Concept of a service-oriented integration architecture for the orthopaedic operating theatre

B. Ibach[1], A. Zimolong[2], C. Bullita[3] and K. Radermacher[1]

[1] Chair of Medical Engineering, Helmholtz-Institute for Biomedical Engineering of the RWTH Aachen University, Aachen, Germany
[2] Department of Orthopaedics, Ruhr-University Bochum, Germany
[3] Siemens AG, Medical Solutions, Germany

Abstract

In modern operating theatres there are an increasing number of technical devices. To enhance possibility of exchange and integrated use of information as well as to support the surgeon during diagnosis and treatment more and more integrated medical operating systems (IORS) are available. Commercial IORS are mainly proprietary developments using usually proprietary communication standards and interfaces, which reduce the possibility to integrate devices from different vendors. To overcome these bottlenecks, there is a need for an open standardized architecture, which is based on standard protocols and interfaces, to be able to integrate devices from different vendors based of heterogeneous software- and hardware-components. The paper presents a concept of such an integration architecture for the OR based on a service-oriented architecture (SOA) and using standardized communication protocols and interface descriptions. A short introduction to the concept of SOA is given, followed by the presentation of the SOA based reference integration concept.

1 Introduction

Nowadays integrated medical operating systems (IOR) are available from many different manufacturers like Brainlab, Olympus, Storz, Stryker and Wolf. All these systems are proprietary developments, which are to some degree modular, but modules from third-party manufacturers can only be integrated in cooperation with the system vendor. Regarding the communication between the different modules, often proprietary protocols are used, like the CAN-based SCB by Storz.

As processes and related devices keep on changing, the risk of not reaching the ROI is high if the IOR is not open for integrating new devices. Limited flexibility and modularity for third party components thus are major bottlenecks of currently available integrated OR solutions. However, standardized and open interface definitions are still missing. Furthermore, legislation and risk-assessment prevent open IOR, as the operator becomes a manufacturer in case of information coupling two different medical devices [4].

While legislative aspects are about to be tackled by the new IEC 80001, standards for a plug and play platform complementing the IEC 80001 are needed. Therefore interfaces and communication protocols for inter-device communication and central control of all devices must be developed.

Based on the state of the art, an integration architecture has been developed within the framework of the OrthoMIT project and the FUSION initiative. Important aspects are modularity and different levels of integration. Modules should be exchangeable in order to allow the use of the most adequate module for a certain tasks. Furthermore, there should be the possibility to use different levels of integration for devices, as many systems should also function in stand-alone mode.

2 Concept

Basis for the development has been the analysis of IT integration techniques and -concepts used in other technological fields, like automation technology and business process management. These are dealing with the integration of software applications and devices from various manufacturers and varying soft- and hardware configuration.

As processes and related devices are changing over time, business process management becomes an important key success factor. In many companies heterogeneous soft- and hardware is supporting a variety of processes, resulting in numerous interfaces needing to be maintained. To cope with an increasing need for change, business process integration is more and more based on the service-oriented architectures (SOA) paradigm [8].

Looking at hospitals, there is a similar situation compared to industry, resulting in a growing need for business process management. Along the clinical pathway of the patient, a number of different devices are used for diagnosis and treatment, providing the surgeon with services like patient information, medical imaging, position tracking or navigation during i.e. joint replacement surgery etc. Therefore an integration concept for the operating room (OR) needs to take into account the different services within an OR and should encompass their easy and flexible adaptation to satisfy the varying requirements of the different surgical interventions and processes.

SOA may generally fulfill the needs mentioned above for an OR integration architecture. Since SOA describes only the global concept for distributed service based integration, a design model for an integrated medical service-oriented architecture has been developed. In the following we will first give a short introduction to a concepts of service oriented architecture and then present the concept of a service-oriented OR integration architecture.

3 Service oriented architecture

SOA is a paradigm for organizing und utilizing distributed capabilities and resources that may be under the control of different ownership domains [7]. The capabilities and resources are implemented as reusable, technical and functional loose coupled, distributed services, which can be used by other components or services.

Figure 1 shows the general communication and entity model of a service-oriented architecture. The basic procedure for interaction using the find, bind and execute paradigm is as follows:

1. Service-providers **register / publish** their services by the service-manager
2. Service-consumer try to **find** services needed in the registry of the service-manager
3. Service-consumer and service-provider interact with each other
    a. The service-consumer **binds** to a service-provider using the contract published
    b. Functions requested by the service-consumer are **executed** by the service-provider and the results are transmitted back

The functionality of services is accessed by interface-descriptions, i.e. the service contract, which is a specification of the way a consumer of a service will interact with the service provider in an implementation independent way. It specifies the format of the request and response from the service. A service contract may require a set of pre- and postconditions. Pre- and postconditions specify the state that the service must be in to execute a particular function. The contract may also specify quality of service (QoS) levels, i.e. specifications for the non-functional aspects of the service.

Complex processes are realized by arranging and coupling of services. Due to the defined interface-description, processes can easily be changed and adapted to new requirement and services can be reused by other components of the architecture [11].

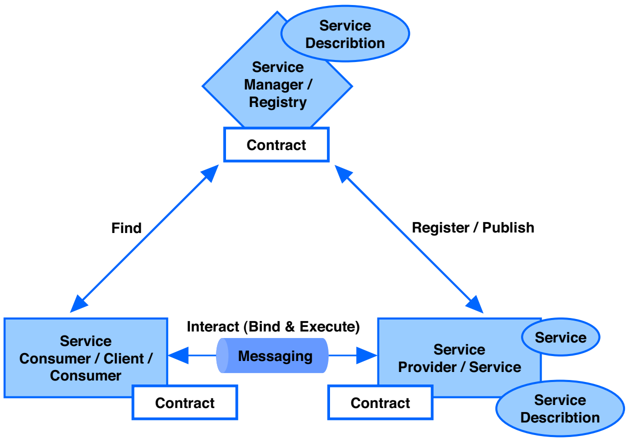

Figure 1 Generic communication model of a service oriented architecture using publish, find, bind, execute paradigm [**9**]

SOA in general is technology independent, as it describes a paradigm and not a precise realization. Common technologies for realization are EJB, CORBA or open standards like SOAP, XML and Web-Services (WSDL), which provide means for implementing an open, manufacturer independent integration architecture based on SOA.

4 OR Integration Reference Architecture

Based on the OASIS “Reference Model for Service-Oriented Architectures” [7] a reference architecture has been develop, consisting of the following components:

- Service-Manger
- Service-Provider
- Service-Consumer
- Workflow-Engine
- Event-Manager cp. [5]
- Communication-Server

The overall architecture model can be seen in Figure 2.

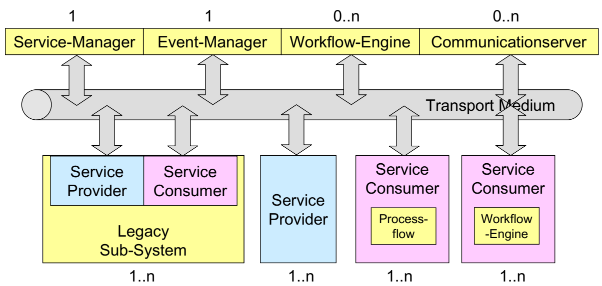

Figure 2 OR Integration reference architecture for a SOA based integration in the operating room

Service-Manager
The Service-Manager (SM) is the central component of the reference architecture. Service-Providers have to reg-

ister their services in a registry provided by the Service-Manager, so that Service-Consumers can find needed services within the network. Access control is another important task of the Service-Manager. It is important to have access control for services to avoid misuse and unauthorized access to data using provided services. The access control list of the service manager will be managed by an IT-Network-Integrator as established with the upcoming IEC 80001 [2].

Event-Manager (EM)
The Event-Manger is a central entity handling events released by Service-Providers. To use the Event-Manger the Service-Providers have to register their events, while Service-Consumers on the other side can subscribe to be notified of certain events. Events received by the Event-Manger will be forwarded to the subscribed Service-Consumers. Registration to the Event-Manger is not mandatory, as Service-Providers might be able to manage the distribution of events on their own. The Event-Manger may be realized as specified in [5].

Communication-Server / Connector
A connector transforms data formats and is used for the integration of components which cannot provide data in a standardized format natively. Proprietary data formats must not be transformed into another proprietary data format. The output of a connector has to be the open, standardized XML format or an established and supported data standard for medical devices. Connectors are maintained by their manufacturer and are normally part of a medical device. Multiple connectors may be integrated into a Communication-Server, whose output is again standardized, whereas the input communication is performed with a proprietary format.

Workflow-Engine (WE)
Within a Workflow-Engine (semi-) automatic, computer assisted workflows are represented, which are processed in a defined way [3]. One ore more process entities are managed by a WE and are supervised using return data of the sub processes. In contrast to simple flow control as it is implemented in software, a WE orchestrates the different applications of various system manufacturers.

Service-Provider
A Service-Provider is a device that offers the use of capabilities by means of a service. Functions and data of the device are encapsulated using an XML based interface description, like WSDL. A Service-Provider can offer access to data, like patient-information, images etc. or offer direct access to functionality (OR-Table control).

Service-Consumer
A Service-Consumer uses the capabilities offered by means of services provided by a Service-Provider.

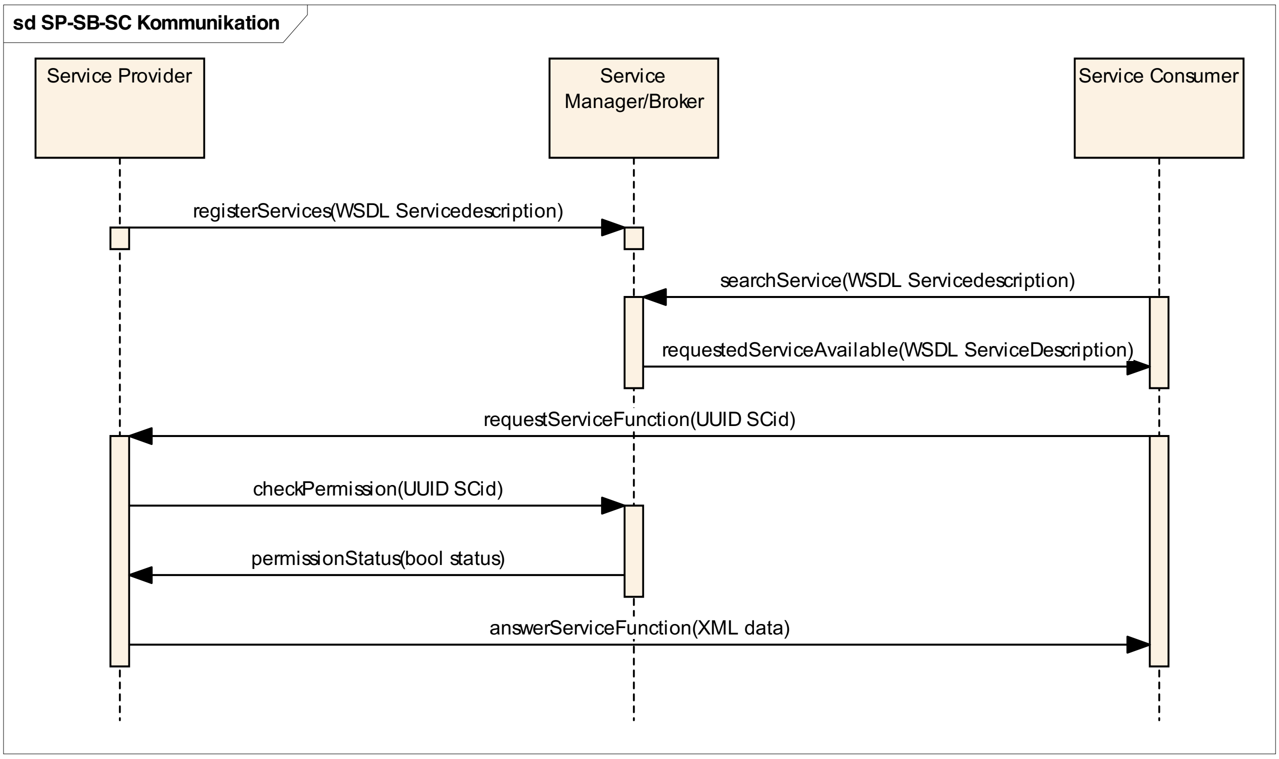

Figure 3 Example of Service-Provider and Service-Consumer communication with authorization using the Service-Manager

The single components are connected through a shared transportation and communication medium like Ethernet or similar. The basic component is the Service Manager (see above), which is responsible for access control and information management. The different medical devices are either service provider, service consumer or both, providing or using services available within the network. Complex components can be integrated as “legacy sub-systems”, offering only part of their functionality using a Communication Server or special Connectors. Event-Manager and Workflow-Engine are optional components which can be used to extend functionality and for more complex scenarios.

Figure 3 shows an example of the communication between Service-Provider (SP), Service-Consumer (SC) and Service-Manager(SM), which illustrates the role of the Service-Manager. The following steps are performed:

1. The SP registers its services at the SM.
2. A SC searches for a specific Service and receives a positive or negative reply from the SM.
3. The SC requests the service from the SP.
4. The SP checks with the SM if the SC has permission to use the service. For a positive answer, the service is executed. Otherwise, an error message is send back to the SC.

Data exchange is based on an open, standardized XML format or established data standards for medical information exchange like DICOM, HL7, IHE etc. Common open standards for an XML based SOA are WSDL, SOAP.

WSDL stands for “Web-Service Description Language”, which is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define an endpoint. Related concrete endpoints are combined into abstract endpoints (services). WSDL is extensible to allow description of endpoints and their messages regardless of what message formats or network protocols are used to communicate. However, the only bindings described in this document describe how to use WSDL in conjunction with SOAP, HTTP GET/POST, and MIME [1].

SOAP is a lightweight protocol intended for exchanging structured information in a decentralized, distributed environment. It uses XML technologies to define an extensible messaging framework providing a message construct that can be exchanged over a variety of underlying protocols. The framework has been designed to be independent of any particular programming model and other implementation specific semantics [10].

4.1 User-Interface Integration

Besides the information exchange architecture the integration of graphical-user-interfaces (GUI) is also important. GUIs might be integrated as Services in the same way as “normal” medical devices. Visualisation devices integrate and process services offered by components of the reference architecture.

The bandwidth requirements of graphical-information are an important factor to be considered. For the simple display of e.g. alphanumeric values only few data have to be transferred as opposed to animations and video were Mega- and/or Gigabytes of data have to be transmitted. Conceivable approaches, based on the combination of soft- and hardware-components will be as follows:

- Hardware based integration using KVM (Keyboard, Video, Mouse) Switches
- Software based integration using remote access software like VNC, RDP etc.

Both approaches have pros and cons. Hardware based approaches are suitable for the visualisation of 3D-accelerated content or videos due to the high data transfer rates, while software based approaches have problems displaying 3D and video information, due to the missing hardware acceleration and bandwidth limitations. On the other hand the cost for the KVM hardware and infrastructure are quite high, while the VNC or RDP are using the same infrastructure used for data and information exchange.

5 Discussion

A concept of a service-oriented integration architecture for orthopaedic operating theatres has been presented, which aims to provide means to integrate medical devices in a modular and flexible way using open standards and standardized interface descriptions. In order to allow the modular integration of medical devices from different manufacturers, standardized service descriptions for different types of devices (OR-Tables, Navigations-Systems, Tracking-Systems etc.) as well as standards for event-classes for the different devices have to be defined, which will be objective of future research.
Furthermore the security of services and authentication procedures have to be evaluated, regarding existing standards and needs for the application in a medical / OR environment.
Regarding the user-interface integration further research is necessary to optimize the integration and data-transfer. Possible solutions could be the extension of the software-protocols (VNC, RDP, etc.) to enhance data transfer rates and hardware acceleration of 3D information.

Other important aspects for the further development of the concept and its realization are semantics and syntax and risk-assessment, besides the technical concept for the integration of medical devices and data-exchange. Regarding the semantics and syntax, the “Therapy Imaging and Model Management System“ TIMMS has been presented which addresses semantics and ontology for medical workflows and data-exchange [6].
Concerning the risk-assessment a first risk-analysis will be performed, taking into account the upcoming IEC 80001 Standard, regarding the Risk-Management Of IT-Networks Incorporating Medical Devices [2].

6 Acknowledgement

The OrthoMIT project is funded by the German Federal Ministry of Education and Research (BMBF 01EQ0402)

7 Literature

[1] Erik Christensen, Francisco Curbera, Greg Meredith, and Sanjiva Weerawarana. Web services description language (wsdl) 1.1. Technical report, W3C, 2001.

[2] Sherman Eagles and IEC. Application of risk management to information technology (it) networks incorporating medical devices. 1st Committee Draft, 2007.

[3] D. Hollingsworth et al. The workflow reference model. *The Workflow Management Coalition*, 1, 1995.

[4] B. Ibach, A. Zimolong, F. Portheine, F.U. Niethard, and K. Radermacher. Integrated medical workstations for computer integrated smart surgery (ciss) - state of the art, bottlenecks and approaches. In *International Journal of Computer Assisted Radiology and Surgery*, volume 1, pages 449–451. Springer, 2006.

[5] IBM. Patterns for e-business (application integration). Technical report, IBM, 2005.

[6] Heinz U. Lemke. The operating room and the need for an it infrastructure and standards. *Int. J. CARS*, 1(3):117–121, November 2006.

[7] CM MacKenzie, K. Laskey, F. McCabe, P. Brown, and R. Metz. Reference model for service oriented architecture 1.0. *OASIS Committee Draft*, 1, 2006.

[8] F. Menge. Service-orientierte architektur (soa) verglichen mit komponentenmiddleware und enterprise application integration (eai), 2006.

[9] Microsoft. Messaging patterns in service-oriented architecture, part 1. Online, April 2004.

[10] N. Mitra et al. Soap version 1.2 part 0: Primer. Technical report, W3C, 2007.

[11] R. Perrey and M. Lycett. Service-oriented architecture. *Applications and the Internet Workshops*, pages 116– 119, 2003.