SDLí92 at Siemens ÷ N IN

The ServiceNode Project

Overview

- Introduction
- The provided Service
- Embedding in the Network
- Software architecture and technology
- Advantages and Disadvantages
- Future
Introduction
Basic IN Architecture
The One Number Service

Requirements

- transparent embedding of a mobile (PHS) network into the existing fixed network
- two personal and online recordable announcements per subscriber
- multi language user interaction dialog to guide announcement recording
- 500,000 subscriber
- 180 calls per second
- 6,000 parallel calls (only call establishment phase)
Network Model

- PHS network providing a mobility feature for PHS hand-helds
- fixed network providing access to domestic fixed line terminals
- service provisioning part providing the One Number Service which integrates the PHS hand-helds into the fixed network numbering scheme

Note: MAP' is a subset of MAP operations comprising the SendRoutingInformation operation only.
Physical Entities Of SN

- SNC is a high available system with doubled hardware
- Local control of SN is provided via a separate workstation
- SDS with doubled CPU and disk
- SDS provides local management capabilities
- Administration of service and subscriber data via SQL interface at SDS
SNC Architecture

Service Logic Program

Resource Control

Switch Control

Resource Manager

Audit Manager

Safe Guard

Resource Platform Control

Oracle Database

Data Provisioning

SCPDB - service data
- subscriber data

MAP

INAP

TCAP

ISUP

SS7

Switch Control Interface

Resource Platform Control Interface

SNMP-Alarming

SNMP TRAP Generator

EventHandler

SNMP Agent

MIB Agent

Management Information Base
(for internal objects)

Management Information Base
(for external objects)

FTP

alarmed events from all components

SNMP-MIB

SNMP TRAP

Generator

SNMP Agent

FTP

Oracle Database

Data Provisioning

SCPDB - service data
- subscriber data

SNC Architecture
Tools

- Standard SDL’92 (Z.100) + ASN.1 (X.208, not Z.105 conform)
- GEODE SDL editor
- C++ Code-Generation (SITE tools from Humboldt University)
- common, flexible, configurable runtime environment (Common Application Library-CAL)
- automatical creation of test tools for applications (TCL based)
```
system ResourceControl;
block RC_Block;
signal IDP;
process Resource;
start;
```

```
RC_Interf DEFINITIONS ::= BEGIN
IDP_Arg ::= SEQUENCE {
    ...
    ...
} END
```
# Software Architecture

<table>
<thead>
<tr>
<th>SDL Application (SDL System)</th>
<th>SDL Runtime Library</th>
<th>SDL Adapter</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CAL</td>
<td>Service Functions</td>
</tr>
<tr>
<td>SDL Runtime Library</td>
<td>Configuration MIB Interface</td>
<td>Trace Debug</td>
</tr>
<tr>
<td>SDL Adapter</td>
<td>SafeGuard Interface</td>
<td></td>
</tr>
<tr>
<td>MIB</td>
<td>MIB</td>
<td>Connection Handling Communication</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Process Control</td>
</tr>
<tr>
<td></td>
<td>Sockets</td>
<td>other APIs</td>
</tr>
<tr>
<td>Operating System</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Used SDL Features

- SDLí88 as basis
- Inheritance of SDL Processes
- Packages
- Newtype ... operators ...
- Mix of ASN.1 and SDL data types
- Communication to ENV
- Mapping SDL System to UNIX process
Advantages

- Complete technology from design to implementation
- Abstract level of programming
- Safe programming environment
- Clear definition of interfaces
- State event model in language
- Common run time environment
- Common management interface
- Off-line / On-line test environment
- High availability features
Disadvantages

- Big generated code
- Long compilation time
- No graphical tool, that full supports the used SDLí92-ASN.1 combination
- No real SDL debugger
- No object oriented data model
Experiences

- low learning effort necessary
- tool chain must completely work and must be fully tested
- direct developer support is very important
- high performance, e.g. on HP-H70
  2 SDL systems, 330 calls/s -> 50% CPU
- high stability
- easily maintainable and extendible
Future

- Porting to Win NT
- Use of SDL Exceptions
- X.680-682 ASN.1
- Implementing Database Access from SDL
- SDL + CORBA
- Persistent SDL Processes
- SDL Debugger
- dynamic loadable SDL units (Packages, Processes)
- Version handling of exchangeable SDL units