



USSD-*SGateway*

Technical Overview v1

Table of Contents

| | |
|--------------------------|----|
| Introduction..... | 3 |
| Capacity..... | 3 |
| Architecture | 4 |
| Hardware | 5 |
| Interactivity View | 6 |
| SNMP Traps | 7 |
| Configuration | 8 |
| 1. MTP2..... | 9 |
| 2. MTP3..... | 10 |
| 3. SCCP | 12 |
| 4. TCAP | 13 |
| Logs Files..... | 14 |
| Scalability Plan | 14 |

Introduction

The Unstructured Supplementary Service Data (USSD) is a UMTS/GSM Service that allows interactive communication between subscribers and application across a UMTS/GSM network.

It is primary used on real-time and instant messaging type phone services because there is no store-and-forward of messages typical of the short message services (SMS).

The response time of a USSD request is measured between 100 and 200ms compared to the 5-10 seconds of SMS.

The functionalities of USSD make it ideal for information queries like available balance, content downloading and any information services.

Capacity

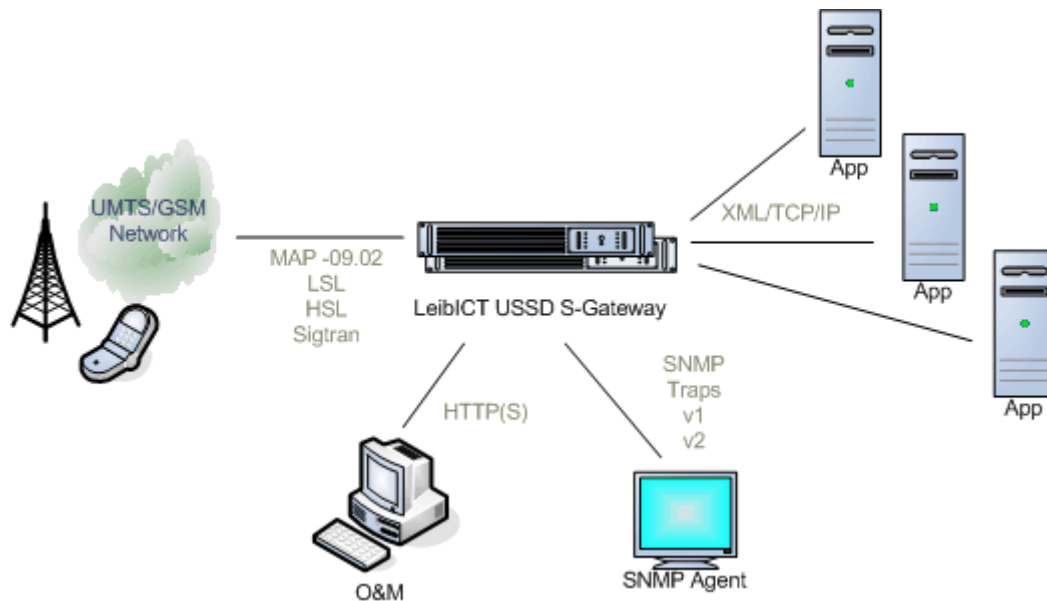
The current USSD-Gateway version supports 16384 simultaneous incoming and outgoing transactions being processed at the same time.

On most deployments the USSD-Gateway is limited by the SS7 Network resources or the Application/Database service load.

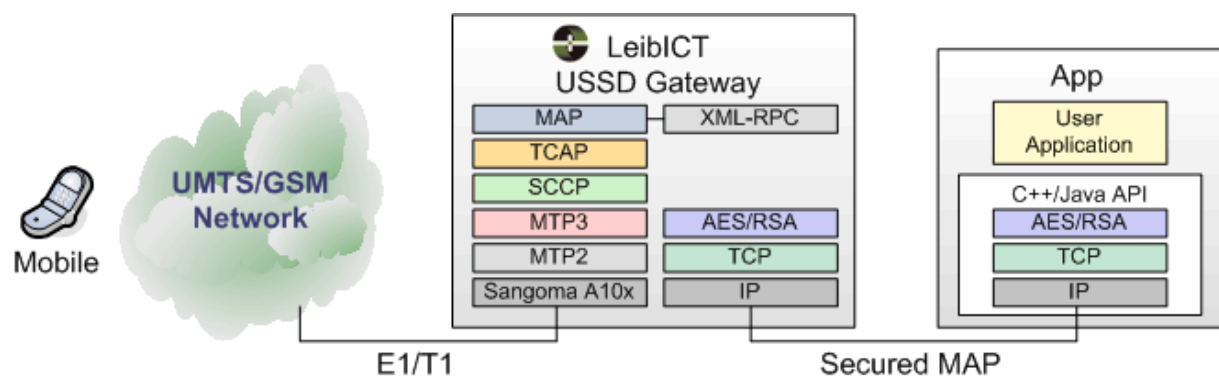
The modular architecture allows increasing system capacity as needed by adding more hardware.

Architecture

The next Implementation View shows the components of the USSD Gateway based on a single server.

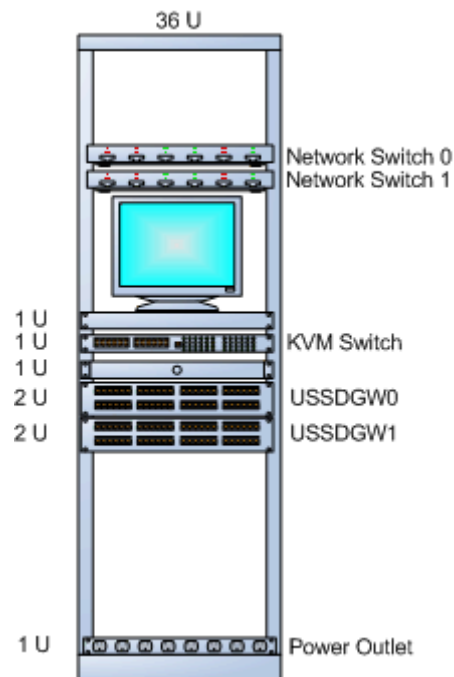


All the software components are Unix daemons that are started at boot time:



Hardware

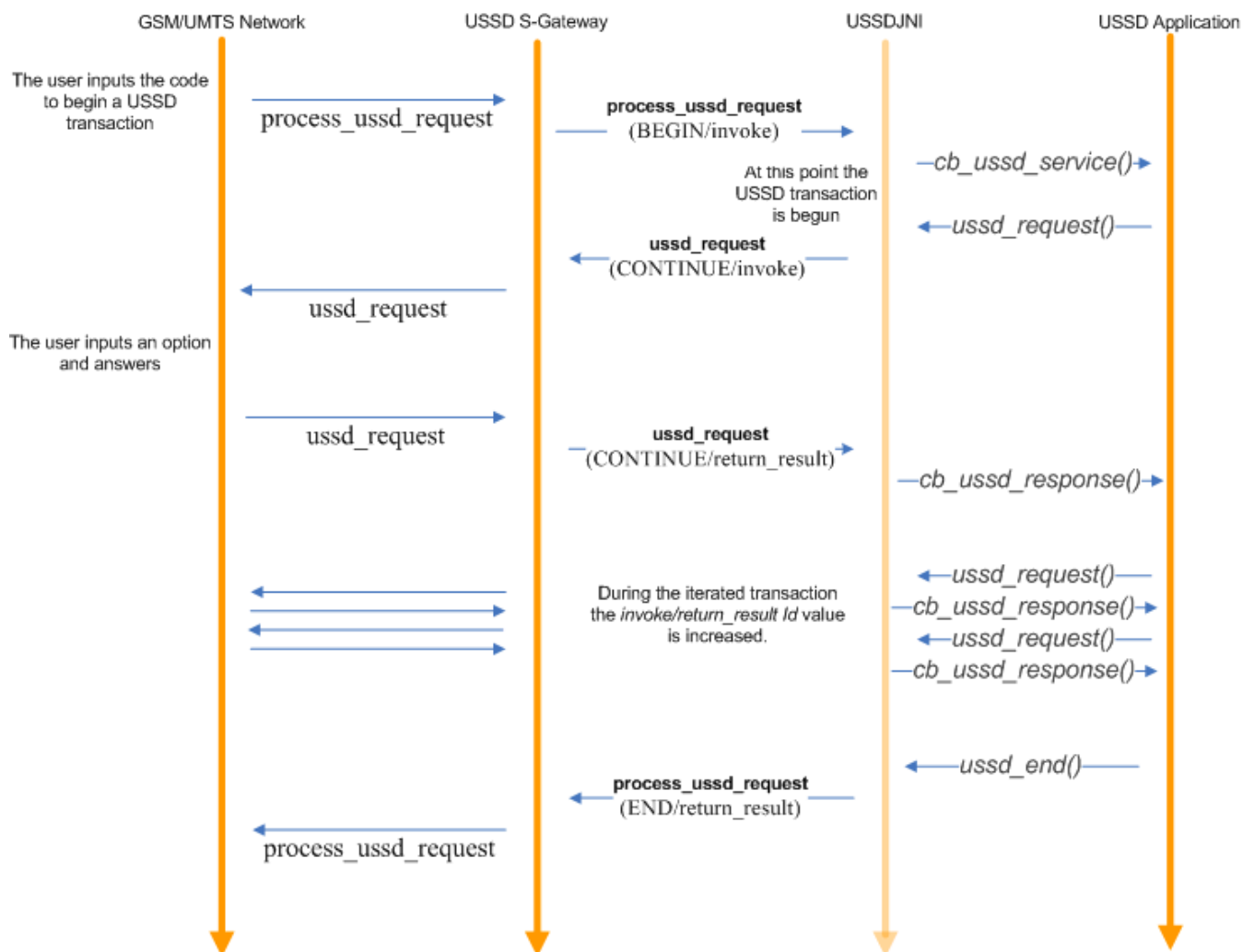
The USSD Gateway is based on standard x86 Servers from HP or Sun. One or more Sangoma A10x boards provides the E1/T1 TDM Interface. The Operating System can be either Linux (RedHat or CentOS). The next picture shows a dual module configuration:



Typical HP/RedHat Solution is based on DL 380 with Intel Core 2 Quad and 4Gb of RAM. Hard disk space depends on customer needs of traffic logging storage. Up to 8Kb bytes are needed per transaction.

Interactivity View

The next scenario shows a ussd transaction ended by the application:



SNMP Traps

The AlarmServer module is responsible of sending traps to the configured destinations. This is the trap list and meanings:

| Trap Name | Meaning |
|--------------------|--|
| notification | Notification o a particular event like user Loggings |
| congestionDetected | Congestion Detected (of the links) |
| congestionAbated | Congestion Abated |
| serviceUp | System is ready to serve |
| serviceDown | System is down because of failure |
| associationUp | Signalling link Up |
| associationDown | Signalling link Down |

All the traps have the same structure with an object name, an instance identification and a status.

Configuration

LeibICT USSD Gateway has its configuration files deployed in a configurable directory. The default config directory is:

/leibict/ussdgateway/cfg

The main configuration of the SS7 Stacks is implemented within a XML file:

```
<siggw name="LeibICT">
  <mtp2 id="0" addr="localhost" port="5400" name="mtp2">
    <card id="0">
      <trunk id="0">
        <channel timeSlot="1" linkId="0"/>
        <channel timeSlot="2" linkId="1"/>
      </trunk>
    </card>
  </mtp2>
  <mtp3 id="0" addr="localhost" port="5401" name="mtp3">
    <user id="0" opc="146" name="stack">
      <linkset id="0" apc="93" name="linkset">
        <link id="0" slc="0" name="link0"/>
        <link id="1" slc="1" name="link1"/>
      </linkset>
      <destination id="0" dpc="93" name="destination0">
        <route id="0" linksetId="0" name="route0"/>
      </destination>
    </user>
    <mtp2_module id="0"/>
  </mtp3>
  <sccp id="0" addr="localhost" port="5402" name="sccp">
    <user id="0"
      addressIndicator="18"
      pointCode="0"
      subsystemNumber="146"
      natureOfAddress="4"
      translationType="0"
      numberingPlan="1"
      encodingScheme="1"
      globalTitle="5001205280001"/>
    <translator id="0"
      prefix=""
      loadsharing="0">
      <mtp3_destination id="0"/>
    </translator>
    <mtp3_module id="0"/>
  </sccp>
  <tcap id="0" addr="localhost" port="5403" name="tcap">
    <sccp_module id="0"/>
  </tcap>
</siggw>
```

The XML configuration file consists in four parts, each referring a single stack: MTP2, MTP3, SCCP and TCAP.

1. MTP2

The MTP2 layer is a wrapper of the Sangoma mtpBoost TM MTP2 Stack, thus it only contains the operative configuration.

The MTP2 Layer has the next parameters:

| Field | Description | Example |
|-------------|--|-------------|
| id | Identification Number of the Stack | "0" |
| addr | DN or IP Address of the machine where the Stack runs | "localhost" |
| port | TCP Port where the Stack listen | "5400" |
| name | Name of the Stack | "mtp2_0" |
| card | E1 Card (s) | |

The Card field has the next parameters:

| Field | Description | Example |
|--------------|-----------------------------------|---------|
| id | Identification Number of the Card | "0" |
| trunk | Trunk (s) | * |

The Trunk field has the next parameters:

| Field | Description | Example |
|----------------|------------------------------------|---------|
| id | Identification Number of the Trunk | "0" |
| channel | Channels (s) | * |

The Channel field has the next parameters:

| Field | Description | Example |
|----------|--------------------------------------|---------|
| id | Identification Number of the Channel | "0" |
| timeSlot | Time Slot of the Signalling Channel | "1" |
| linkId | MTP3 Link ID | "1" |

2. MTP3

The MTP3 Stack implements the MTP3 Protocol and State Machines:

The MTP3 Layer has the next parameters:

| Field | Description | Example |
|-------------|--|-------------|
| id | Identification Number of the Stack | "0" |
| addr | DN or IP Address of the machine where the Stack runs | "localhost" |
| port | TCP Port where the Stack listen | "5401" |
| name | Name of the Stack | "mtp3_0" |
| user | MTP3 User (s) | * |

The MTP3 User field has the next parameters:

| Field | Description | Example |
|--------------------|-----------------------------------|---------|
| id | Identification Number of the User | "0" |
| opc | Originating Point Code | "321" |
| name | Name of the User | "user0" |
| destination | MTP3 Destination (s) | * |
| linkset | MTP3 LinkSet (s) | * |

The MTP3 Destination field has the next parameters:

| Field | Description | Example |
|--------------|--|---------|
| id | Identification Number of the Destination | "0" |
| dpc | Destination Point Code | "123" |
| name | Name of the User | "dest0" |
| route | MTP3 Route (s) | * |

The MTP3 Route field has the next parameters:

| Field | Description | Example |
|-----------|--------------------------------------|----------|
| id | Identification Number of the Route | "0" |
| linksetId | Identification Number of the LinkSet | "0" |
| name | Name of the Route | "route0" |

The MTP3 LinkSet field has the next parameters:

| Field | Description | Example |
|-------------|--------------------------------------|------------|
| id | Identification Number of the LinkSet | "0" |
| apc | Adjacent Point Code | "123" |
| name | Name of the User | "linkset0" |
| link | MTP3 Link (s) | * |

The MTP3 Link field has the next parameters:

| Field | Description | Example |
|-------|-----------------------------------|---------|
| id | Identification Number of the Link | "0" |
| slc | Signalling Link Code | "0" |
| name | Name of the Route | "link0" |

3. SCCP

The SCCP Stack implements the SCCP Protocol and State Machines:

The SCCP Layer has the next parameters:

| Field | Description | Example |
|-------------------|--|-------------|
| id | Identification Number of the Stack | "0" |
| addr | DN or IP Address of the machine where the Stack runs | "localhost" |
| port | TCP Port where the Stack listen | "5402" |
| name | Name of the Stack | "sccp0" |
| user | SCCP User (s) | * |
| translator | SCCP Translator (s) | * |

The SCCP User field has the next parameters:

| Field | Description | Example |
|-----------------|------------------------------------|-------------|
| id | Identification Number of the User | "0" |
| adresIndicator | ITUT Q.713 SCCP Address parameters | "18" |
| pointCode | | "0" |
| subsystemNumber | | "146" |
| NatureOfAddress | | "4" |
| translationType | | "0" |
| numberingPlan | | "1" |
| encodingScheme | | "1" |
| globalTitle | | "205280001" |

The SCCP Translator field has the next parameters:

| Field | Description | Example |
|-------------------------|---|---------|
| id | Identification Number of the Translator | "0" |
| prefix | Prefix of the Global Title to be translated | "205" |
| loadsharing | Enable load sharing between entities | "0" |
| mtp3_destination | MTP3 Destination (s) | * |

The MTP3 Destination field has the next parameters:

| Field | Description | Example |
|-------|---|---------|
| id | Identification Number of the MTP3 Destination | "0" |

4. TCAP

The TCAP Stack implements the TCAP Protocol and State Machines:

The TCAP Layer has the next parameters:

| Field | Description | Example |
|-------|--|-------------|
| id | Identification Number of the Stack | "0" |
| addr | DN or IP Address of the machine where the Stack runs | "localhost" |
| port | TCP Port where the Stack listen | "5403" |
| name | Name of the Stack | "tcap0" |

Logs Files

All the process modules have a standard, debug and error logs. All the logs are generated in a configurable directory. The default log directory is:

/leibict/ussdgateway/logs

Debug logs are disabled by default, enabling them may cause a performance decrease.

Scalability Plan

The scalability of the solution can be performed by increasing the hardware modules in multiple ways: sharing the same SS7 network addresses (SCCP and MTP) or not and sharing the same TCP/IP network address or not.

By sharing the same SS7 network addresses, up to four nodes can be paired.

Beyond four nodes, more SS7 network addresses must be defined.

By sharing the same TCP/IP network address, a load balancer must be deployed like the one provided in the RedHat Cluster Suite.