Experiences in Using Integrated Multimedia Streaming Services to Support E-Learning in Manufacturing Processes

Martin Zimmermann
Production Systems Engineering
University of Applied Sciences
Offenburg, Germany
m.zimmermann@fh-offenburg.de

Abstract— **Most E-Learning projects tend to separate learning activities from everyday work. This paper presents an approach where closer integration between learning and work is achieved by integrating multimedia services into manufacturing processes.**

The goal of E-Learning services integration in manufacturing is, through the development of new multimedia solutions, to accelerate and enhance the ability of manufacturing industry to capitalise on the emergence of a powerful global information infrastructure. In this paper we suggest to combine the areas of media streaming services and manufacturing processes, by providing electronic learning offerings as collections of media streaming services. The key components of our approach are 1) an xml based streaming service specification language, 2) automated configuration of distributed E-Learning streaming applications, 3) web services for searching, registration, and creation of E-Learning streaming services.

Keywords*— just-in-time learning; media streaming services; web services; manufacturing processes*

I. Introduction

Manufacturing processes involve the control and management of manufacturing systems ranging from basic assembly processes to the high-tech manufacture of pharmaceutical, telecommunications and electronic equipment. Categories for such manufacturing processes are assembly line / flow shop, and cellular manufacturing / group technology. E.g., in case of assembly line based processes a line of dissimilar machines are grouped in the line (sometimes more than one to balance flow). Innovation, productivity, flexibility, and continuous improvement are key ingredients to success in the constantly evolving world of manufacturing.

A major challenge to which companies must respond is the integration of advanced E-Learning technologies. What is actually needed is a learning "on demand", embedded into work processes, responding to both requirements from the work situation and from employee interests, a form of learning crossing boundaries of e-learning and performance support.

The goal of E-Learning services integration in manufacturing processes is, through the development of new IT solutions, to accelerate and enhance the ability of manufacturing industry to capitalise on the emergence of a powerful global information infrastructure.

Multimedia networking services support monitoring, controlling and supervising production processes in order to achieve high levels of efficiency and environmentally friendly production. The new flexibility of workers and work environments makes traditional conceptions of training in advance, in rather large units and separate from work activities, more and more obsolete. Manufacturing scenarios where E-Learning services can be integrated are shown in Fig. 1.

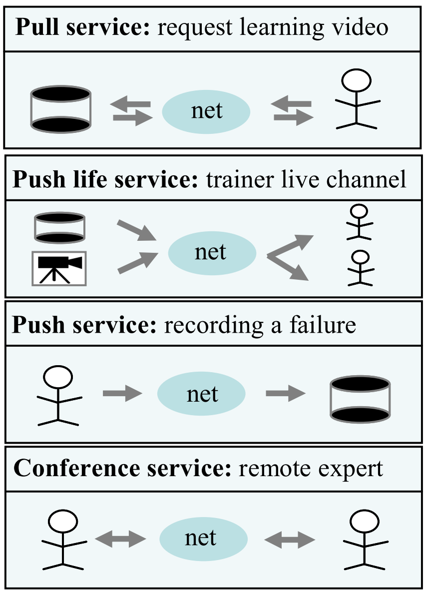

Figure 1. E-Learning services in manufacturing processes

The following streaming services can be identified:

- Pull (on demand) service: allows on demand access to remote E-Learning content, e.g. video content illustrating configuration of a CNC machine.

- Push service: e.g. a push channel from a remote expert to the manufacturing personal ("how to configure / operate a machine"). Live audio and video (expert) as well as media files can be delivered (see Fig. 1).
- Push service: for recording a failure scenario during a manufacturing process by recording the comments (voice) of a worker and by recording the machine status by video. This documentation can be used later to analyse details.
- Conference service for maintenance and remote diagnosis: teleoperation and remote diagnosis and maintenance of distant plant and production equipment can be achieved by using peer to peer E-Learning streaming services.

The remainder of this document is as follows. Section II makes a comparative digest into related work in this field of interest. In Section III we discuss a xml based language which allows specification of E-Learning streaming services while section IV outlines the architecture and describes the major components and technologies to solve the problems stated in section I. Finally, section V gives a brief summary and concludes with a note on future work.

II. RELATED WORK

[1] presents a framework for an on-demand E-learning management system that make use of broadband network for the delivery of distributed "educational activities" such as distributed courses, tutoring sessions, lectures, workshops, etc. The developped scheme is tailored towards personalized learning using distributed information in a dynamic and heterogeneous learning setting, i.e. a connected network of learning management entities and educational systems where learners are individually supported in accessing distributed resources. However it is restricted to on demand scenarios and multimedia files.

[2] introduces a framework for design and development of an interactive multimedia E-learning system for engineering training. The main goal of the project is to encourage low cost developing of effective and customized e-learning systems for engineering training by using popular and inexpensive software tools especially for **virtual simulation of engineering system operations.**

The Venice [3] project proposes a web services based framework for VoIP applications. By using a service oriented architecture, the authors aim at easing the integration of supplementary services, the compatibility between different signaling layer protocols for call control and the installation of software updates on client devices. However, the Venice project is specifically tailored to VoIP scenarios, i.e., the authors do not address reusability between different multimedia applications nor provide a generic platform for the development of these.

Based on a distributed messaging middleware, the Global Multimedia Collaboration System (Global MMCS) [4][5] provides a framework for an audio/video collaboration system, which bridges the gaps between nowadays multimedia applications by providing a common signaling protocol with gateways to existing protocols like SIP or H.323 [9]. However, the authors do not address other multimedia applications and the reuse of components between these.

In [11] a mobile streaming media CDN (Content Delivery Network) architecture is presented in which content segmentation, request routing, prefetch scheduling, and session handoff are controlled by SMIL [12] (Synchronized Multimedia Integrated Language) modification. The approach concentrates on the segmentation aspect, which is important for mobile users. In [13] also an XML based specification technique for streaming services is introduced. However, it does not enable the specification of quality of service attributes and distribution policies.

In [6] an approach is suggested to combine the areas of e-learning and Web services, by providing electronic learning offerings as (individual or collections of) Web services as well. It elaborates on this by showing how content providers and content consumers (i.e., learners) can communicate appropriately through a Web service platform with its common description, publication, and retrieval functionalities. However, it does not support live channels.

The JSR 309 [16] is designed to provide server-based Java applications with multimedia capabilities. It targets a large range of applications from simple ring-back tone applications to complex conferencing applications, by providing: media network connectivity to establish media streams, IVR functions to play/record/control multimedia contents from file or streaming server, ways to join/mix IVR function to network connection to create, conferences and call bridges. However the proposed API approach is restricted to Java based applications and it does not support enhanced VoIP functions, e.g ring groups or call queues. It mainly provides low level objects like players, recorders, mixers and connections that developers can manipulate or combine together to obtain all the multimedia capabilities.

[17] presents a generic framework for multimedia applications consisting of a set of reusable Web service components, a modeling language based on finite state automata and a compiler. The authors concentrate on the signaling plane protocols, especially their similar structure and purpose, i.e. the definition of possible states for both, client and server, and the transitions between these states via the exchange of messages.

[10] describes a service-oriented communication (SOC) paradigm based on Web services for real-time communication and converged communication services over IP. This approach extends Web services from a methodology for service integration to a framework for SOC. In particular, it introduces the generic Web services-based application session management (WS-session), the two-way full duplex Web services interaction for communication, and the development of Web Services Initiation Protocol.

III. SPECIFICATION OF E-LEARNING SERVICES

A. Model

Manufacturing processes are composed of manufacturing resources, e.g. manufacturing automation devices, equipment & machinery, material & manufactured parts, and manufacturing personnel. E-Learning streaming services are related to such manufacturing processes and / or single manufacturing resources. E.g. for a single machine a video on demand can be requested by the manufacturing personal to study the detailed configuration steps for such a machine.

The class diagram in Fig. 2 represents a model of our E-Learning streaming services (UML class diagram). Different fundamental components of the system are shown in the form of classes and relationships.

An E-Learning service can be described by its media objects, related manufacturing resource, distribution and replication policy, and the quality of service used for content delivery. Media streams can also be categorized according to how the media objects are delivered. In pull services data delivery is initiated and controlled from a client whereas in push services a server initiates data transfer and controls the flow. Conference services support scheduled and ad-hoc conferences. Such conferences contain the following attributes: date and time when the conference will take place, the invited participants and as an optional part the leader, the agenda as well as conference documents.

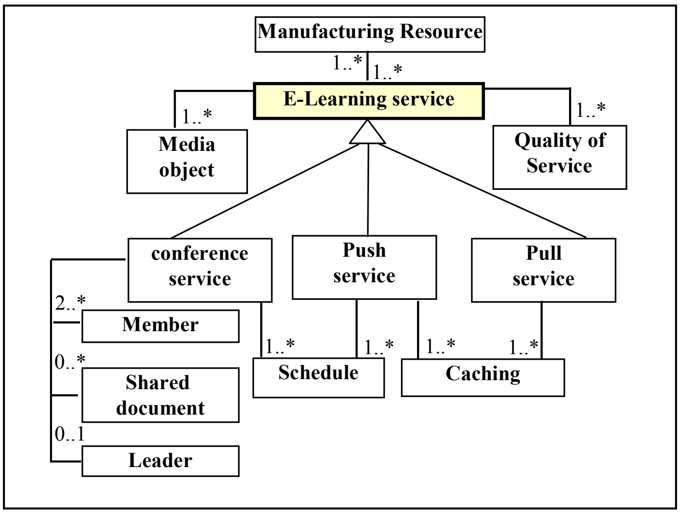

Figure 2. E-learning service model

B. Use Case: Life training

The following scenario (Fig. 3) illustrates one of our implemented use cases, a technical training (how to configure and operate a machine) for a computerized numerical control machine (CNC) as the related manufacturing resource: The operator shows the handling of a machine to the distributed audience. A background speaker (in a separate location, e.g. office) explains the configuration steps. To enable the interaction between the speaker and the learning community, each participant is offered an audio back channel. In the scenario there are different QoS requirements, e.g. for the video transmission showing the manufacturing machine a high resolution and frame rate is required.

The specification of media objects is based on SMIL [12]. SMIL (Synchronized Multimedia Integration Language) is an XML-based language, which facilitates the construction of accessible multimedia applications for the internet and mobile devices. Collections of XML elements and attributes can be used to describe the temporal and spatial coordination of one or more media objects.

In addition to SMIL, the type of media objects (live / file), the delivery policy and the roles which consume and supply media objects can be specified.

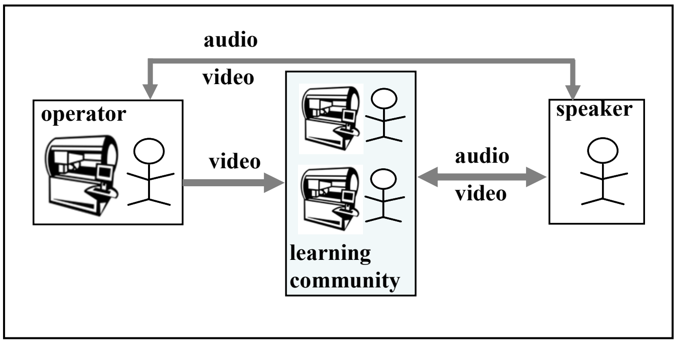

Figure 3. Example scenario

C. XML Specification

A specification of an E-Learning service is a structured composition of autonomous objects (see Fig. 4):

- resources or manufacturing processes (category, type, keywords)
- a collection of media objects, e.g. audio and video objects (life or files)
- distribution policy, e.g. push or pull
- roles and related media objects, e.g. speaker

This information can be used later, e.g. by the manufacturing personal, for searching and accessing E-Learning services.

Fig. 4 illustrates the specification of the life training scenario introduced in subsection A. The first part contains the description of manufacturing resources related to the E-Learning service. The list of media objects which are used in the service is part of the section "MediaObjects" together with the type, bitrate and source. The media streams are delivered based on a push policy. This includes also the definition when the media objects must be delivered (date and time), together with the available bit rates. Finally the roles, e.g. a speaker role or a operator role are introduced together with a list of media streams which are consumed and supplied by each role. E.g. the speaker consumes and supplies video as well as audio objects referenced in the section MediaObjects.

```
<StreamingService>
<ManufacturingResources>
   <resource  category = "num control machine" />
   <keywords>live training </keywords>
</ManufacturingResources>

<MediaObjects>
  <video id = "v1" type = "live"
                bitrate= … src =  … />
  <audio id = "a1 type = live"  src = …/>
   . . .
</MediaObjects>

<DistributionPolicy>
  <push codec="MPEG4" date= …  time=…
        recording="yes"
  </push>
</DistributionPolicy>
<Roles>
   <role name="Speaker"/>
     <consumerOf>
        <video id=… />
        <audio id=… />
   </role>
   . . .
</Roles>
. . .
```

Figure 4. E-Learning service specification

IV. SERVICE ORIENTED ARCHITECTURE

A service-oriented architecture (SOA) is basically a set of services interacting with each other and coordinating some activity. Service providers and service consumers are the two main entities acting on behalf of a user. The Web service technology additionally addresses a standardized description of a service's functionality using an XML dialect [14]. Using the Web Service Description Language (WSDL) [7], a service provider describes the functionality (interface) of a service in a platform, language, and operation system neutral way while a service requestor talks to these services using SOAP over HTTP (or other transport protocols).

A. E-Learning Streaming Services

Streaming services are managed by a set of web services (see Fig. 5). Such a web service is a URL-addressable software resource that performs operations and provides answers. In our case, the operations offered by the service interface are:

- searchStreamingService: allows searching a streaming service (based on the elements and attributes which form the xml service specification (e.g. category of a manufacturing resouce, type of media object, distribution policy, etc.).
- registerStreamingService: enables validation and registration of new streaming services.
- createStreamingService: instantiates streaming software components and establishes interconnections between components.
- startStreamingService: the operation start is typically invoked by a scheduler (in case of push based services, driven by date and time values).
- subscribeToStreamingService: allows users to subscribe to (future) streaming services (e.g. a remote video based training related to a certain CNC machine). When a new media streaming service is registered or started, a user will be notified automatically by the service manager (as soon as such a service has been registered to the xml database).

After searching a streaming service, the client (e.g. RTSP media player) sets up three network channels with the RTSP server when media data is delivered using the RTP over UDP, as shown in Fig. 5. A full-duplex TCP connection is used for control and negotiation. A simplex UDP channel is used for media data delivery using the RTP packet format. A full-duplex UDP channel called RTCP is used to provide synchronization information to the client and packet loss information to the server. RTSP initiates and controls delivery. The XML service description includes references to media streams, using the URL method rtsp.

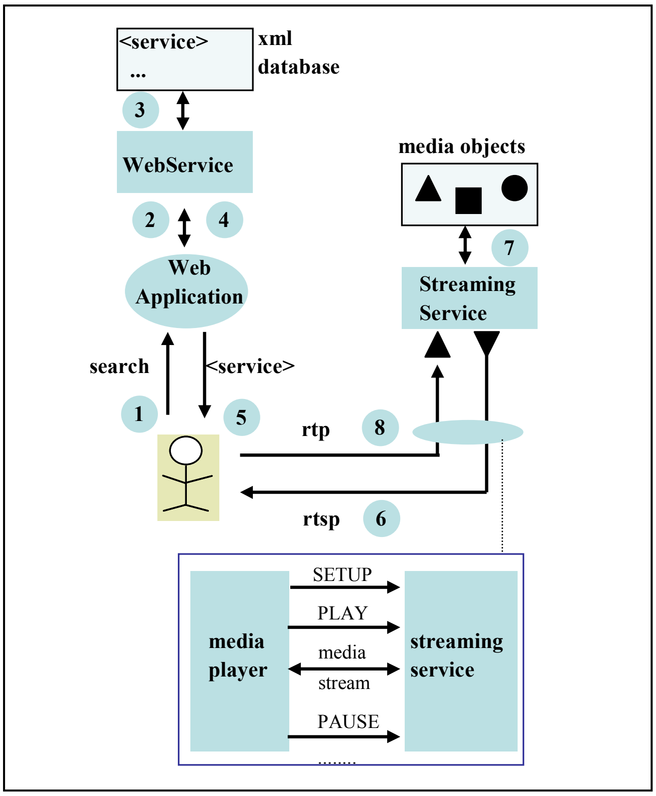

Figure 5. E-Learning Web services

According to our service specification language, the following properties of E-Learning services can be used as part of a query:

- manufacturing resources, e.g. category or keywords
- media objects properties, e.g. recorded live training
- distribution policy, e.g. on demand service

- QoS properties, e.g. bitrate, used codec
- Date / time duration attributes

B. Prototype

Our approach is based on a clear separation of a streaming service specification - and its implementation by a distributed application and can be used for different streaming paradigms, e.g. push and pull services.

The following figure (Fig. 6) illustrates the user interface and the management interface. The services are managed by a service manager, which provides in the current implementation a simple interface to search, and start (request) services. Moreover, a management interface enables creation of new services, and deletion of existing services. Service specifications are stored as XML documents in a XML database.

A new e-Learning streaming service specification is first analyzed by a web service (operation create) . Driven by a component library which contains existing streaming components, such as encoders, media servers, etc., and a set of configuration rules, a web service creates a distributed streaming application configuration. Based on such a service specification, the service manager also supports retrieval of existing services.

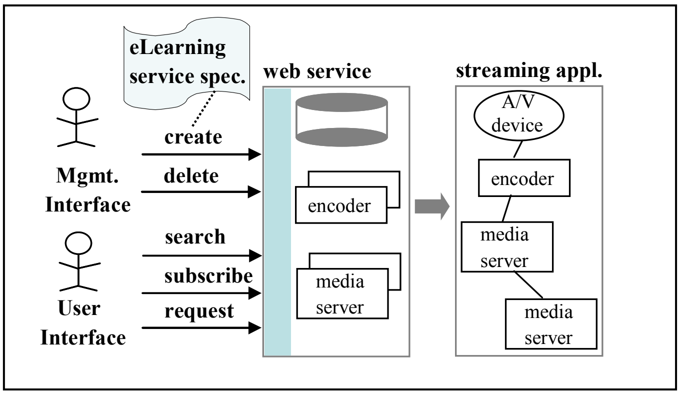

Figure 6. User interface and management interface

Push services are started automatically according to the specified date and time values. Starting a service means to create the required streaming components, e.g. an encoder and a media server component as well as to establish the communication relationships. Before a new service is created, a consistency checker is responsible to test the availability of nodes, and media objects according to the definition as part of a service specification. A service specification is then analyzed by a web service, which will select appropriate streaming software components.

Multimedia services are stored as XML documents in a XML database. We use Apache's Xindice, which is a database designed from the ground up to store XML data or what is more commonly referred to as a native XML database. Advantages are: faster for XML than other databases, no mapping to relational required, quick fragment retrieval provided, and optimized XML querying supported.

The prototype has been implemented using the PHP programming language and the PHP Web Services Development Pack nusoap [8] for the creation of and access to web services. We use the existing media server components from RealNetworks, and the OSS software Asterisk [15] as a VoIP server. The first (subjective) test results based on implementation of push and pull services as well as conference management are very promising. However, the used OSS VoIP system Asterisk does not offer scheduled conferences, i.e. a direct mapping to the Asterisk conference management functions is not possible. Information related to scheduled conferences are stored in a xml database. As the prototype is still under development, an objective measurement of processing time and delay has not yet been made.

C. User Experience

In a production environment robust and easy to use search functions are one of the key requirements. Our original search functions had to be improved. Context-driven search results, driven by the concrete manufacturing resources which are part of a single work place are regarded to be important for the machine operators. E.g. depending on the available machine(s) which are part of a single work place, the search functions have to offer the "right" E-Learning services related to the given local manufacturing resources.

The most accepted E-Learning streaming services are on demand video services. From the production management point of view, push services are essential for quality improvements. Such services allow a "just in time" documentation of occurred failures, during runtime of a machine by ad hoc video and audio recordings. These recordings can be analyzed later and appropriate solutions can be developed to improve manufacturing processes.

V. CONCLUSION AND FUTURE WORK

In this paper we have presented a service oriented architecture for E-Learning streaming services – realized using web services technology. The major contributions and extensions of our approach aim to provide a high level specification of E-Learning services in terms of user roles, media objects, distribution policies, etc.. Based on such a service specification, a web service supports automated creation of tailor made media streaming applications, based on existing software components, e.g. media server or VoIP software components.

We report the current status of our prototype. The prototype has shown that our XML based language is well suited for automatic generation of implementations. At implementation level, the different aspects are integrated in a general object-oriented architecture supporting modularity and reuse of software. The deployment of new service operations is very easy to accomplish due to the modular structure of the service oriented architecture and the well designed and easy to use PHP development pack nusoap.

We do not use JMF [18], because it is not stable enough, especially for long running streaming applications. Instead, we use existing media streaming components (e.g. media server from RealNetworks).

Future work will concentrate on the extension of the service model to support additional functionality, e.g. application sharing. Another important aspect will be the integration of authentication services (authentication of users).

The proposed framework is part of a multimedia project, financed by the state government of Baden-Württemberg in Germany.

REFERENCES

[1] B. Khasawneh and S, A. El-Seoud: Framework for on-demand e-learning resources allocation and distribution: OERAD International Journal of Computing & Information Sciences, Vol. 4, No. 2, 2006

[2] D. Borissova and I. Mustakerov: A Framework of Multimedia E-Learning Design for Engineering Training, ICWL 2009, in LNCS 5686, pp. 88–97, 2009, Springer, Berlin Heidelberg 2009

[3] M. Hillenbrand, J. Götze, and P. Müller. Voice over IP - Considerations for a Next Generation Architecture. In EUROMICRO-SEAA, pages 386–395. IEEE Computer Society, 2005.

[4] A. Uyar, W. Wu, H. Bulut, and G. Fox. Service-Oriented Architecture for a Scalable Videoconferencing System. In Proc. of IEEE Int. Conf. on Pervasive Services 2005, pages 445–448. IEEE Computer Society, 2005.

[5] W. Wu, G. Fox, H. Bulut, A. Uyar, and T. Huang. Service Oriented Architecture for VoIP conferencing.In Int. Journal of Communication Systems, Special Issue on Voice over IP - Theory and Practice, volume 19, pages 445–462. John Wiley & Sons, 2006.

[6] G. Vossen, P. Westerkamp: "E-Learning as a Web Service," ideas, Seventh International Database Engineering and Applications Symposium (IDEAS'03), 2003

[7] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana: Web Services Description Language (WSDL) 1.1, 2001. http://www.w3.org/TR/wsdl.

[8] http://sourceforge.net/projects/nusoap/

[9] ITU. Recomm. H.323 (1999), Packet-base multimedia communications systems.

[10] L-L. Wu Chou , L. Feng :Web services for communication over IP, In: Communications Magazine, IEEE, Volume: 46, Issue: 3, 136-143, 2008

[11] T. Yoshimura.: Mobile Streaming Media CDN Enabled by Dynamic SMIL, WWW2002, May 7-11, 2002, Honolulu, Hawaii, USA, ACM 1-58113-449-5/02/0005

[12] SMIL: http://www.w3.org/TR/smil20/

[13] ipdr.org: Service Specification – Streaming Media (SM), www.ipdr.org

[14] D. Fallside. XML Schema Part 0: Primer, 2001. http://www.w3c.org/TR/xmlschema-0.

[15] Asterisk: The Open Source Telephony Project, http://www.asterik.org

[16] T. Ericson, M. Brandt: JSR 309 - Overview of Media Server Control API, Public Final Draft, Media Server Control API v1.0, 2009

[17] A. Scholz et al: WS-AMUSE - web service architecture for multimedia services, International Conference on Software Engineering, In Proceedings of the 30th international conference on Software engineering, Leipzig, Germany, 2008

[18] http://java.sun.com/javase/technologies/desktop/media/jmf/index.jsp