Monitoring and Diagnosis of Networked Medical Hardware and Software for the Integrated Operating Room

Stefan Bohn, Michael Lessnau, Oliver Burgert
Innovation Center Computer Assisted Surgery (ICCAS), Medical Faculty, University of Leipzig, Germany
stefan.bohn@medizin.uni-leipzig.de

1. Purpose

Interoperability of heterogeneous medical devices, clinical information systems and components of computer assisted surgery (CAS) has been recognized for its potential to improve the overall clinical workflow as well as ergonomic conditions by centralized access and control of the integrated system. However, the installation of an integrated IT infrastructure with additional computer hardware, software, and network components heavily increases the overall technical complexity within the operating room (OR). The life critical domain within the OR demands safe and reliable operation of the integrated OR components. Therefore, an appropriate technical supervision framework is required that supports high confident functionality by facilitating systems monitoring and diagnosis of the networked hardware and software. System failures, network bottlenecks or unstable conditions should be detected to enable appropriate interventions and mitigation strategies.

2. Methods

The structural design of our modular OR integration infrastructure follows the Therapy Imaging and Model Management System (TIMMS) meta-architecture, which was published by Lemke and Vannier in 2006 [1]. Our prototype TIMMS implementation interconnects standard CAS components such as tracking, PACS, display and video routing as well as navigation, patient modeller, workflow software, and the central surgical display. In contrast to existing proprietary integration solutions, we are focussing on the development of an open architecture using standard communication protocols (e.g. DICOM, RTP, SNMP, ZeroConf, TCP/IP) and standard network technologies such as Ethernet. The integrated system has a central management unit, the TIMMS Component Controller (TCC), which facilitates service discovery, session management, time synchronization and component control. The

TCC also implements the supervisory control and data acquisition (SCADA) module (Figure 1), which realizes systems monitoring and supervision functionality for the entire OR network on three levels: 1. Network Backbone Hardware, 2. Computer Hardware, and 3. Software Applications. The diagnostic information elements from the supervised systems are maintained in the form of Management Information Base (MIB) (RFC 1450) objects. The MIB tree index structure describes the hierarchical order of all monitored variables that can be obtained or modified, their data types and by which operations they can be accessed. Management agents handle the access to the MIB objects between managed components and the SCADA module using the Simple Network Management Protocol (SNMP).

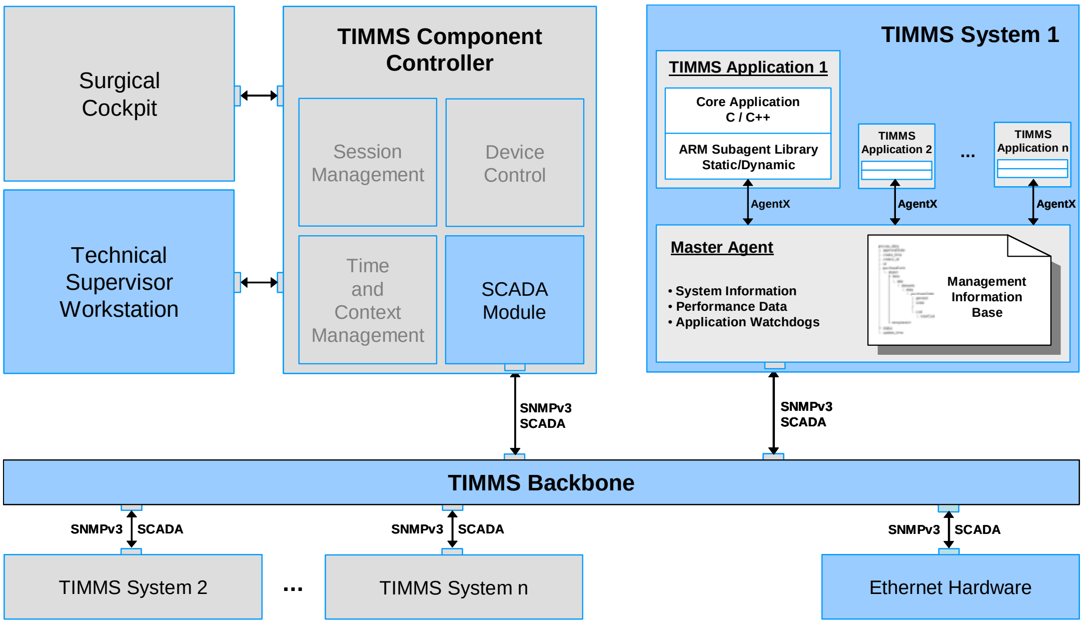


Figure 1: Supervisory control and data acquisition (SCADA) architecture for the integrated TIMMS OR network.

2.1 Simple Network Management Protocol

The transfer of diagnostic and management information among management agents and the SCADA module is based on the SNMP protocol standard (RFC 3411 – 3418), which is an application layer protocol within the OSI model. The SNMP network protocol is based on TCP/IP and thus directly applicable within the TIMMS network environment. SNMP provides few operations on MIB items such as SNMP GET and SNMP SET for retrieval or change of a MIB variable as well as SNMP TRAP for unsolicited event notifications sent by an agent to a management application.

2.2 Monitoring of Hardware Components

Diagnostic information from network and computer systems is gathered by hardware management agents, which reside in the supervised components. Most of today's standard network hardware such as routers or bridges already implement SNMP agents, which maintain access to configuration information and network traffic parameters (Level 1). Using the *Agent++* Library [2], we developed a master agent for personal computers (PC) that acquires system and surveillance information from each TIMMS PC connected to the network (Level 2). These master agents obtain performance indicators such as CPU usage, hard disc load, physical and virtual memory load as well as network interface card traffic and translate these into the corresponding SNMP MIB objects (Figure 1).

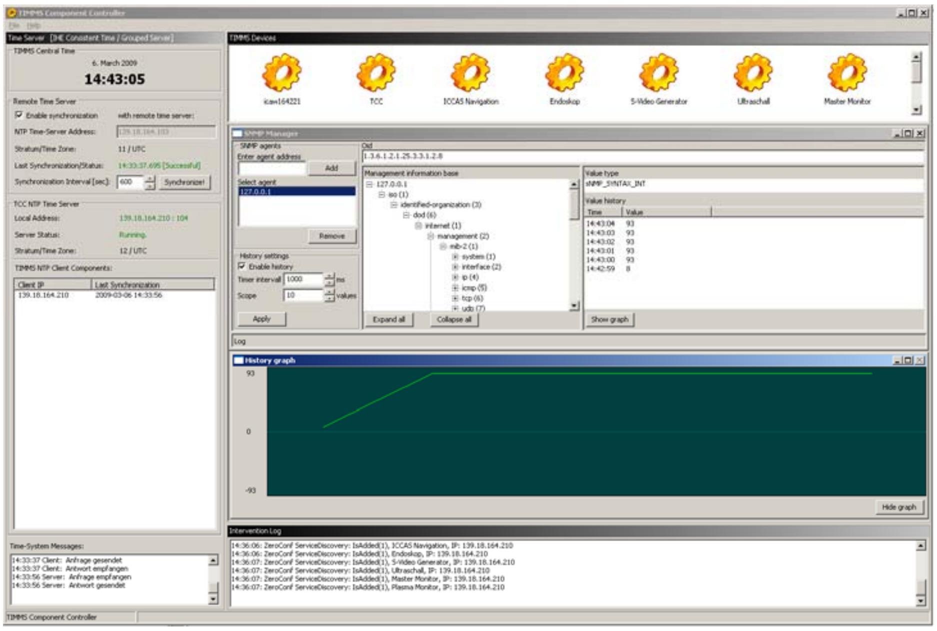
2.3 Monitoring of Software Applications

Monitoring at code implementation level (Level 3) is facilitated using the Open Group Standard "Application Response Measurement - ARM 4.1" [3]. The ARM interface standard accommodates bindings for the programming languages C and JAVA. ARM enables the measurement of application performance by introducing transactions as "units of work". The application calls the ARM API before a transaction starts, optionally an update during processing, and after it ends. We defined two custom transaction types for the SCADA framework: 1st) Transactions for measuring the duration of (critical) code sections and 2nd) Transactions that periodically report update events for the monitoring of the application's alive status. To integrate the ARM functionality into the SCADA framework, we implemented an ARM compliant subagent library that communicates with the corresponding Level 2 master agent using the Agent Extensibility (AgentX) Protocol (RFC 2741). Upon application start, the subagent registers the application and all transactions at the master agent, which creates the corresponding information items within its MIB. Watchdog alive heart beats from the application are processed by sub- and master agent and deployed using SNMP TRAP events to the SCADA module, which supervises the alive status of TIMMS applications.

2.4 TIMMS Component Controller & SCADA Module

Automatic Configuration and Plug-and-Play Service Discovery of TIMMS components are realized using the ZeroConf protocol [4]. Whenever a TIMMS component joins the network, the TCC connects to the component's master agent and retrieves the corresponding management information base object. By passing through the MIB tree elements, the SCADA module periodically queries the particular diagnostic information elements from the master- and subagent. The diagnostic information is processed and analyzed, e.g. to raise alarms if a previously defined threshold for a given element is exceeded. The TCC also provides a database logger, which stores all alarms and events for documentation purposes.

The resulting information from the SCADA module is visualized on different workstations according to the particular user group. Clinical users obtain an intuitive view of the overall system status at the surgical cockpit while the technical supervisor has access to all in depth information and control over configurable elements (Figure 2).



TIMMS Central Time
6. March 2009
14:43:05

Remote Time Server
☑ Enable synchronization with remote time server:
NTP Time-Server Address: 139.18.164.103
Stratum/Time Zone: 11 / UTC
Last Synchronization/Status: 14:33:37.695 [Successful]
Synchronization Interval [sec]: 600 Synchronize!

TCC NTP Time Server
Local Address: 139.18.164.210 : 104
Server Status: Running.
Stratum/Time Zone: 12 / UTC

TIMMS NTP Client Components:

Client IP	Last Synchronization
139.18.164.210	2009-03-06 14:33:56

SNMP agents
Enter agent address
Select agent
127.0.0.1

History settings
☑ Enable history
Timer interval 1000 ms
Scope 10 values

Oid
1.3.6.1.2.1.25.3.3.1.2.8

Management information base
```
⊟ 127.0.0.1
  ⊟ iso (1)
    ⊟ identified-organization (3)
      ⊟ dod (6)
        ⊟ internet (1)
          ⊟ management (2)
            ⊟ mib-2 (1)
              ⊞ system (1)
              ⊞ interface (2)
              ⊞ ip (4)
              ⊞ icmp (5)
              ⊞ tcp (6)
              ⊞ udp (7)
```

Value type
sNMP_SYNTAX_INT

Value history

Time	Value
14:43:04	93
14:43:03	93
14:43:02	93
14:43:01	93
14:43:00	93
14:42:59	8

Time-System Messages:
```
14:33:37 Client:  Anfrage gesendet
14:33:37 Client:  Antwort empfangen
14:33:56 Server:  Anfrage empfangen
14:33:56 Server:  Antwort gesendet
```

```
14:36:06: ZeroConf ServiceDiscovery: IsAdded(1), ICCAS Navigation, IP: 139.18.164.210
14:36:06: ZeroConf ServiceDiscovery: IsAdded(1), Endoskop, IP: 139.18.164.210
14:36:07: ZeroConf ServiceDiscovery: IsAdded(1), S-Video Generator, IP: 139.18.164.210
14:36:07: ZeroConf ServiceDiscovery: IsAdded(1), Ultraschall, IP: 139.18.164.210
14:36:07: ZeroConf ServiceDiscovery: IsAdded(1), Master Monitor, IP: 139.18.164.210
14:36:07: ZeroConf ServiceDiscovery: IsAdded(1), Plasma Monitor, IP: 139.18.164.210
```

Figure 2: TIMMS technical supervisor SCADA application displaying the networked TIMMS components (top right), the management information base of the selected component (middle right) and time course of the monitored variable (bottom right).

3. Results

We designed and implemented a technical supervisory control and data acquisition (SCADA) framework for the monitoring of networked medical hardware and software components. Information about the overall system status and controlling access is designed at different abstraction levels for different user groups (clinical/technical) with separate user interfaces. The master and sub-agents are implemented as C++ class library and are fully compliant with SNMP Versions 1 to 3. The first prototype of the SCADA module is able to retrieve diagnostic information from Ethernet network devices, computer hardware and TIMMS software applications. The user interface for the technical supervisor (Figure 2) comprises simple numerical values of performance measurements as well as graphical trend views for time-dependent values (e.g. network load). Methods of auto-configuration facilitate a highly automated monitoring process without the need for manual interaction.

4. Conclusion

The life critical environment within the operation room requires reliable and safe operation of medical device hardware and software, especially when a large number of different technologies are applied. The proposed SCADA framework is based on standard protocols and encounters these requirements by introducing technical means for the acquisition of performance indicators at hardware and software levels. The framework provides information to detect system anomalies such as network bottlenecks, cache and hard disc space exceeds or CPU consuming software processes and announces these using appropriate alarms to the corresponding user groups. The combination of AgentX subagents with ARM enables the assessment of software performance as well as the detection of hanging or crashed applications with the SCADA watchdog functionality. Further developments focus on automatic reasoning of the overall system status as well as appropriate user interface feedback for the clinical users at the surgical cockpit.

5. References

[1] Lemke H, Vannier M, "*The operating room and the need for an IT infrastructure and standards*", International Journal of Computer Assisted Radiology and Surgery, Vol. 1, No. 3, pp. 117 - 121, 2006.

[2] Fock F, ”*Agent++, An Object Oriented Application Programmers Interface for Development of SNMP Agents Using C++ and SNMP++.*”, http://www.agentpp.com, Last visited 03/06/2009.

[3] The Open Group, “*Application Response Measurement (ARM) Issue 4.0 V2 - C Binding*”, ISBN 1931624380, 2004; available at http://www.opengroup.org/bookstore/catalog/c041.htm, Last visited 03/06/2008.

[4] E. Guttman, “*Zero Configuration Networking*” Proc. INET 2000, Internet Society, Reston, VA; available at http://www.isoc.org/inet2000/cdproceedings/3c/3c_3.htm, Last visited 03/06/2009.