. Matching is performed on a subject level and the subscribers' privileges. Each listener operates independently of every other listener, giving the system a high degree of active redundancy. Subsequently, each listener's subscriber data is retrieved independently from the subscriber's database »subdb« in the subscriber's tier [figure 8], and is fetched anew periodically or upon request when changes have occurred.

If a subject, or its parent subject, is included in the subscriber's profile and the subscriber has the privileges of consuming the contents of the message at the given time, the listener's publishing sequence is initiated.

Multiple listeners for e.g. a fax service can be added, each handling a portion of the subscribers' database. This gives load balancing to the listener category in question as well as scalability to the entire publishing system; the workload can be projected on an array of identical listeners on the data network.

New listeners can be added to the system as new publishing channels evolve. Interfaces to new applications can easily be constructed; the open socket listens to generic XML-formatted messages and can thus be coupled with any proprietary format after message conversion.

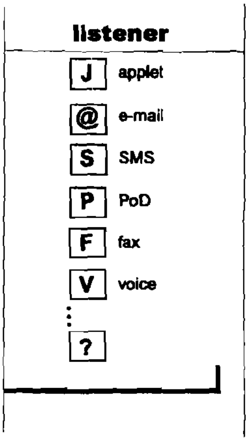

Figure 7. The third tier containing the listeners.

4.4 Subscriber tier

The construction and design of the subscriber interface will probably be the most delicate task of the system construction. It is from this tier that the flow control of information in the system is managed. A salesperson would start of with a typical sales profile, which then is personalized and augmented according to the subscriber's preferences in her profile.

It is probably a wise design choice not to let the typical user see all the subject that lie beyond the scope of her current privileges throughout the CIS system, since it may well a source of frustration and anguish when displaying a large number of inaccessible options. All the items presented on the subject menu should be those that can be consumed at the current state and time. The data entered in the subscribers' database, guided by the set of read privileges, orchestrates the process flow.

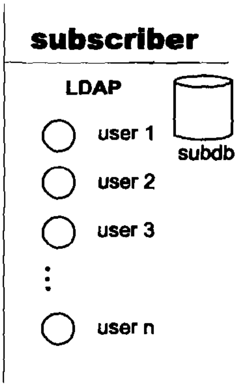

Figure 8. The subscriber tier from where the system is driven by data from the subscriber profiles.

It is therefore of utmost importance that the profile registration process is well facilitated. When a new user enters the system, a new profile is set up in the subscriber database »subdb«, either from scratch or rather starting off with a default profile constructed with the user's job description in mind. All categories can preferably be enlisted in a main directory, with information how to access these subject structures and a form for requesting additional privileges at the proper instance [figure 8].

The relational database »infodb«, where all messages are stored, along with their time to live (TTL value) becomes, over time, a cornerstone in the process of retaining a company's intellectual capital. It can be used for information retrieval purposes; free text and conceptual queries, as well as for backtracking previously published information within a subject, a project or other time-sliced studies. The topics are stored in a hierarchical structure, while the messages that are sent to the listeners are rather placed on the same level.

5. Conclusions

Concepts and techniques usually found in financial systems, such as multicasting and concurrent programming can easily find its usage in distributed publishing systems. By changing from a producer focus to a subscriber focus, using subscription profiles, information flows will become faster and more flexible. The introduction of defined trusted groups on a message subject level, the responsibility concerning information authentication and depth of intention is distributed throughout the organization. By allowing each user in the system to define preferred publishing and subscription channels herself, the information flow can be customized to each user's needs and working environment.

The chosen conceptual approach requires considerable preparations in regards to information and training for each individual using the system. However, the benefits are several and much is gained by allowing the subscribing user to the information she wants, when she wants and in her preferred publishing channel.

6. Future Work

The next natural step is to place the constructed CIS prototype in a full-scale production environment. Much as the project has been described on a technical process-oriented level, much work lies within explaining the concepts to the users in the system; the importance of submitting a well-formed set of metadata, keeping the subscription profiles updated continuously and not keeping information to themselves for personal gain.

The system and tools described in this paper are powerful in a full-scale production environment. Nevertheless, it is of utmost importance that introduction of a system such as CIS, is introduced on a management level in an organization. Otherwise, however powerful, the publishing system will not render an overall change to the better, and segregation will increase between those who are active users and those who are not. A publishing tool or system is only useful if each and every individualuses it.

Being a publisher or subscriber is not necessarily the role for a human. Any computer, machine or apparatus that is connected to a computer network can act the part with a proper API (Application Program Interface). For example, using a fast publishing system with a minimum of system latency, a »publisher« could easily be a light switch whereas the »subscribers« could be lamps, subscribing on the message »true« for turning itself on and »false« for turning itself off, if its corresponding lamp ID is called. The possibilities are vast and this technology can bring fourth new methods through which humans and machines can interact and exchange information.

Acknowledgements

I wish to extend my warm gratitude to Casper Winsnes et al, at Gear Management & Digital Media Consultants AB in Stockholm for intense interest collaboration, conceptual ideas and support. Thanks also to Professor Nils Enlund and the Interactive Institute in Stockholm.

References

[1] **Turpeinen M**, *Customizing News Content for Individuals and Communities*, Helsinki University of Technology, Laboratory of Computer Science, Helsinki, Finland

[2] **Saarela J**, *The Role of Metadata in Electronic Publishing*, Pro Solutions, Helsinki, Finland

[3] **Jonsson, A**, *Methods and Techniques for Enhancing On-line Publishing Workflows*, Royal Institute of Technology, Media Technology & Graphic Arts, Stockholm, Sweden, ISSN 1400-1853

[4] **TIBCO**, *TIB/Rendezvous Concepts*
http://www.rv.tibco.com

[5] **Goldfarb C**, *The SGML handbook*, Oxford University Press, 1990
ISBN-0-19-863737-9

[6] **W3C**, *XML 1,0*
http://www.w3c.org/TR/REC-xml

[7] **Tapscot, W**, Wired Magazine, issue 2/99
http://www.wired.com

[8] **Winsnes C**, Interview at Gear Management & Digital Media Consultants AB, Stockholm, Sweden