SMART ESPA-X

FUNCTIONAL SPECIFICATIONS

SMART LINK

EDIZIONE 0	0	1	REVISIONI 2	3	4	5
Data	04/4/14	22/05/18				
Redattore	R&D	R&D				
Firma redattore						
Verifica RGQ						
Approvazione DIG						

1 Summary 3
2 Overall Architecture 3
2.1 Smart ESPA-X implementation 3
2.2 Smart ESPA-X internals 4
2.2.1 Service startup 4
2.2.2 ESPA 4.4.4 protocol data flow 4
2.2.3 ESPA-X protocol data flow 4
2.2.4 Protocol initiation 4
2.2.5 Message coordination between endpoints 5
3 ESPA-X Messages 5
3.1 Echo messages 5
3.2 Alarm Messages 5
3.3 Status Messages 6

1 Summary

This document contains technical specifications of a software module, named “Smart ESPA-X”, implementing an ESPA-X client. This software module has the purpose of sending alarms, originated by Mindray patient monitors, to a Tetronik DAKS server, implementing the ESPA-X protocol.

2 Overall Architecture

The following picture describes the overall architecture of the solution.

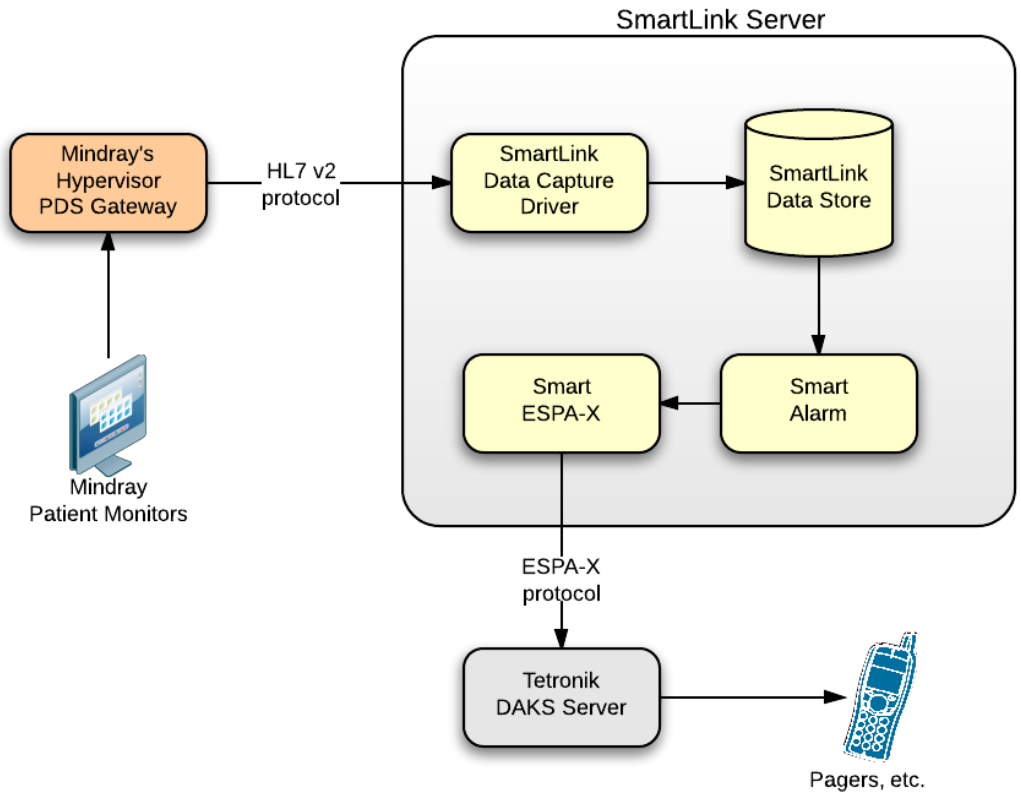

Figure 1 - Overall system architecture

This document is focused on the component “Smart ESPA-X”, because any other component in the above figure is already in place; so the development target is actually a software module, able to translate requests to pagers carried out by ESPA 4.4.4 protocol to corresponding requests carried out by ESPA-X protocol, and vice-versa.

2.1 Smart ESPA-X implementation

The Smart ESPA-X module will be implemented as a NT service, implementing two distinct endpoints towards an ESPA 4.4.4 interface and an ESPA-X interface. Both endpoints are based on TCP/IP communication, that is mandatory for ESPA-X, an option for ESPA 4.4.4; in fact, ESPA 4.4.4 was designed to work on asynchronous serial interfaces, but can be implemented on top of TCP/IP as well. The PABX component inside the SmartLink server already works in TCP/IP; however, use of a serial interface instead

of TCP/IP is straightforward, this functionality can be obtained by means of a MOXA – or similar – protocol converter.

2.2 Smart ESPA-X internals

The following picture described the proposed internal architecture of the ESPA-X server.

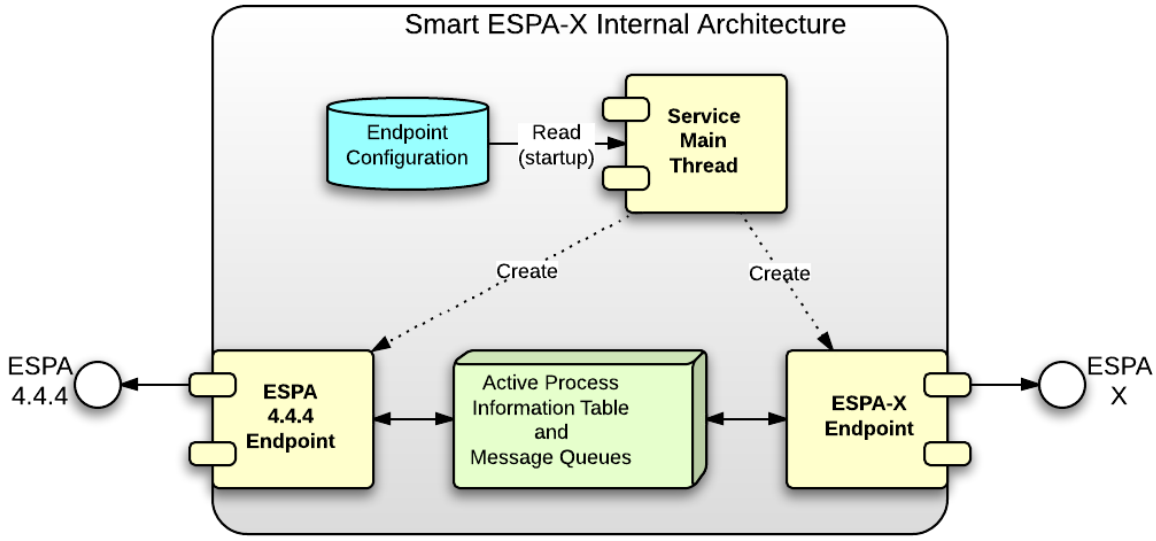

Figure 2 - Smart ESPA-X internal architecture

2.2.1 Service startup

The Smart ESPA-X service, at startup, reads from a configuration file the endpoint configuration. In order to be able to work, the system must have a couple of endpoints: one devoted to the ESPA 4.4.4 interface (left side of Figure 2), one devoted to the ESPA-X interface (right side of Figure 2).

Both endpoints are implemented as TCP/IP channels.

2.2.2 ESPA 4.4.4 protocol data flow

ESPA 4.4.4 is based on polling-selecting, a multi-drop, master-slave protocol, where the PABX component (see Figure 1) acts as a master, while the Smart ESPA-X acts as a slave.

ESPA 4.4.4 is message based, messages are ASCII-based, plain messages.

2.2.3 ESPA-X protocol data flow

The ESPA-X protocol has no predefined master, both parties must be able to send and/or receive messages at all times, bot solicited (i.e., in reply to messages sent before) and unsolicited (notification messages).

ESPA-X is message based too, but messages are structured in XML and allow slots to fit proprietary extensions.

2.2.4 Protocol initiation

At system startup, when both endpoints are instantiated, the ESPA 4.4.4 endpoint system will try to set up a TCP/IP connection with the PABX gateway. Because the PABX Gateway is responsible for actively create a connection, the ESPA 4.4.4 endpoint will actually wait for an incoming TCP/IP connection, and the ESPA-X endpoint will not be opened until completion of the ESPA 4.4.4 channel activation.

As soon as a TCP/IP channel is opened on the ESPA 4.4.4 side, the ESPA-X endpoint will try to connect to the ESPA-X server, login by supplying credentials (also stored in endpoint configuration, see Figure 1).

2.2.5 Message coordination between endpoints

Due to different behavior between the two protocols, both endpoints must adhere to strict coordination of message exchange. As an example, consider when a notification message comes from the ESPA-X server, regarding a status change of a pager call. The ESPA-X server will forward such notification to the ESPA-X endpoint in an unsolicited manner, but this message cannot be delivered to the ESPA 4.4.4 side because the ESPA 4.4.4 endpoint must wait for next polling before transmission can take place.

For this reason, the Smart ESPA-X must be able to keep track of active processes (i.e. pager calls, system status notifications, end so on) in a dedicated area. Entries in this area, named “Active Process Information Table and Message Queues” (from now on: “APT” acronym for “active process table”) in Figure 2, will hold relevant information coming from one side, until delivery to the other side. After that, entries will be deleted.

Note that the two endpoint will be implemented as separate threads, therefore access to the process table must be synchronized and thread-safe.

3 ESPA-X Messages

Messages flowing from the SmartAlarm component to the DAKS server can be classified as follows:

- Echo messages
- Alarm messages
- Status messages

3.1 Echo messages

Echo messages have the purpose of ensure that both parties are still “alive” and able to exchange information. Both ESPA 4.4.4 protocol and ESPA-X protocol have a sort of echo message (in ESPA-X the official name is “heartbeat” message). In order to avoid dependency between the two sides, both endpoints exchange echo messages independently from each other.

In case one side will not be able to receive echo messages back, the APT will be updated accordingly so that the other party will be informed by means of status information.

3.2 Alarm Messages

Alarm messages are initiated at PABX site and propagated to the DAKS server. The following figure shows a typical data flow for an alarm message.

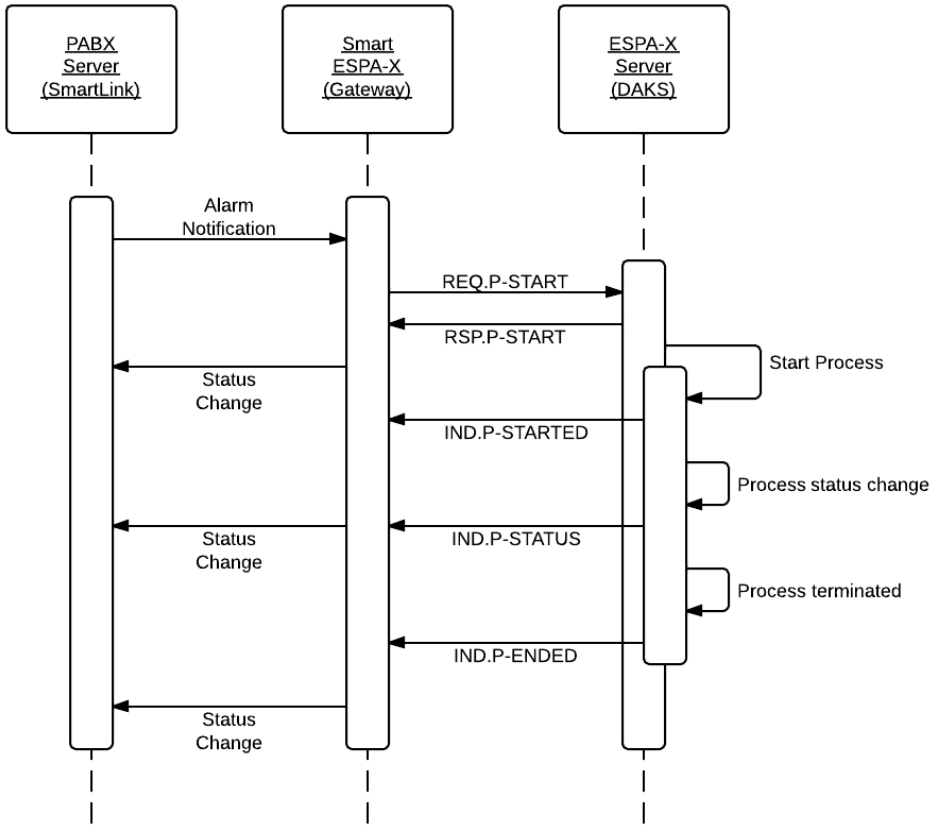

Figure 3 - Message flow during alarm notification

3.3 Status Messages

Status messages are originated at the DAKS server, and propagated to the SmartAlarm component.

Status messages are of two kinds:

- Status information regarding the alarm call
- Status information regarding the DAKS server itself and/or the Smart ESPA-X service