

REST API

←

OmniSS7 REST API Swagger UI

-
- HTTP
- Swagger UI
- API
-
-
-
- (Prometheus)
-

OmniSS7 REST API MAP API

- MAP SRI-SRI-for-SM UpdateLocation
- MAP
- Prometheus

API



HTTP 配置

配置

属性	值	说明
协议	HTTP	协议
IP 地址	0.0.0.0	监听地址
端口	8080	监听端口
插件	Plug.Cowboy	插件

URL: `http://[server-ip]:8080`

配置 HTTP 服务

配置 HTTP 服务

```
config :omniss7,  
  start_http_server: true # 默认 false
```

配置: `true`

配置: HTTP 服务 REST API/Swagger UI

Swagger UI

API 文档 **Swagger UI** 是 API 文档

配置 Swagger UI

URL: `http://[server-ip]:8080/swagger`

□□:

- □□□ API □□
- □□□□□□□□□□
- □□/□□□□
- □□□□□□

Swagger JSON

OpenAPI □□□□□□□□□□

URL: `http://[server-ip]:8080/swagger.json`

□□:

- □□□ Postman □□□ API □□□
- □□□□□□
- API □□□□□□

API □□

□□ MAP □□□□□□□□□□ `POST /api/{operation}`

□□□□

□□	□□	□□	□□
<code>/api/sri</code>	POST	□□□□□□	10s
<code>/api/sri-for-sm</code>	POST	□□ SM □□□□□	10s
<code>/api/send-auth-info</code>	POST	□□□□□□	10s
<code>/api/MT-forwardSM</code>	POST	□□□□□□ SM	10s
<code>/api/forwardSM</code>	POST	□□ SM	10s
<code>/api/updateLocation</code>	POST	□□□□	10s
<code>/api/prn</code>	POST	□□□□□□	10s
<code>/api/uszd/send</code>	POST	□□□□□ USSD □□	10s
<code>/metrics</code>	GET	Prometheus □□	N/A
<code>/swagger</code>	GET	Swagger UI	N/A
<code>/swagger.json</code>	GET	OpenAPI □□	N/A

□□: □□ MAP □□□□□□ □□□□ **10** □□□□

SendRoutingInfo (SRI)

□□□□□□□□□□□□□□□□□□□□

□□: POST `/api/sri`

□□□:

```
{
  "msisdn": "1234567890",
  "gmsc": "5551234567"
}
```

□□:

□□	□□	□□	□□
msisdn	□□□	□	□□□ MSISDN
gmsc	□□□	□	□□ MSC □□□□

□□ (200 OK):

```
{
  "result": {
    "imsi": "001001234567890",
    "msrn": "5551234999",
    "vlr_number": "5551234800",
    ...
  }
}
```

□□ (504 □□□□):

```
{
  "error": "timeout"
}
```

cURL □□:

```
curl -X POST http://localhost:8080/api/sri \
  -H "Content-Type: application/json" \
  -d '{
    "msisdn": "1234567890",
    "gmsc": "5551234567"
  }'
```

SendRoutingInfoForSM (SRI-for-SM)

SendRoutingInfoForSM SMS routing information

Request: `POST /api/sri-for-sm`

Request:

```
{
  "msisdn": "1234567890",
  "service_center": "5551234567"
}
```

Response:

Field	Type	Required	Description
<code>msisdn</code>	String	Yes	MSISDN
<code>service_center</code>	String	Yes	Service Center

Response (200 OK):

```
{
  "result": {
    "imsi": "001001234567890",
    "msc_number": "5551234800",
    "location_info": {...},
    ...
  }
}
```

cURL ☐☐:

```
curl -X POST http://localhost:8080/api/sri-for-sm \
-H "Content-Type: application/json" \
-d '{
  "msisdn": "1234567890",
  "service_center": "5551234567"
}'
```

SendAuthenticationInfo

☐☐☐☐☐☐☐☐☐☐

☐☐: POST /api/send-auth-info

☐☐☐:

```
{
  "imsi": "001001234567890",
  "vectors": 3
}
```

☐☐:

Field	Type	Value	Description
imsi	String	001001234567890	IMSI
vectors	Integer	3	Number of authentication vectors

Response (200 OK):

```
{
  "result": {
    "authentication_sets": [
      {
        "rand": "0123456789ABCDEF...",
        "xres": "...",
        "ck": "...",
        "ik": "...",
        "autn": "..."
      }
    ],
    ...
  }
}
```

cURL Request:

```
curl -X POST http://localhost:8080/api/send-auth-info \
-H "Content-Type: application/json" \
-d '{
  "imsi": "001001234567890",
  "vectors": 3
}'
```

MT-ForwardSM

Request: POST /api/MT-forwardSM

Request: POST /api/MT-forwardSM

Request:

```
{
  "imsi": "001001234567890",
  "destination_service_centre": "5551234567",
  "originating_service_center": "5551234568",
  "smsPDU": "0001000A8121436587F900001C48656C6C6F20576F726C64"
}
```

Response:

Field	Type	Length	Description
imsi	String	15	International Mobile Subscriber Identity (IMSI)
destination_service_centre	String	10	Destination Service Centre (GT)
originating_service_center	String	10	Originating Service Centre (GT)
smsPDU	String	255	Short Message Service (SMS) TPDU

Response: smsPDU [0001000A8121436587F900001C48656C6C6F20576F726C64]

Response (200 OK):

```
{
  "result": {
    "delivery_status": "success",
    ...
  }
}
```

cURL Request:

```
curl -X POST http://localhost:8080/api/MT-forwardSM \  
-H "Content-Type: application/json" \  
-d '{  
  "imsi": "001001234567890",  
  "destination_service_centre": "5551234567",  
  "originating_service_center": "5551234568",  
  "smsPDU": "0001000A8121436587F900001C48656C6C6F20576F726C64"  
}'
```

ForwardSM

☐☐ SMS ☐☐☐☐☐☐☐ MO-SMS☐☐

☐☐: POST /api/forwardSM

☐☐☐: ☐ MT-ForwardSM ☐☐

cURL ☐☐:

```
curl -X POST http://localhost:8080/api/forwardSM \  
-H "Content-Type: application/json" \  
-d '{  
  "imsi": "001001234567890",  
  "destination_service_centre": "5551234567",  
  "originating_service_center": "5551234568",  
  "smsPDU": "0001000A8121436587F900001C48656C6C6F20576F726C64"  
}'
```

UpdateLocation

☐☐ HLR ☐☐☐☐☐☐☐ VLR ☐☐☐☐

☐☐: POST /api/updateLocation

☐☐☐:

```
{
  "imsi": "001001234567890",
  "vlr": "5551234800"
}
```

□□:

□□	□□	□□	□□
imsi	□□□	□	□□ IMSI
vlr	□□□	□	VLR □□◆◆◆□□□

□□ (200 OK):

```
{
  "result": {
    "hlr_number": "5551234567",
    "subscriber_data": {...},
    ...
  }
}
```

□□: □ HLR □□□□□□□□□□ 10 □□□□ InsertSubscriberData (ISD) □□□

cURL □□:

```
curl -X POST http://localhost:8080/api/updateLocation \
-H "Content-Type: application/json" \
-d '{
  "imsi": "001001234567890",
  "vlr": "5551234800"
}'
```

ProvideRoamingNumber (PRN)

MSRN

Method: POST /api/prn

Request:

```
{  
  "msisdn": "1234567890",  
  "gmsc": "5551234567",  
  "msc_number": "5551234800",  
  "imsi": "001001234567890"  
}
```

Response:

Field	Type	Length	Description
msisdn	String	10	MSISDN
gmsc	String	10	MSC GT
msc_number	String	10	MSC ID
imsi	String	15	IMSI

Response (200 OK):

```
{  
  "result": {  
    "msrn": "5551234999",  
    ...  
  }  
}
```

cURL:

```
curl -X POST http://localhost:8080/api/prn \
-H "Content-Type: application/json" \
-d '{
  "msisdn": "1234567890",
  "gmsc": "5551234567",
  "msc_number": "5551234800",
  "imsi": "001001234567890"
}'
```

USSD

USSD `ussd_gateway_enabled: true` USSD

POST `/api/ussd/send`

:

```
{
  "msisdn": "+254712345678",
  "text": "1",
  "callback_url": "http://billing-app:9000/ussd"
}
```

:

Field	Type	Required	Description
<code>msisdn</code>	String	Yes	MSISDN
<code>text</code>	String	Yes	USSD GSM 7
<code>callback_url</code>	String	Yes	URL HTTP POST

(200 OK):

□□□□

□□□□□ **JSON** □□□

□□□□

HTTP □□: 200 OK

□□:

```
{  
  "result": {  
    // □□□□□□□□  
  }  
}
```

□□□□

HTTP □□:

- 400 □□□□ - □□□□□ JSON
- 504 □□□□ - MAP □□□□□10 □□
- 404 □□□ - □□□□

□□:

```
{  
  "error": "timeout"  
}
```

□

```
{  
  "error": "invalid request"  
}
```

□□□□

□□□□

□□	HTTP □□	□□	□□□□
□□ JSON	400	□□□□□□□□ JSON	□□ JSON □□
□□□□	400	□□□□□□	□□□□□□□□
□□	504	MAP □□□□ 10 □□□	□□ M3UA □□□□HLR/VLR □□□
□□□	404	□□□□	□□□□ URL

□□□□

□□ MAP □□□□□□ □□□□ **10** □□□□

1. □□□□□ MapClient GenServer
2. □□□□□□ 10 □
3. □□□□□□ → □□ 504 □□□□
4. □□□□□□ → □□ 200 OK □□□

□□□□□□:

- □□ M3UA □□□□□□ Web UI → M3UA □□□
- □□□□□□□□ HLR/VLR/MSC □□□□□□
- □□□□□□
- □□ SS7 □□□□□□□□□□

□□ (Prometheus)

□ API □□ Prometheus □□□□□□□□

□□□□

URL: `http://[server-ip]:8080/metrics`

□□: Prometheus □□□□

□□□□:

```
# HELP map_requests_total □ MAP □□□
# TYPE map_requests_total counter
map_requests_total{operation="sri"} 42
map_requests_total{operation="sri_for_sm"} 158
map_requests_total{operation="updateLocation"} 23

# HELP cap_requests_total □ CAP □□□
# TYPE cap_requests_total counter
cap_requests_total{operation="initialDP"} 87
cap_requests_total{operation="requestReportBCSMEEvent"} 91

# HELP map_request_duration_milliseconds MAP □□/□□□□□□□□□□□□
# TYPE map_request_duration_milliseconds histogram
map_request_duration_milliseconds_bucket{operation="sri",le="10"}
5
map_request_duration_milliseconds_bucket{operation="sri",le="50"}
12
map_request_duration_milliseconds_bucket{operation="sri",le="100"}
35
...

# HELP map_pending_requests □□□□ MAP TID □□□□□
# TYPE map_pending_requests gauge
map_pending_requests 3

# HELP omniss7_license_status □□□□□□□□1 = □□□□0 = □□□
# TYPE omniss7_license_status gauge
omniss7_license_status 1
```

メトリクス

メトリクス名	単位	方向	説明
<code>map_requests_total</code>	操作回数	operation	MAP 操作の総回数
<code>cap_requests_total</code>	操作回数	operation	CAP 操作の総回数
<code>map_request_duration_milliseconds</code>	ミリ秒	operation	MAP 操作の平均実行時間
<code>map_pending_requests</code>	-	-	現在 MAP 操作中の回数
<code>ussd_requests_total</code>	操作回数	direction	USSD 操作の総回数 (方向別)
<code>ussd_active_sessions</code>	-	-	現在 USSD 操作中の回数
<code>omniss7_license_status</code>	-	-	ライセンスステータス (1 = 有効, 0 = 無効)

Prometheus 設定

設定ファイル `prometheus.yml`

```
scrape_configs:  
  - job_name: 'omniss7'  
    static_configs:  
      - targets: ['server-ip:8080']  
    metrics_path: '/metrics'  
    scrape_interval: 15s
```

□□□□

Python □□

```
import requests
import json

# SRI-for-SM □□
url = "http://localhost:8080/api/sri-for-sm"
payload = {
    "msisdn": "1234567890",
    "service_center": "5551234567"
}

response = requests.post(url, json=payload, timeout=15)

if response.status_code == 200:
    result = response.json()
    print(f"□□: {result}")
elif response.status_code == 504:
    print("□□ - □□□□")
else:
    print(f"□□: {response.status_code} - {response.text}")
```

JavaScript ☐☐

```
const axios = require('axios');

async function sendSRI() {
  try {
    const response = await
axios.post('http://localhost:8080/api/sri', {
  msisdn: '1234567890',
  gmsc: '5551234567'
}, {
  timeout: 15000
});

    console.log('☐☐:', response.data);
  } catch (error) {
    if (error.code === 'ECONNABORTED') {
      console.error('☐☐ - ☐☐☐☐');
    } else {
      console.error('☐☐:', error.response?.data || error.message);
    }
  }
}

sendSRI();
```

Bash/cURL ☐☐

```
#!/bin/bash

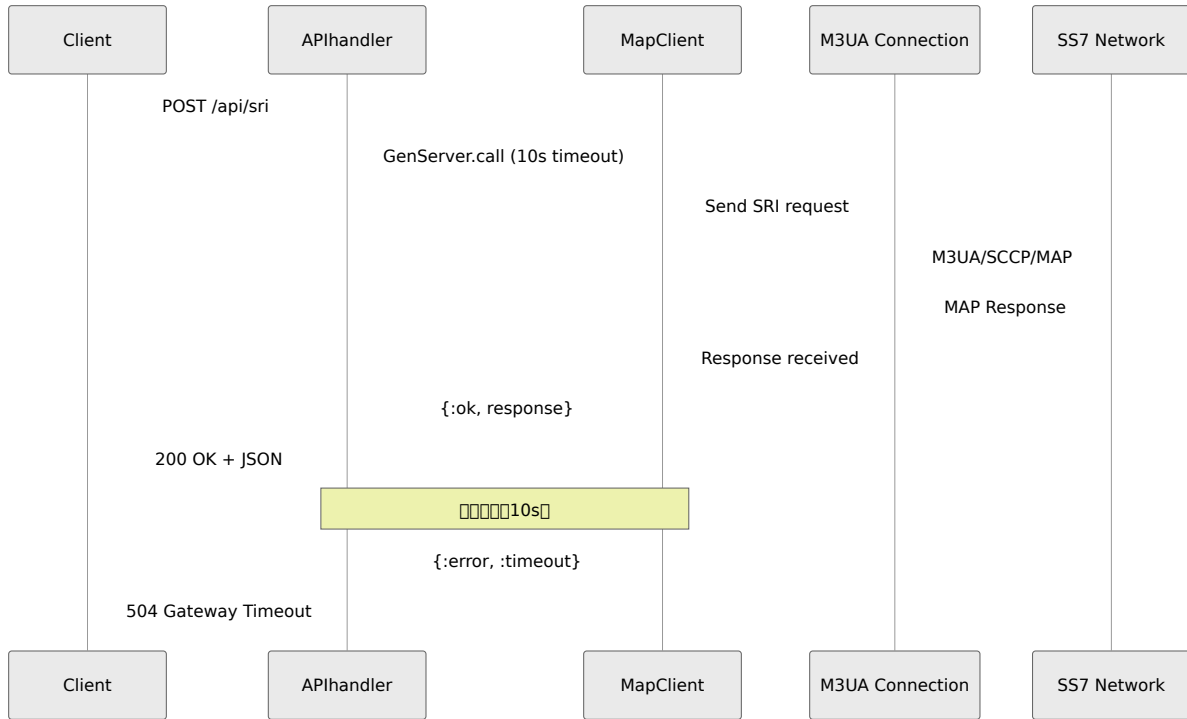
# UpdateLocation ☐☐
response=$(curl -s -w "\n%{http_code}" -X POST
http://localhost:8080/api/updateLocation \
-H "Content-Type: application/json" \
-d '{
  "imsi": "001001234567890",
  "vlr": "5551234800"
}')

http_code=$(echo "$response" | tail -n 1)
body=$(echo "$response" | sed '$d')

if [ "$http_code" -eq 200 ]; then
  echo "☐☐: $body"
elif [ "$http_code" -eq 504 ]; then
  echo "☐☐ - ☐☐☐☐☐☐"
else
  echo "☐☐ $http_code: $body"
fi
```



API



OmniSS7 REST API

MAP - SRI SRI-for-SM UpdateLocation SMS

Swagger UI - API

Prometheus -

- MAP 10

HTTP - 8080 `start_http_server`

Web UI **Web UI**

□□□□□□□□

← □□□□

SS7□□□OmniSS7□□□□□□□□

SS7□□□



MAP□□□

□□	□□□	□□
updateLocation	2	□□□□□□
cancelLocation	3	□VLR□□
provideRoamingNumber	4	□□MSRN
sendRoutingInfo	22	□□□□□□
mt-forwardSM	44	□SMS□□□□□□
sendRoutingInfoForSM	45	□□SMS□□
mo-forwardSM	46	□□□□□SMS
sendAuthenticationInfo	56	□□□□□□

TCAP

- **BEGIN** -
 - **CONTINUE** -
 - **END** -
 - **ABORT** -
-

SCCP

- **E.164** - 447712345678
- **E.212** - IMSI 234509876543210
- **E.214** -

SSN

- **SSN 6**: HLR
 - **SSN 7**: VLR
 - **SSN 8**: MSC/SMSC
 - **SSN 9**: GMLC
 - **SSN 10**: SGSN
-

SMS TPDU

- **SMS-DELIVER** (MT) -
- **SMS-SUBMIT** (MO) -
- **SMS-STATUS-REPORT** -
- **SMS-COMMAND** -

□□□□

- **GSM7** - 7□GSM□□□□□SMS 160□□□□
 - **UCS2** - 16□Unicode□□□SMS 70□□□□
 - **8-bit** - □□□□□□□SMS 140□□□□
-

M3UA□□

- **DOWN** - □SCTP□□
 - **CONNECTING** - SCTP□□□
 - **ASPUP_SENT** - □□ASPUP□□
 - **INACTIVE** - ASP□□□□□□□
 - **ASPAC_SENT** - □□ASPAC□□
 - **ACTIVE** - □□□□□□
-

□□SS7□□

□□□□□14□□ITU□□24□□ANSI□□□□

□□□□□ITU□□

- □□□3□
 - □□□8□
 - □□□3□
-

SCCP□□□□

- **0** - □□□□□
- **1** - □□□□□□□
- **2** - □□□□□
- **3** - □□□□□

- 4 - □□□□
- 5 - MTP□□
- 6 - □□□□
- 7 - □□□
- 8 - □□□□□□

MAP□□□□

□□	□□	□□
1	unknownSubscriber	□□□□HLR□
27	absentSubscriber	□□□□□
34	systemFailure	□□□□
35	dataMissing	□□□□□□□
36	unexpectedDataValue	□□□□□

□□□□

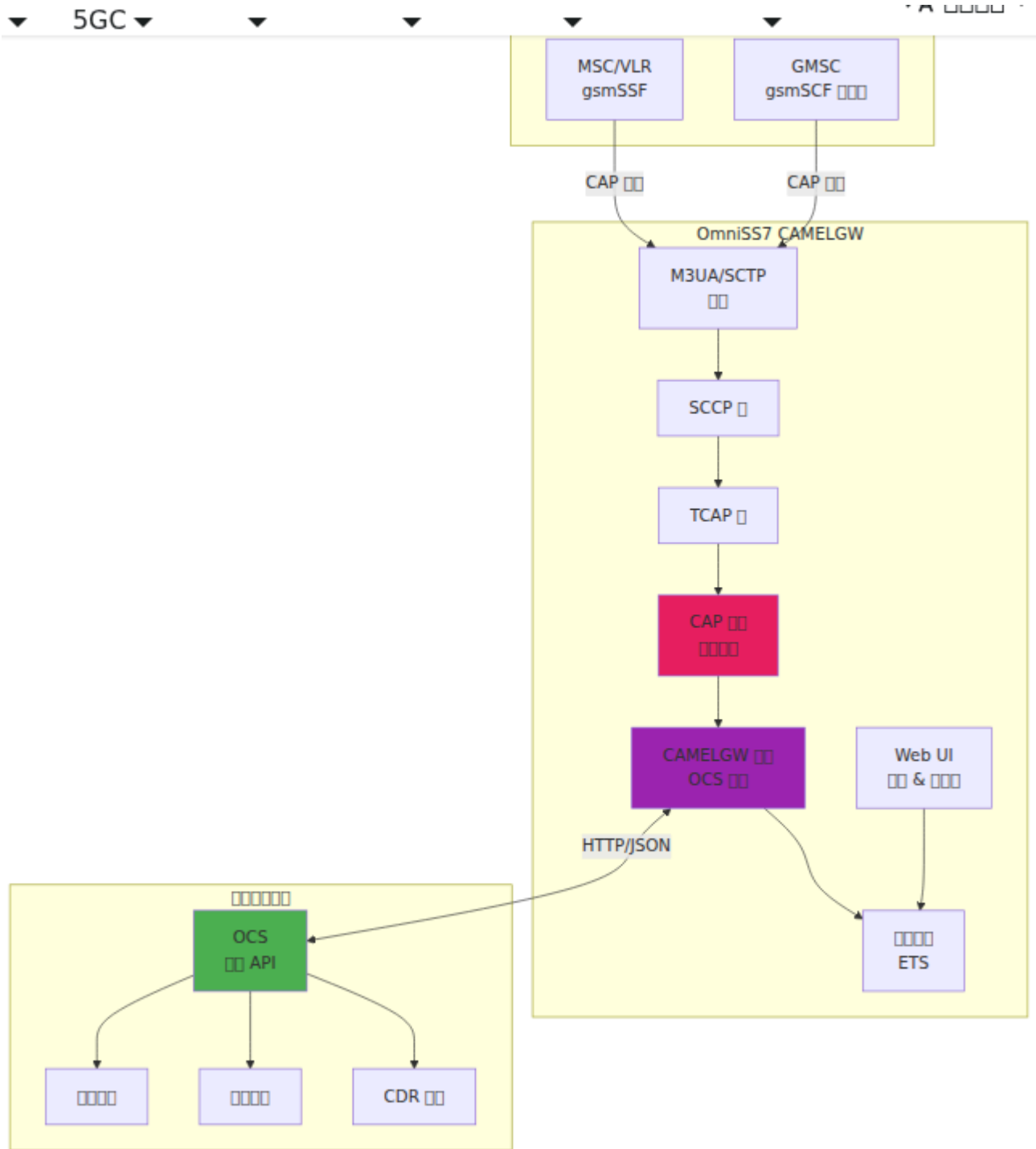
- ← □□□□□
- STP□□
- MAP□□□□□
- □□□□□□
- HLR□□
- □□□□

CAP 表

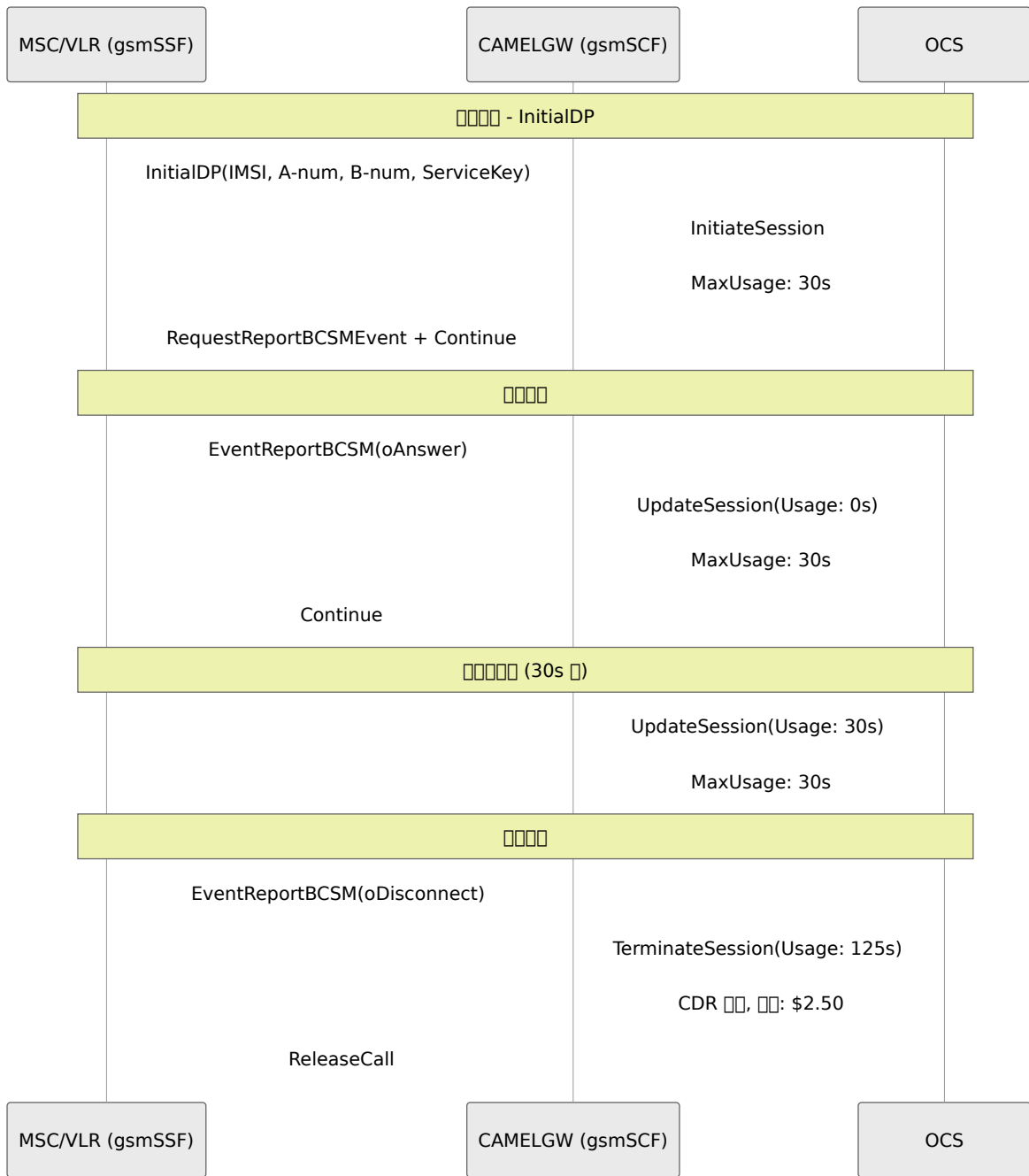
OmniSS7 CAMEL GW 表 CAP 表

表名	表名	表名
CAP v1	CAMEL 表 1	表
CAP v2	CAMEL 表 2	表 SMS
CAP v3	CAMEL 表 3	表 GPRS 表
CAP v4	CAMEL 表 4	表

表 CAP v2 表



□□□□□□



□□

□□□□

- □□□□□□ OmniSS7
- □ MSC/GMSC (gsmSSF) □ M3UA □□

- `config/runtime.exs` API `config/runtime.exs`

`config/runtime.exs` **CAMEL** `config/runtime.exs`

`config/runtime.exs` `config/runtime.exs` CAMEL `config/runtime.exs`

```

config :omniss7,
  # CAP/CAMEL
  cap_client_enabled: true,
  camelgw_mode_enabled: true,

  #
  map_client_enabled: false,
  hlr_mode_enabled: false,
  smsc_mode_enabled: false,

  # CAP/CAMEL
  # CAP
  # : :v1, :v2, :v3, :v4
  cap_version: :v2,

  # OCS
  ocs_enabled: true,
  ocs_url: "http://your-ocs-server/api/charging",
  ocs_timeout: 5000, #
  ocs_auth_token: "your-api-token" # OCS

  # CAMEL M3UA
  # ASP CAP
  cap_client_m3ua: %{
    mode: "ASP",
    callback: {CapClient, :handle_payload, []},
    process_name: :camelgw_client_asp,

    # CAMEL GW
    local_ip: {10, 179, 4, 13},
    local_port: 2905,

    # MSC/GMSC - gsmSSF
    remote_ip: {10, 179, 4, 10},
    remote_port: 2905,

    # M3UA
    routing_context: 1,
    network_appearance: 0,
    asp_identifier: 13
  }

```

Web UI

Web UI CAMEL

```
config :control_panel,  
  use_additional_pages: [  
    {SS7.Web.EventsLive, "/events", "SS7"},  
    {SS7.Web.TestClientLive, "/client", "SS7"},  
    {SS7.Web.M3UAStatusLive, "/m3ua", "M3UA"},  
    {SS7.Web.CAMELSessionsLive, "/camel_sessions", "CAP"},  
    {SS7.Web.CAMELRequestLive, "/camel_request", "CAP"}  
  ],  
  page_order: ["/events", "/client", "/m3ua", "/camel_sessions",  
               "/camel_request", "/application", "/configuration"]
```

CAP

gsmSSF → gsmSCF

Message	Priority	Direction	Function
InitialDP	0	gsmSSF - gsmSCF	handle_initial_dp/1
EventReportBCSM	6	gsmSCF gsmSSF gsmSCF	handle_event_report_bcsm/1
ApplyChargingReport	71	gsmSSF gsmSCF	handle_apply_charging_report
AssistRequestInstructions	16	gsmSCF gsmSSF	handle_assist_request_instructions

gsmSCF → gsmSSF

Event	Priority	Direction	Function
Connect	20	Uplink	CapRequestGenerator.connect_rec
Continue	31	Uplink	CapRequestGenerator.continue_re
ReleaseCall	22	Uplink	CapRequestGenerator.release_cal
RequestReportBCSMEvent	23	Uplink	CapRequestGenerator.request_rep

名前	タイプ	値	コメント
ApplyCharging	35		CapRequestGenerator.apply_charg

Web UI

CAMEL

URL: http://localhost/camel_sessions

CAMEL

-
- OTID ID
- **CAP** - InitialDP (CAP v1/v2/v3/v4)
- IMSI A B
-
-

- ID, IMSI, OTID

InitialDP

- InitialDP
- CDR

CAP InitialDP CAP MSC CAP

CAMEL

URL: http://localhost/camel_request

CAP

InitialDP

- InitialDP, Connect, ReleaseCall
- SCCP/M3UA
- 20
- OTID
- /

InitialDP

1. InitialDP -

-
- A
- B

2. Connect -

-

3. ReleaseCall -

- 16=17=31=

4. RequestReportBCSMEEvent -

- AnswerDisconnectAnswerDisconnect

5. Continue -

-

6. ApplyCharging -

- 1-864000
-
- CAMEL

SCCP

-
-
- SSN146 = gsmSSF
- SSN146

M3UA

- OPC5013
- DPC5011

OCS

1. InitialDP

MSC InitialDP CAMELGW

1. CAP - CAP v1/v2/v3/v4
2. CAP - IMSI/

3. **OCS** - `InitiateSession` API
4. `MaxUsage` = 30
5. `SessionStore` (ETS) `CAP`
6. **MSC** - `RequestReportBCSMEvent` + `Continue` `CAP`

Code

```
# InitialDP
%{
  imsi: "310150123456789",
  calling_party_number: "14155551234",
  called_party_number: "14155556789",
  service_key: 1,
  msc_address: "19216800123",
  cap_version: :v2 #
}

# OCS
{:ok, %{max_usage: 30}} # 30

# SessionStore
%{
  call_id: "CAMEL-4B000173",
  initial_dp_data: %{...},
  cap_version: :v2, #
  start_time: 1730246400,
  state: :initiated
}
```

2. `EventReportBCSM - oAnswer`

Code

1. **oAnswer** - MSC
2. **OCS** - `UpdateSession` = 0
3. `OCS`
4. `SessionStore` `:answered`
5. MSC `Continue`

3. Code

□□□□□□□□□□□□□□□□

```
# □ 30 □  
OCS.Client.update_session(call_id, %{} , current_usage)
```

□□ MaxUsage □□ 0□□□□□□□□□□ → □□ ReleaseCall

4. □□□□ (EventReportBCSM - oDisconnect)

□□□□□□□□

1. □□ **oDisconnect** □□ - □□ MSC
2. □□□□□□□□ - □□□□□□□□
3. □□ **OCS** □□ - TerminateSession API
4. □□ **CDR** - □ OCS □□□□□□□□
5. □□□□ - □ SessionStore □□□□
6. □□ **ReleaseCall** - □□□□□□ MSC

CDR □□

CDR □□□ OCS □□□□□□□□□□

□□ **CAMEL** □ **CDR** □□□□

- Account - IMSI □□□□□□
 - Destination - □□□□□□
 - OriginID - □□□□□□□□ (CAMEL-OTID)
 - Usage - □□□□□□□□□□□□
 - Cost - □□□□□
 - IMSI - □□□□ IMSI
 - CallingPartyNumber - A □
 - CalledPartyNumber - B □
 - MSCAddress - □□ MSC □□□□
 - ServiceKey - CAMEL □□□□□
-

□□

□□□□□□□□□□□□□□

1. □□□□□□□□

```
http://localhost/camel_request
```

2. □□ **InitialDP** □

- □□□□□□□□ "InitialDP"
- □□□□□ 100
- □□□□□ 14155551234
- □□□□□ 14155556789
- □□ "□□ InitialDP □□"
- □□□□□ OTID

3. □□□□□

- □□□□□□□□ http://localhost/camel_sessions
- □□□□□ "□□□□" □□□□□

4. □□□□□□□

- □□□□□□□□
- □□ "EventReportBCSM"
- □□□□□□ oAnswer
- □□ "□□ EventReportBCSM □□"
- □□□□□□□□ "□□□□"

5. □□□□□

- □□ "ReleaseCall"
- □□□□□□ 16 □□□□□
- □□ "□□ ReleaseCall □□"
- □□□□□□□□ "□□□□"

0000 MSC 0000

00 MSC CAMEL 00

000 MSC/VLR 0000 CAMEL 000

```
# 0000 MSC 00
ADD CAMELSERVICE:
  SERVICEID=1,
  SERVICEKEY=100,
  GSMSCFADDR="55512341234", # CAMELGW 0000
  DEFAULTCALLHANDLING=CONTINUE;

ADD CAMELSUBSCRIBER:
  IMSI="310150123456789",
  SERVICEID=1,
  TRIGGERTYPE=TERMCALL;
```

0000

00 CAMELGW 00000000 CAP 000

```
# 0000000
tail -f /var/log/omniss7/omniss7.log

# 00 CAP 00
grep "CAP:" /var/log/omniss7/omniss7.log

# 00000000JSON 000
curl http://localhost/api/events | jq '.[ ] | select(.map_event |
startswith("CAP:"))'
```

0000

000000000000000000000000

```
# 100 InitialDP
for i in {1..100}; do
  curl -X POST http://localhost/api/camel/initial_dp \
    -H "Content-Type: application/json" \
    -d '{
      "service_key": 100,
      "calling_number": "1415555'$i'",
      "called_number": "14155556789"
    }'
  sleep 0.1
done
```

□□□□□

Prometheus □□

CAMELGW □ <http://localhost:8080/metrics> □□□□□□

□□□ **CAP** □□□□

- [cap_requests_total{operation}](#) - □□□□□□□□□□initialDP□
requestReportBCSMEEvent□□□□□ CAP □□

□□ **MAP/API** □□□

- [map_requests_total{operation}](#) - □□□□□□□□□□ MAP □□
- [map_request_duration_milliseconds{operation}](#) - □□□□□□□□□□
- [map_pending_requests](#) - □□□□ MAP □□□□□□

M3UA STP □□□□□□□□ STP □□□□

- [m3ua_stp_messages_received_total{peer_name,point_code}](#) - □□□□□□□□□□□□
- [m3ua_stp_messages_sent_total{peer_name,point_code}](#) - □□□□□□□□□□
- [m3ua_stp_routing_failures_total{reason}](#) - □□□□□□□□□□□□

□□□□□

```
# CAP []
curl http://localhost:8080/metrics | grep cap_requests_total

# [] [] InitialDP
curl http://localhost:8080/metrics | grep
'cap_requests_total{operation="initialDP"}'

# MAP [] [] []
curl http://localhost:8080/metrics | grep map_pending_requests
```

[] [] [] []

```
# [] M3UA []
curl http://localhost/api/m3ua-status

# [] OCS []
curl http://localhost/api/ocs-status

# [] [] [] []
curl http://localhost/api/camel/sessions/count
```

[] [] [] []

[] `config/runtime.exs` [] [] [] [] [] [] [] []

```
config :logger,
  level: :info # [] []:debug, :info, :warning, :error

# [] CAP [] [] []
config :logger, :console,
  metadata: [:cap_operation, :otid, :call_id]
```

□□□□

□□□□□□ **CAP** □□

□□ □□□□□□□□□□ MSC □□ InitialDP

□□□

1. M3UA □□□□ `curl http://localhost/api/m3ua-status`
2. MSC CAMEL □□□□□□□□□□gsmSCF □□□
3. SCCP □□□□□□□□□□□ CAMELGW□
4. □□□□□□□□ SCTP □□ 2905□

□□□□□

```
# □□ M3UA □□  
tcpdump -i eth0 sctp  
  
# □□ MSC □□□□□□ CAMELGW  
ss -tuln | grep 2905
```

□□□**OCS** □□

□□□ `INSUFFICIENT_CREDIT` □□□□□

□□□

1. OCS □□□ `curl http://your-ocs-server/api/health`
2. □□□ OCS □□□□
3. OCS □□□□□□□□
4. □ OCS □□□□□
5. □□□□□□□□□□□□

□□□□□

- □□ `runtime.exe` □□ OCS URL □□
- □□ OCS □□□□□□

- curl OCS API
-

EventReportBCSM

EventReportBCSM "OTID"

OTID

OTID

1. OTID
- 2.
3. DTID Continue/End OTID

```
#
iex> CAMELGW.SessionStore.list_sessions()
```

InitialDP

Failed to decode InitialDP

CAP

CAP

1. CAP MSC
2. ASN.1
3. PCAP Wireshark

```
# CAP
tcpdump -i eth0 -w cap_trace.pcap sctp port 2905
```

```
# Wireshark m3ua
wireshark cap_trace.pcap
```

□□□□

□□ **CAP** □□

□□□□□□□□□□ CAP □□□

```
config :omniss7,  
  cap_version_map: %{\br/>    100 => :v2, # □□□□ 100 □□ CAP v2  
    200 => :v3, # □□□□ 200 □□ CAP v3  
    300 => :v4 # □□□□ 300 □□ CAP v4  
  },  
  cap_version: :v2 # □□
```

□□

CAMEL □□□□ OmniSS7 □□□□□□□□□□□□□□□□□□

□ □□□ **CAP** □□□□ (v1/v2/v3/v4)

□ □□ **OCS** □□□□□□□□

□ □□□□□□ (Connect, Release, Continue)

□ □□ **ETS** □□□□□□□□

□ □□ **Web UI** □□□□□□□□□□□□

□ □□□□□□□□□□□□

□ □□□□□□□□□ **CDR** □□

□ □□□□□□□□□□□□

□□□□□□□□

- **CAMEL** □□□□□□□□
- □□□□ - **CAP** □□

□□□□ OmniSS7 CAMEL □□

□□□□□□ 1.0

□□□□□□ 2025-10-26

CAMEL -

LiveView CAMEL/CAP UI
InitialDP CAMEL

1. CAMEL LiveView

- UI CAMEL
- - **InitialDP** -
 - **Connect** -

- **ReleaseCall** - 00/0000
- **RequestReportBCSMEEvent** - 000000
- **Continue** - 000000
- **ApplyCharging** - 00000000/000000

00000

- 0000000000
- 0000000000000000
- 00 SCCP/M3UA 0000000000
 - 00/00000000
 - SSN0000000000
 - OPC/DPC000000
- 0000000000 20 0000
- 00 OTID 000000
- 00/00???
- 000000

000 /camel_request

2. 000 EventLog 000 CAMEL

0000

- paklog_camel/2 - 000 CAMEL/CAP 000000
- lookup_cap_opcode_name/1 - CAP 00000
- find_cap_opcode/1 - 0 JSON 000 CAP 000
- extract_cap_tids/1 - 0 CAP 00000 OTID/DTID
- format_cap_to_json/1 - 0 CAP PDU 000 JSON 00

000 **CAP** 0000

```
0 => "initialDP"
5 => "connect"
6 => "releaseCall"
7 => "requestReportBCSMEvent"
8 => "eventReportBCSM"
10 => "continue"
13 => "furnishChargingInformation"
35 => "applyCharging"
... ( 47 rows)
```

□□□

- □□ CAMEL □□/□□□ JSON □□□□
- □□ TCAP □□□□□□□□/□□/□□/□□□
- SCCP □□□□
- □□□□□□□□□□□□□□
- □□□□□□□□□□□□
- □□□□□ "CAP:" □□□□□□

3. □□□ CapClient

□□□

- □□□□□□□□□□□□ paklog_camel/2 □□
- □□□□□□□□□□□□ MAP (paklog) □ CAP (paklog_camel) □□□□□□
- □□□□□□□□ sccp_m3ua_maker/2 □
- □□□□□□□□ handle_payload/1 □

□□

□□ LiveView □□□□□□□□□□□□□□

```
# config/runtime.exs

config :control_panel,
  use_additional_pages: [
    {SS7.Web.EventsLive, "/events", "SS7 []"},
    {SS7.Web.TestClientLive, "/client", "SS7 [][]"},
    {SS7.Web.M3UAStatusLive, "/m3ua", "M3UA"},
    {SS7.Web.HlrLinksLive, "/hlr_links", "HLR []"},
    {SS7.Web.CAMELSessionsLive, "/camel_sessions", "CAMEL []"},
    {SS7.Web.CAMELRequestLive, "/camel_request", "CAMEL [][]"}
  ],
  page_order: ["/events", "/client", "/m3ua", "/hlr_links",
               "/camel_sessions", "/camel_request",
               "/application", "/configuration"]
```

□□

□□□□□□□□

1. □□□□ `https://your-server:8087/camel_request`
2. □□□□□□□□□□□□
3. □□□□□□
4. □□□□□“□□ SCCP/M3UA □□”□□□□□
5. □□“□□ [RequestType] □□”

□□□□

InitialDP□□□□□

1. □□□□□□□□□□100□
2. □□□□□□□□A-Party□
3. □□□□□□□□B-Party□
4. □□□□ → □□□□ OTID
5. OTID □□□□□□□□□□□□□□

□□□□□**Connect, ReleaseCall** □□

1. InitialDP OTID
2. OTID
3. OTID

□□□□

InitialDP

-
- ISDN
- ISDN

Connect

-

ReleaseCall

- 16 = 17 = 31 =

RequestReportBCSMEvent

- BCSM Answer, Disconnect

Continue

- OTID

ApplyCharging

- 1-864000 -
- -

□□□□

SCCP

- GT
- GT

- SSN 146 = gsmSSF
- SSN 146

M3UA

- OPC 5013
- DPC 5011

JSON

CAMEL JSON

- /
- **TCAP** /
- **CAP** "CAP:initialDP" "CAP:connect"
- **SCCP** /
- **TIDs** OTID/DTID
- JSON CAP PDU

```
{
  "map_event": "CAP:initialDP",
  "direction": "outgoing",
  "tcap_action": "Begin",
  "otid": "A1B2C3D4",
  "sccp_called": {
    "SSN": 146,
    "GlobalTitle": {
      "Digits": "55512341234",
      "NumberingPlan": "isdn_tele",
      "NatureOfAddress_Indicator": "international"
    }
  },
  "event_message": "{ ... full CAP PDU ... }"
}
```

□□□□

UI □□□□ 20 □□□□□□□□

- □□□
- □□□□□□□□□□□□□□□□
- OTID□□ 8 □□□□□□□□□□
- □□□□□□/□□□□
- □□□□□□□□□□□□□□

□□□□

□□□□□□□□□□

- □□□□ OTID
- □□□□□□□□□□□□
- □□□□□□□□□□□□□□

□□□□□□□□

1. □□□□□□

- □□ InitialDP → □□ OTID
- □□□□□□

2. □□□□□□

- □□ RequestReportBCSMEEvent → □□□□
- □□ ApplyCharging → □□□□□□□□□□□□□□□□290 □□
- □□ Connect → □□□□□□□□
- □□□□ ReleaseCall → □□


3. □□□□□□

- □□□□□□□□
- □□ CAMEL □□□□□□

- 参数 "CAP:" 参数

ApplyCharging - 参数

参数

ApplyCharging 参数  参数

参数

- 参数
- 参数
- 参数
- **OCS** 参数

参数

参数 maxCallPeriodDuration

- 参数 1-86400
- 参数
- 参数
 - 60 = 1
 - 290 = 4 50
 - 3600 = 1
 - 86400 = 24

参数 releaseIfDurationExceeded

- 参数 true/false
- 参数 true
- 参数
 - true 参数/参数
 - false 参数 gsmSCF 参数

- gsmSCF 呼叫

呼叫

1. 呼叫 **Connect** 呼叫 **ApplyCharging**

- 呼叫
- 呼叫

2. 呼叫 **RequestReportBCSMEEvent** 呼叫

- 呼叫 `oAnswer` 呼叫 `oDisconnect` 呼叫
- 呼叫
- 呼叫

3. 呼叫

- 呼叫
- 呼叫
- 呼叫 60-300 呼叫

4. 呼叫

- 呼叫 `release=false` 呼叫
- 呼叫

呼叫

呼叫

- 呼叫 **OTID** 呼叫 InitialDP
- 呼叫 1-864000 呼叫
- 呼叫 SSF 呼叫 ApplyCharging
- 呼叫 1 呼叫

呼叫

呼叫 ApplyCharging 呼叫

- 000000000000 ApplyCharging 00
- 00000000 "CAP:applyCharging"
- **CAMEL** 0000000000000000
- **TCAP** 00000000/0000

0000

0000

- LiveView 00000000
- OTID 00000000
- 00000000 20 000
- 0000000000000000

0000

- 000000 `CapRequestGenerator` 00
- 000000 TCAP/CAP 00
- 00 `:TCAPMessages` 00000000
- 00 `CapClient.sccp_m3ua_maker/2` 0000 SCCP 0

0000

- 00 M3UA 0000 `:camelgw_client_asp`
- 00000000 1
- 00 SCCP/M3UA 00

0000

- 0000000000
- 00000000 OTID
- UI 00000000
- 00000000

□□□□

□□□□□

1. □□□□/□□
2. □□□□□□□
3. □□□□□□□
4. □□□□□□□□
5. □□□□□□
6. □□□□□□□□□□
7. □□□□□ PCAP □□
8. CAP □□□□

□□□□

- □□□□ MAP □□□□ paklog □□□
- □ MAP □□□□□□□□□□□
- □□□□□ SCCP/M3UA □□□□
- □ CAMELSessionsLive □□□□□□□□□
- □□□□ M3UA □□□□

□□□□□

- config/runtime.exs - □□□

□□□

- □□□ CapRequestGenerator
- □□ M3UA □□□ CapClient
- □□□□□□□□ M3UA.Server
- □□□□□□□□□□ EventLog
- Phoenix LiveView □□

- 00 UI 0000000000



← □□□□

□□□□□□ OmniSS7 □□□□□□□□□□



1. Web UI □□
2. API □□
3. □□□□
4. □□□□
5. □□□□□□ SCTP □□□

Web UI □□

Web UI □□□□□□□□ Web □□□□□□□□

API

- API - API SS7 API
- API - API
- API - API
- **M3UA** API - M3UA API STP API
- **SMS** API - API SMS API SMSc API

API Web UI

1. API Web API
2. API API `http://localhost`
3. API API

Swagger API API

API API

http://your-server/swagger

Web UI

config/runtime.exs

```
config :control_panel,  
  #  
  page_order: ["/events", "/application", "/configuration"],  
  
  # Web  
  web: %{  
    listen_ip: "0.0.0.0", # IP 0.0.0.0  
    port: 80, # HTTP 443  
    hostname: "localhost", # URL  
    enable_tls: false, # true HTTPS  
    tls_cert: "cert.pem", # TLS  
    tls_key: "key.pem" # TLS  
  }
```

이름	타입	값	설명
page_order	배열	["/events", "/application", "/configuration"]	페이지 순서
listen_ip	문자열	"0.0.0.0"	Web 서버가 수신할 IP 주소
port	정수	80	HTTP 서버가 수신할 포트 번호 (443은 HTTPS용)
hostname	문자열	"localhost"	서버의 URL 주소
enable_tls	부울	false	TLS를 활성화할지 여부 (HTTPS용)
tls_cert	문자열	"cert.pem"	TLS 인증서 파일 경로
tls_key	문자열	"key.pem"	TLS 키 파일 경로

로그 설정

config/runtime.exs에 로그 설정을 추가합니다.

```
config :logger,
  level: :debug # 기본: :debug, :info, :warning, :error
```

□□□□

- `:debug` - □□□□□□
 - `:info` - □□□□□□
 - `:warning` - □□□□□□□□
 - `:error` - □□□□
-

API □□

API □□ URL

`http://your-server/api`

□□□□

- **200** - □□
- **400** - □□□□
- **504** - □□□□

OpenAPI □□

`http://your-server/swagger.json`

□□□□□

Prometheus □□□□

`http://your-server/metrics`

□□□□□□

M3UA/SCTP □□□

- SCTP □□□□□□
- M3UA ASP □□□□
- □□/□□□□□□□□□□

M2PA □□□

- □□□□□□□DOWN → ALIGNMENT → PROVING → READY□
- □□□□□□□/□□□□□□□□□□
- □□□□□□□□□□□□□□□□SCTP□

STP □□□

- □□□□□□□□□□□□□□□□
- □□□□□□□□□□
- □□□□□□□□□□□□□□□□

MAP □□□□□□

- □□□□□□□ MAP □□
- □□□□□□□□□□□□
- □□□□□□□□

CAP □□□

- □□□□□□□ CAP □□
- CAMEL □□□□□

SMSc □□□

- □□□□
- □□□
- □□□□□

Grafana

OmniSS7 Prometheus Grafana



1.

- VLAN
-
- SCTP

2. Web UI

- TLS
-
- IP

3. API

-
- API OAuth
-



1. TPS

- TPS
-
- SCTP

2.


```

# ✓ [] - [] remote_ip
%#123;
  role: :server,
  remote_ip: [#123;10, 0, 2, 100#125;, # []
  # ...
#125;

# x [] - []
%#123;
  role: :server,
  remote_ip: [#123;10, 0, 2, 100#125;, [#123;10, 0, 2,
101#125;], # []
  # ...
#125;

```

[] 1 [] STP [] - []

```

# STP []
config :omniss7,
  m3ua_peers: [
    %{
      peer_id: 1,
      name: "Partner_STP_Redundant",
      role: :client, # [] - []
      # [] IP []
      local_ip: [{213, 57, 23, 200}, {213, 57, 23, 201}],
      local_port: 0,
      # [] - []
      remote_ip: [{213, 57, 23, 100}, {213, 57, 23, 101}],
      remote_port: 2905,
      routing_context: 1,
      point_code: 100,
      network_indicator: :international
    }
  ]

```

[] 2 [] MAP []

```

# 配置 MAP 客户端
config :omniss7,
  map_client_enabled: true,
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :hlr_client_asp,
    # 本地 IP 地址
    local_ip: [{10, 0, 0, 100}, {10, 0, 0, 101}],
    local_port: 2905,
    # STP 配置
    remote_ip: [{10, 0, 0, 1}, {10, 0, 0, 2}],
    remote_port: 2905,
    routing_context: 1
  }

```

3. 配置 STP 客户端

```

# 配置 STP 客户端
config :omniss7,
  sctp_handler: %{
    enabled: true,
    # 本地 IP 地址
    local_ip: [{172, 16, 0, 10}, {172, 16, 0, 11}],
    local_port: 2905,
    point_code: 100
  }

```

4. 配置 - 配置

```
# HLR
config :omniss7,
  m3ua_peers: [
    %{
      peer_id: 1,
      name: "HLR",
      role: :server, # HLR
      # IP
      local_ip: [{10, 0, 1, 10}, {10, 0, 1, 11}],
      local_port: 2905,
      # HLR SCTP INIT IP
      # SCTP HLR IP
      remote_ip: {10, 0, 2, 100}, # IP
      remote_port: 0, #
      routing_context: 1,
      point_code: 100,
      network_indicator: :international
    }
  ]
```

5

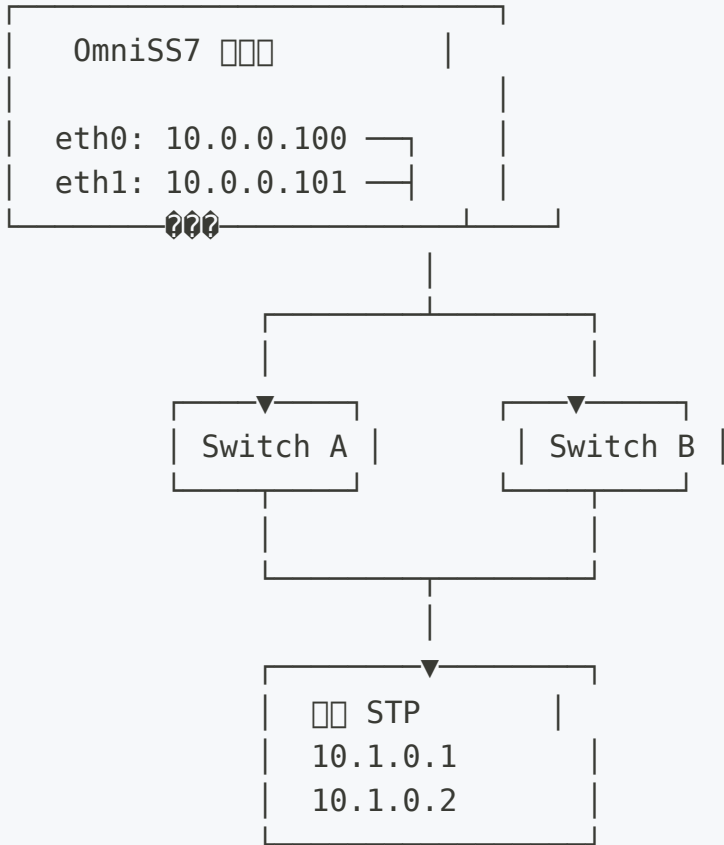
```

# 配置示例
config :omniss7,
  m3ua_peers: [
    # 配置 - 本地 IP
    %{
      peer_id: 1,
      name: "Legacy_STP",
      role: :client,
      local_ip: {10, 0, 0, 1},      # 本地 IP 地址
      local_port: 0,
      remote_ip: {10, 0, 0, 10},
      remote_port: 2905,
      routing_context: 1,
      point_code: 100
    },
    # 配置 - 冗余
    %{
      peer_id: 2,
      name: "Redundant_STP",
      role: :client,
      local_ip: [{10, 0, 0, 2}, {10, 0, 0, 3}], # IP 地址列表
      local_port: 0,
      remote_ip: [{10, 0, 0, 20}, {10, 0, 0, 21}],
      remote_port: 2905,
      routing_context: 2,
      point_code: 200
    }
  ]
]

```

配置示例

配置 1 个 NIC 配置示例



000

```

local_ip: [{10, 0, 0, 100}, {10, 0, 0, 101}] # 00 NIC
remote_ip: [{10, 1, 0, 1}, {10, 1, 0, 2}] # 0000
  
```

000

- 00000000 NIC 000
- 0000000000000000
- 0000000000 1 0

00 2000000


```
[warning] [MULTIHOMING] 10.0.0.100 Partner_STP
assoc_id=1
[info] [MULTIHOMING] 10.0.0.101 000 Partner_STP
assoc_id=1
[info] [MULTIHOMING] 10.0.0.100 Partner_STP assoc_id=1
```

Web UI

Web UI

M3UA

- IP 10.0.0.100
- IP 10.0.0.100 (+1) 10.0.0.100 (+2)
- IP

1.

- NIC
-
-
- -

2. IP

- IP IP -
- - IP
- -

3.

```
# 断网
sudo ip link set eth0 down

# 查看日志
tail -f /var/log/omniss7.log | grep MULTIHOMING

# 恢复网络
sudo ip link set eth0 up
```

4. 配置防火墙

- 配置防火墙规则
- 配置防火墙策略
- 配置防火墙日志

5. 配置 Sctp

```
# 配置 Sctp IP 规则
iptables -A INPUT -p sctp --dport 2905 -s 10.0.0.0/24 -j ACCEPT
iptables -A INPUT -p sctp --dport 2905 -s 10.1.0.0/24 -j ACCEPT
```

验证

配置防火墙

配置 Sctp IP

验证

1. 验证 Erlang Sctp 配置 `erl -eval 'gen_sctp:open(9999, [binary, {ip, {127,0,0,1}}]).'`
2. 验证 Sctp 模块 `lsmod | grep sctp`
3. 验证 Sctp 模块 `sudo modprobe sctp`
4. 验证 Sctp IP `ip addr show`

配置防火墙

配置 Sctp IP

□□□

1. □□ SCTP □□□□
2. □□□□□□□□□□□□□□
3. □□□□□□□□ IP □□ SCTP
4. □□ SCTP □□□□□□


□□□□□□□□□□

□□□□□□ UP □ DOWN □□□□□□

□□□

1. □□□□□ - □□□□□□
2. Sctp □□□□□□ - □□□□□□
3. □□□□□ Sctp □□
4. □□□□□□ MTU □□

□□□□□

- □□□□□ Sctp □□□□□□□□□□
- □□□□□□□□□□□□□□□□
- □□□□□□□□□□□□ 1 □□□□□□□□
- □□□□□□□□□□□□

□□□□

- □□□□□□□ IP □□□□□□□□
- □□□□□□□□□□ IP □□ IP □□□
- □□□□□□□□ STP□HLR□SMS□ MAP □□□□□□□□□
- **Erlang** □□□□□□□□□ Sctp □ Erlang

□□□□□□

□□□□□

- M3UA □□□□

項目	型別	デフォルト値	説明	備考
<code>page_order</code>	配列	空	<code>["/events", "/application", "/configuration"]</code>	優先順位
<code>web.listen_ip</code>	文字列	空	<code>"0.0.0.0"</code>	Web サービスの IP アドレス
<code>web.port</code>	整数	空	<code>80</code>	HTTP/HTTPS のポート番号
<code>web.hostname</code>	文字列	空	<code>"localhost"</code>	ホスト名
<code>web.enable_tls</code>	ブール値	空	<code>false</code>	HTTPS を有効にするかどうか
<code>web.tls_cert</code>	文字列	TLS 証明書	<code>"cert.pem"</code>	TLS 証明書ファイル
<code>web.tls_key</code>	文字列	TLS キー	<code>"key.pem"</code>	TLS キーファイル

SCTP SocketHandler 項目 (`:omniss7`)

```

config :omniss7,
  sctp_handler: %{
    enabled: false,
    local_ip: {127, 0, 0, 1},
    local_port: 2905
  },
  enable_gt_routing: true,
  m3ua_peers: [...],
  m3ua_routes: [...],
  m3ua_gt_routes: [...]

```

配置项	数据类型	默认值	配置值	说明
sctp_handler.enabled	布尔	否	false	是否启用 STP
sctp_handler.local_ip	IP 地址		{127, 0, 0, 1}	M3UA 的 IP 地址
sctp_handler.local_port	端口		2905	M3UA 的 SCTP 端口
enable_gt_routing	布尔	否	false	是否启用 GT 路由

M3UA 配置

属性	型	制約	説明
peer_id	整数	一意	ピアID
name	文字列	一意	名前
role	文字列	一意	:client または :server
local_ip	文字列	一意 :client	ローカルIPアドレス {10, 0, 0, 1} を含む [{10, 0, 0, 1}, {10, 0, 0, 2}]
local_port	整数	一意 :client	ローカルポート番号
remote_ip	文字列	一意	リモートIPアドレス {10, 0, 0, 10} を含む [{10, 0, 0, 10}, {10, 0, 0, 11}]
remote_port	整数	一意 :client	リモートポート番号
routing_context	文字列	一意	M3UA コンテキスト
point_code	文字列	一意	SS7 ポイントコード
network_indicator	文字列	一意	:international または :national

M3UA 属性

項目	型別	長さ	説明
dest_pc	文字列	10	宛先IPアドレス
peer_id	文字列	10	ピアID
priority	文字列	10	優先度
network_indicator	文字列	10	:international 0 :national

M3UA GT 項目

項目	型別	長さ	説明
gt_prefix	文字列	10	GTプレフィックス
peer_id	文字列	10	ピアID
priority	文字列	10	優先度
description	文字列	10	説明
source_ssn	文字列	10	送信SSN
dest_ssn	文字列	10	宛先SSN

MAP 項目 (:omniss7)

```
config :omniss7,  
  map_client_enabled: false,  
  map_client_m3ua: %{  
    mode: "ASP",  
    callback: {MapClient, :handle_payload, []},  
    process_name: :map_client_asp,  
    local_ip: {10, 0, 0, 100},  
    local_port: 2905,  
    remote_ip: {10, 0, 0, 1},  
    remote_port: 2905,  
    routing_context: 1  
  }  
}
```

名前	型	デフォルト値	説明
<code>map_client_enabled</code>	bool	false	MAP クライアントの有効化フラグ
<code>map_client_m3ua.mode</code>	enum	"ASP"	M3UA モード。ASP または SGP を指定する
<code>map_client_m3ua.callback</code>	void (*)	{MapClient, :handle_payload, []}	呼び出しバック関数
<code>map_client_m3ua.process_name</code>	string	:map_client_asp	プロセス名
<code>map_client_m3ua.local_ip</code>	string	-	ローカル IP アドレス
<code>map_client_m3ua.local_port</code>	uint16_t	2905	ローカル SCTP ポート番号
<code>map_client_m3ua.remote_ip</code>	string	-	リモート STP/SGP IP アドレス
<code>map_client_m3ua.remote_port</code>	uint16_t	2905	リモート SCTP ポート番号
<code>map_client_m3ua.routing_context</code>	string	-	M3UA ルーティングコンテキスト

SMS 配置 (:omniss7)

```
config :omniss7,  
  auto_flush_enabled: false,  
  auto_flush_interval: 10_000,  
  auto_flush_dest_smsc: nil,  
  auto_flush_tps: 10
```

配置项	类型	默认值	说明
auto_flush_enabled	布尔	false	是否启用 SMS 自动刷新
auto_flush_interval	整数	10000	自动刷新的间隔 (毫秒)
auto_flush_dest_smsc	字符串/nil	nil	指定 SMS 发送的 SMSC 号码, nil = 默认
auto_flush_tps	整数	10	每秒发送的 SMS 数量

HTTP API 配置 (:omniss7)

SMS 服务通过 HTTP API 与外部系统交互

```
config :omniss7,  
  smsc_api_base_url: "https://10.5.198.200:8443",  
  frontend_name: "omni-smsc01" # 前端名称 hostname_SMSc
```

API 配置

項目	種別	必須	値	説明
smsc_api_base_url	文字列	必須	"https://10.5.198.200:8443"	SMS API URL
frontend_name	文字列	必須	"{hostname}_SMS"	フロントエンド名

API 一覧

- POST /api/frontends - フロントエンド登録
- POST /api/messages_raw - SMS 送信
- GET /api/messages - SMS 一覧取得 (smsc 指定)
- PATCH /api/messages/{id} - SMS 更新
- PUT /api/messages/{id} - SMS 削除
- POST /api/events - イベント登録
- GET /api/status - ステータス取得

設定

API 接続に 5 分間隔で接続を試みます

- SMS 送信
- 接続
- 接続エラー
- JSON 形式

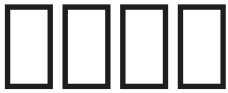
その他

- SSL 接続
- HTTP 接続 5 分間隔
- ISO 8601 形式
- API 形式 JSON 形式/形式

□□□□

- ← □□□□
- STP □□
- MAP □□□□
- SMS □□□□
- HLR □□

OmniSS7 □ Omnitouch □□□□□□



← [←](#)

OmniSS7 [←](#)



1. [←](#)
 2. [←](#)
 3. [HLR ←](#)
 4. [SMSc ←](#)
 5. [STP ←](#)
 6. [CAMEL ←](#)
 7. [← NAT ←](#)
 8. [M3UA ←](#)
 9. [←](#)
 10. [←](#)
 11. [←](#)
-



OmniSS7 [←](#) `config/runtime.exe` [←](#)

- **STP** ← - [←](#)
- **HLR** ← - [←](#)
- **SMSc** ← - [←](#)
- **CAMEL GW** ← - [CAMEL ←](#)

[←](#): `config/runtime.exe`

配置选项

配置选项列表

配置项	数据类型	默认值	描述	关联功能
<code>map_client_enabled</code>	布尔	false	是否启用 MAP 到 M3UA 的转换	MAP
<code>hlr_mode_enabled</code>	布尔	false	是否启用 HLR 模式	HLR
<code>smsc_mode_enabled</code>	布尔	false	是否启用 SMSc 模式	SMSc
<code>cap_client_enabled</code>	布尔	false	是否启用 CAP 到 CAMEL 的转换	CAMEL GW
<code>camelgw_mode_enabled</code>	布尔	false	是否启用 CAMEL 模式	CAMEL GW
<code>ussd_gateway_enabled</code>	布尔	false	是否启用 USSD 网关 (HTTP/JSON 格式)	USSD GW

示例:

```
config :omniss7,  
  map_client_enabled: true,  
  hlr_mode_enabled: true,  
  smsc_mode_enabled: false
```

HLR 配置

HLR 配置选项列表

HLR API

Property	Type	Default Value	Required	Description
<code>hlr_api_base_url</code>	String	-	Yes	HLR API URL
<code>hlr_api_verify_ssl</code>	Boolean	false	Yes	Whether to verify SSL for HLR API
<code>hlr_service_center_gt_address</code>	String	-	Yes	UpdateLocation HLR GT address
<code>smsc_service_center_gt_address</code>	String	-	Yes	SRI-for-SM SMSC GT address

Example:

```
config :omniss7,
  hlr_api_base_url: "https://10.180.2.140:8443",
  hlr_api_verify_ssl: false,
  hlr_service_center_gt_address: "55512341111",
  smsc_service_center_gt_address: "55512341112"
```

AlertServiceCenter

UpdateLocation HLR SMSc GT alertServiceCenter

項目	単位	値	単位	説明
hlr_smsc_alert_gts	分	[]	分	alertServiceCenter SMS Sc 設定
hlr_alert_location_expiry_seconds	分	172800	分	48 時間 0 分 設定

例:

```
config :omniss7,
  hlr_smsc_alert_gts: [
    "15559876543",
    "15559876544"
  ],
  hlr_alert_location_expiry_seconds: 172800 # 48 分
```

MSISDN ↔ IMSI 設定

MSISDN 設定 IMSI 設定 HLR 設定 MSISDN ↔ IMSI 設定

項目	単位	値	説明
hlr_imsi_plmn_prefix	桁	"50557"	IMSI の MCC+MNC (MCC+MNC)
hlr_msisdn_country_code	桁	"61"	IMSI→MSISDN の国番号
hlr_msisdn_nsn_offset	桁	0	MSISDN の NSN のオフセット
hlr_msisdn_nsn_length	桁	9	MSISDN の NSN の長さ

例2 の設定:

```
config :omniss7,
  hlr_imsi_plmn_prefix: "50557",      # MCC 505 + MNC 57
  hlr_msisdn_country_code: "99",     # 桁 2 の国番号
  hlr_msisdn_nsn_offset: 2,          # 桁 2 の NSN オフセット
  hlr_msisdn_nsn_length: 9           # 桁 9 の NSN
```

例3 の設定:

```
config :omniss7,
  hlr_imsi_plmn_prefix: "50557",      # MCC 505 + MNC 57
  hlr_msisdn_country_code: "999",    # 桁 3 の国番号
  hlr_msisdn_nsn_offset: 3,          # 桁 3 の NSN オフセット
  hlr_msisdn_nsn_length: 8           # 桁 8 の NSN
```

例: `nsn_offset` の設定例

- 国番号 "9" の場合 → `nsn_offset: 1`
- 国番号 "99" の場合 → `nsn_offset: 2`

- "999" 3 → nsn_offset: 3

InsertSubscriberData (ISD) []

UpdateLocation [] VLR [] ISD [] HLR []
 InsertSubscriberData []

Field	Type	Value	Unit	Description
isd_network_access_mode	enum	:packetAndCircuit	enum	Access mode options: :packetAndCircuit, :packetOnly, :circuitOnly
isd_send_ss_data	boolean	true	boolean	Send SS data for ISD #2
isd_send_call_barring	boolean	true	boolean	Send call barring for ISD #3

Example:

```
config :omniss7,
  isd_network_access_mode: :packetAndCircuit,
  isd_send_ss_data: true,
  isd_send_call_barring: true
```

CAMEL []

CAMEL [] CAMEL [] HLR [] CAMEL []

項目	単位	値	説明
camel_service_key	整数	11_110	サービスキー
camel_trigger_detection_point	文字列	:termAttemptAuthorized	CAM :termAttemptAuthorized :termAttemptAuthorized
camel_gsmcf_gt_address	文字列	(GT)	GT CAM GT

例:

```
config :omniss7,
  camel_service_key: 11_110,
  camel_trigger_detection_point: :termAttemptAuthorized
```

VLR

PRN HLR

項目	単位	値	説明
home_vlr_prefixes	文字列	["5551231"]	VLR GT

例:

```
config :omniss7,
  home_vlr_prefixes: ["5551231", "5551234"]
```

SMSc 配置

配置项说明

SMSc API 配置

配置项	数据类型	默认值	是否必填	说明
<code>smsc_api_base_url</code>	字符串	-	否	SMSc API 接口 URL
<code>smsc_api_verify_ssl</code>	布尔值	false	否	SMSc API 是否验证 SSL 证书
<code>smsc_name</code>	字符串	"{hostname}_SMSc"	否	SMSc 名称
<code>smsc_service_center_gt_address</code>	字符串	-	否	SMSc 服务中心 GT 地址

示例:

```
config :omniss7,  
  smsc_api_base_url: "https://10.179.3.219:8443",  
  smsc_api_verify_ssl: false,  
  smsc_name: "ipsmgw",  
  smsc_service_center_gt_address: "55512341112"
```

例: 例 5 例 SMS.FrontendRegistry 例

例

例	例	例	例	例
auto_flush_enabled	例	true	例	例 SMS 例
auto_flush_interval	例	10_000	例	例
auto_flush_dest_smsc	例	-	例	例 SMSC 例
auto_flush_tps	例	10	例	例/例

例:

```
config :omniss7,  
  auto_flush_enabled: true,  
  auto_flush_interval: 10_000,  
  auto_flush_dest_smsc: "ipsmgw",  
  auto_flush_tps: 10
```

STP 例

M3UA 例 STP 例

STP

Property	Type	Default Value	Required	Description
<code>sctp_handler.enabled</code>	Boolean	<code>false</code>	Optional	Whether SctpHandler is enabled.
<code>sctp_handler.local_ip</code>	List of IP addresses	<code>{127, 0, 0, 1}</code>	Optional	Local IP addresses for SctpHandler. Example: <code>{10, 0, 0, 1}</code> , <code>{10, 0, 0, 2}</code> .
<code>sctp_handler.local_port</code>	Integer	<code>2905</code>	Optional	Local port for SctpHandler.
<code>sctp_handler.point_code</code>	Integer	-	Optional	SS7 point code for SctpHandler.

Example IP:

```
config :omniss7,
  sctp_handler: %{
    enabled: true,
    local_ip: {10, 179, 4, 10},
    local_port: 2905,
    point_code: 100
  }
```

Example SctpHandler:

```

config :omniss7,
  sctp_handler: %{
    enabled: true,
    # IP
    local_ip: [{10, 179, 4, 10}, {10, 179, 4, 11}],
    local_port: 2905,
    point_code: 100
  }

```

配置: Sctp 配置 Sctp 配置

配置

配置	配置	配置	配置	配置
enable_gt_routing	配置	false	配置	配置 PC 配置 GT 配置

配置:

```

config :omniss7,
  enable_gt_routing: true

```

M3UA/M2PA 配置

配置 m3ua_peers 配置 M3UA 配置 M2PA 配置 STP 配置 M2PA 配置

配置

項目	型	初期値	単位	説明
peer_id	文字列	-	文字列	ピア ID
name	文字列	-	文字列	名前
protocol	文字列	:m3ua	文字列	プロトコル名 :m3ua または :m2pa
role	文字列	:client	文字列	役割名 :client, :server, :asp, :sgp
local_ip	文字列	-	文字列	ローカル IP アドレス
local_port	整数	-	整数	SCTP ポート番号 M3UA: 2905, M2PA: 3565
remote_ip	文字列	-	文字列	リモート IP アドレス
remote_port	整数	-	整数	SCTP ポート番号
routing_context	文字列	-	文字列	M3UA ルーティングコンテキスト M3UA

項目	型	値	型	説明
<code>point_code</code>	int	-	int	ポイントコード
<code>network_indicator</code>	int	<code>:international</code>	int	ネットワークインジケータ : <code>international</code> (国際) : <code>national</code> (国内)
<code>initiate_connection</code>	bool	<code>true</code>	bool	SCTP を開始するかどうか

M2PA 設定

項目	型	値	説明	デフォルト
<code>adjacent_point_code</code>	int	-	隣接 M2PA ポイントコード	0

設定例

M2PA を有効にするには `SCTP.SocketHandler` を有効にし、M2PA を有効にする必要がある場合は `SCTP.SocketHandler` を `M2PA.SocketHandler` に変更する必要があります。

隣接 M3UA の設定:

```
config :omniss7,  
  m3ua_peers: [  
    %{  
      peer_id: 1,  
      name: "HLR_East",  
      protocol: :m3ua,  
      role: :sgp,  
      local_ip: {10, 179, 4, 10},  
      local_port: 2905,  
      remote_ip: {10, 179, 4, 20},  
      remote_port: 2905,  
      point_code: 100,  
      network_indicator: :international  
    }  
  ]  
]
```

配置M2PA:

```
config :omniss7,  
  sctp_handler: %{  
    enabled: true,  
    local_ip: {10, 179, 4, 10},  
    local_port: 3565,  
    point_code: 100  
  },  
  m3ua_peers: [  
    %{  
      peer_id: 2,  
      name: "M2PA_Link_STP_West",  
      protocol: :m2pa,  
      role: :client,  
      local_ip: {10, 179, 4, 10},  
      local_port: 3565,  
      remote_ip: {10, 179, 4, 30},  
      remote_port: 3565,  
      point_code: 100,  
      adjacent_point_code: 200  
    }  
  ]  
]
```

M3UA 配置

配置 M3UA 路由表

名称	类型	格式	值	描述
m3ua_routes	列表	[]		配置 M3UA 路由表

配置 M3UA 路由表 m3ua_routes

- **dest_pc**: 目的 PC 地址
- **peer_id**: 对端 ID
- **priority**: 优先级 - 值越小 = 优先级越高
- **network_indicator**: 网络指示符 :international 或 :national

配置:

```
config :omniss7,
  m3ua_routes: [
    # 目的 PC 100
    %{dest_pc: 100, peer_id: 1, priority: 1, network_indicator:
:international},
    # 目的 PC 200
    %{dest_pc: 200, peer_id: 2, priority: 1, network_indicator:
:international},
    # 目的 PC 300
    %{dest_pc: 300, peer_id: 3, priority: 1, network_indicator:
:international},
    %{dest_pc: 300, peer_id: 4, priority: 2, network_indicator:
:international}
  ]
```

M3UA 配置

配置 M3UA 路由表 SCCP 网络指示符

項目	型	範囲	単位	説明
m3ua_gt_routes	配列	[]	個	グローバルルーティング情報

グローバルルーティング情報は m3ua_gt_routes 配列で定義される。

項目

- **gt_prefix**: グローバルルーティングプレフィックス - グローバルルーティング
- **peer_id**: グローバルルーティングピアID - 0 から開始
- **priority**: 優先度 - 優先度 = 優先度
- **description**: 説明

項目

- **source_ssn**: ソースSSN
- **source_tt**: ソースタイプ
- **source_npi**: ソースNPI
- **source_nai**: ソースNAI

項目

- **dest_ssn**: デスティネーションSSN
- **dest_tt**: デスティネーションタイプ
- **dest_npi**: デスティネーションNPI
- **dest_nai**: デスティネーションNAI

項目

- **タイプ (TT)**: 0=グローバル, 1=ローカル, 2=グローバル, 3=ローカル
- **NPI**: 0=グローバル, 1=ISDN(E.164), 6=グローバル(E.212)
- **NAI**: 0=グローバル, 1=ローカル, 3=グローバル, 4=ローカル
- **SSN**: 6=HLR, 7=VLR, 8=MSC, 9=EIR, その他

項目:

```

config :omniss7,
  m3ua_gt_routes: [
    # 1234
    %{gt_prefix: "1234", peer_id: 1, priority: 1, description: "1234"},
    %{gt_prefix: "44", peer_id: 2, priority: 1, description: "44"},

    # 61
    %{
      gt_prefix: "61",
      peer_id: 3,
      priority: 1,
      description: "61 TT 0→1 61",
      source_tt: 0, # 61 TT=0
      dest_tt: 1 # 61 TT=1
    },

    # NPI 49
    %{
      gt_prefix: "49",
      peer_id: 1,
      priority: 1,
      description: "49→ISDN NPI 49",
      source_npi: 6, # 49 NPI=6/E.212
      dest_npi: 1 # 49 NPI=1/ISDN/E.164
    },

    # 86 SSN 86
    %{
      gt_prefix: "86",
      source_ssn: 8, # 86 SSN=8/MS
      peer_id: 3,
      dest_ssn: 6, # 86 SSN=6/HLR
      priority: 1,
      description: "86",
      source_tt: 0,
      dest_tt: 2,
      source_npi: 6,
      dest_npi: 1,
      source_nai: 4,
      dest_nai: 3
    },
  ],






```

```
# CAMEL/CAP
%{
  gt_prefix: "",
  peer_id: 1,
  priority: 99,
  description: "CAMEL/CAP"
}
]
```
















CAMEL

CAMEL  CAP 






CAMEL

 CAMEL/CAP   `cap_client_enabled: true` 
`camelgw_mode_enabled: true` 

CAP

			 	
<code>cap_version</code>		<code>:v2</code>		CAP  <code>:v1</code> , <code>:v2</code> , <code>:v3</code> ,  <code>:v4</code>
<code>camel_gsmscf_gt_address</code>	 	( GT)		 gsmSCF  CAMEL 

CAP

- `:v1` →  OID: 0.4.0.0.1.0.50.0
- `:v2` →  OID: 0.4.0.0.1.0.50.1 - 
- `:v3` →  OID: 0.4.0.0.1.21.3.4

- :v4 → `OID: 0.4.0.0.1.23.3.4`

OID: `OID 0.4.0.0.1.23.3.4`

OID:

```
config :omniss7,
  cap_client_enabled: true,
  camelgw_mode_enabled: true,
  cap_version: :v2,
  camel_gsmcf_gt_address: "68988411553"
```

CGrateS

CGrateS `CGrateS`

OID	Value	Description
<code>cgrates_enabled</code>	<code>false</code>	CGrateS
<code>cgrates_url</code>	-	CGrateS JSON-RPC URL
<code>cgrates_tenant</code>	<code>"cgrates.org"</code>	CGrateS
<code>cgrates_request_type</code>	<code>"*prepaid"</code>	Request types: <code>"*prepaid"</code> , <code>"*postpaid"</code> , <code>"*pseudoprepaid"</code>
<code>cgrates_timeout</code>	<code>5000</code>	CGrateS

☐☐:

```
config :omniss7,  
  cgrates_enabled: true,  
  cgrates_url: "http://localhost:2080/jsonrpc",  
  cgrates_tenant: "cgrates.org",  
  cgrates_request_type: "*prepaid",  
  cgrates_timeout: 5000
```

CAP M3UA ☐☐

CAMEL ☐☐☐☐☐☐ M3UA ☐☐☐☐ CAP ☐☐☐

☐☐	☐ ☐	☐ ☐ ☐	☐ ☐	☐☐
cap_client_m3ua	☐ ☐	-	☐	CAP M3UA ☐☐☐☐☐☐ map_client_m3ua ☐☐ ☐☐☐☐☐☐☐☐ opc ☐ dpc ☐☐☐

☐☐:

```
config :omniss7,  
  cap_client_m3ua: %{\br/>    mode: "ASP",  
    callback: {CapClient, :handle_payload, []},  
    process_name: :camelgw_client_asp,  
    local_ip: {10, 5, 198, 200},  
    local_port: 2905,  
    remote_ip: {10, 179, 4, 10},  
    remote_port: 2905,  
    routing_context: 4,  
    opc: 5013,      # ☐☐☐☐☐  
    dpc: 5011      # ☐☐☐☐☐  
  }  
}
```

GT NAT

GT NAT = ...
GT NAT = ...

Key	Type	Default	Unit	Description
<code>gt_nat_enabled</code>	bool	<code>false</code>	bool	GT NAT enabled
<code>gt_nat_rules</code>	list	<code>[]</code>	list	GT NAT rules

GT NAT rules

- `calling_prefix`: GT calling prefix
- `called_prefix`: GT called prefix
- `weight`: weight = 100 - weight
- `response_gt`: response GT

GT NAT

- weight = 100 - weight
- ...
- `calling_prefix` & `called_prefix` ...

GT:

```
config :omniss7,  
  gt_nat_enabled: true,  
  gt_nat_rules: [  
    # 0000000000000000 "8772" 0000 "555"  
    %{calling_prefix: "8772", called_prefix: "555", weight: 1,  
response_gt: "111111"},  
  
    # 0000000000000000 "8772"  
    %{calling_prefix: "8772", weight: 10, response_gt:  
"68988411553"},  
  
    # 0000000000000000 "555"  
    %{called_prefix: "555", weight: 10, response_gt:  
"68988411554"},  
  
    # 00000000 "8773"  
    %{calling_prefix: "8773", weight: 10, response_gt:  
"68988411554"},  
  
    # 000000000000000000000000  
    %{weight: 100, response_gt: "68988411555"}  
  ]
```

00: GT NAT 00 00000000000000

M3UA 0000

M3UA 000000 MAP 000000000000000000000000 MAP 000000

属性名	属性タイプ	デフォルト値	必須	説明
<code>map_client_m3ua.mode</code>	文字列	-	否	ASP のモード。デフォルトは "ASP"。
<code>map_client_m3ua.callback</code>	関数ポインタ	-	否	MapC のコールバック関数。デフォルトは <code>MapC_handle_payload</code> 。
<code>map_client_m3ua.process_name</code>	文字列	-	否	プロセス名。デフォルトは "map_client_m3ua"。
<code>map_client_m3ua.local_ip</code>	IP アドレス	-	否	ローカル IP アドレス。デフォルトは <code>{10, 0, 0, 1}</code> 。
<code>map_client_m3ua.local_port</code>	ポート番号	2905	否	SCTP のローカルポート番号。
<code>map_client_m3ua.remote_ip</code>	IP アドレス	-	否	リモート IP アドレス。デフォルトは <code>{10, 0, 0, 10}</code> 。
<code>map_client_m3ua.remote_port</code>	ポート番号	2905	否	SCTP のリモートポート番号。
<code>map_client_m3ua.routing_context</code>	文字列	-	否	M3UA のルーティングコンテキスト ID。
<code>map_client_m3ua.opc</code>	文字列	5013	否	MAP のオペレーティングポイントコード (OPC)。デフォルトは 5013。

名前	タイプ	値	デフォルト	説明
map_client_m3ua.dpc	整数	5011	なし	MAP クライアントの DPC 番号。5011 がデフォルト。
map_client_m3ua.receive_watchdog	ブール値	true	なし	ASP の receive_watchdog を有効にするかどうか。SCTP の receive_watchdog が false の場合は黄色で表示されます。
map_client_m3ua.receive_watchdog_idle	整数	15	なし	SCTP の receive_watchdog_idle を設定する。true の場合、デフォルトは 15。

IP アドレス:

```

config :omniss7,
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :hlr_client_asp,
    local_ip: {10, 179, 4, 11},
    local_port: 2905,
    remote_ip: {10, 179, 4, 10},
    remote_port: 2905,
    routing_context: 1,
    opc: 5013,      # 2-114-5
    dpc: 5011      # 2-114-3
  }

```

IP アドレス: X-Y-Z (X * 2048) + (Y * 8) + Z (2-114-5) = (2 * 2048) + (114 * 8) + 5 = 5013

配置 SCTP 参数:

```
config :omniss7,  
  map_client_m3ua: %{\br/>    mode: "ASP",  
    callback: {MapClient, :handle_payload, []},  
    process_name: :hlr_client_asp,  
    # 本地 IP 地址  
    local_ip: [{10, 179, 4, 11}, {10, 179, 4, 12}],  
    local_port: 2905,  
    # 远程 IP 地址 STP 地址  
    remote_ip: [{10, 179, 4, 10}, {10, 179, 4, 20}],  
    remote_port: 2905,  
    routing_context: 1  
  }  
}
```

注意: 配置 SCTP 参数时，请务必配置 **receive_watchdog** 和 **receive_watchdog_idle** 参数。

配置说明

配置 SCTP 参数 **receive_watchdog** 和 **receive_watchdog_idle** 时，请务必配置 **receive_watchdog** 参数为 **ESTABLISHED** 状态，以确保 SCTP 连接正常。

配置 SCTP 参数 **receive_watchdog** 和 **receive_watchdog_idle** 时，请务必配置 **receive_watchdog** 参数为 **NOTIFY** 状态，请参考 [RFC 4666 §3.8](#) M3UA BEAT 参数 — SCTP 参数 **receive_watchdog** 和 **receive_watchdog_idle** 参数。

参数	数据类型	默认值	说明
<code>receive_watchdog</code>	布尔值	<code>true</code>	配置 <code>receive_watchdog</code> 参数为 <code>false</code> 时，SCTP 连接将无法正常建立。
<code>receive_watchdog_idle</code>	整数	<code>15</code>	配置 <code>receive_watchdog</code> 参数为 <code>false</code> 时，SCTP 连接将无法正常建立。

注意 — 配置 ASP 参数时:

```
config :omniss7,  
  map_client_m3ua: %{\br/>    mode: "ASP",  
    local_ip: {10, 179, 4, 11},  
    local_port: 2905,  
    remote_ip: {10, 179, 4, 10},  
    remote_port: 2905,  
    routing_context: 1,  
    receive_watchdog: false      # 00 - 00 SG 00000000  
  }  
}
```

00 - 00000000:

```
config :omniss7,  
  map_client_m3ua: %{\br/>    mode: "ASP",  
    local_ip: {10, 179, 4, 11},  
    local_port: 2905,  
    remote_ip: {10, 179, 4, 10},  
    remote_port: 2905,  
    routing_context: 1,  
    receive_watchdog: true,  
    receive_watchdog_idle: 30    # 0 30 00000000000015 00  
  }  
}
```

00000000

000000000000000000000000Web 000API 0000000000

□□□□□

□□	□□	□□ □	□ □	□□
<code>license_client.license_server_api_urls</code>	□□□ □□	-	□	□□□□□□ API □□ URL
<code>license_client.licensee</code>	□□□	-	□	□□□□□□□□□□

□□:

```
config :license_client,  
  license_server_api_urls: ["https://localhost:10443/api"],  
  licensee: "OmniTouch Network Services Pty. Ltd."
```

Web

Property	Type	Value
<code>control_panel.parent_application_readable_name</code>	String	"OmniSS7 Stack"
<code>control_panel.use_additional_pages</code>	Boolean	[]
<code>control_panel.page_order</code>	Array	[]
<code>ControlPanelWeb.Endpoint.url.host</code>	String	"0.0.0.0"
<code>ControlPanelWeb.Endpoint.https.port</code>	Integer	8087
<code>ControlPanelWeb.Endpoint.https.keyfile</code>	String	"priv/cert/omnit"
<code>ControlPanelWeb.Endpoint.https.certfile</code>	String	"priv/cert/omnit"

□□:

```
config :control_panel, ControlPanelWeb.Endpoint,  
  url: [host: "0.0.0.0", path: "/"],  
  https: [  
    port: 8087,  
    keyfile: "priv/cert/omnitouch.pem",  
    certfile: "priv/cert/omnitouch.crt"  
  ],  
  parent_application_readable_name: "OmniSS7 Stack STP"  
  
config :control_panel,  
  use_additional_pages: [  
    {SS7.Web.EventsLive, "/events", "SS7 □□"},  
    {SS7.Web.M3UAStatusLive, "/m3ua", "□□□"}  
  ],  
  page_order: ["/events", "/m3ua", "/application",  
"/configuration"]
```

REST API 配置

配置项	数据类型	配置值	说明
<code>start_http_server</code>	布尔	<code>true</code>	是否启动 HTTP 服务，默认 8080 端口
<code>api_ex.api.port</code>	整数	<code>8445</code>	API 服务的端口
<code>api_ex.api.listen_ip</code>	字符串	<code>"0.0.0.0"</code>	API 服务的监听 IP
<code>api_ex.api.product_name</code>	字符串	<code>"OmniSS7"</code>	API 服务的产品名称
<code>api_ex.api.title</code>	字符串	<code>"API - OmniSS7"</code>	Swagger UI 的标题
<code>api_ex.api.hostname</code>	字符串	<code>"localhost"</code>	API 服务的宿主名
<code>api_ex.api.enable_tls</code>	布尔	<code>true</code>	是否启用 API 的 TLS 加密
<code>api_ex.api.tls_cert_path</code>	字符串	<code>"priv/cert/omnitouch.crt"</code>	API 的 TLS 证书路径

名前	型	値	型	説明
api_ex.api.tls_key_path	string	"priv/cert/omnitouch.pem"	string	API の TLS 証明書

例:

```

config :omniss7,
  start_http_server: true

config :api_ex,
  api: %{
    port: 8445,
    listen_ip: "0.0.0.0",
    product_name: "OmniSS7",
    title: "API - OmniSS7",
    hostname: "localhost",
    enable_tls: true,
    tls_cert_path: "priv/cert/omnitouch.crt",
    tls_key_path: "priv/cert/omnitouch.pem"
  }

```

API 例:

- REST API: [https://\[server-ip\]:8445/api/](https://[server-ip]:8445/api/)*
- Swagger UI: [http://\[server-ip\]:8080/swagger](http://[server-ip]:8080/swagger)
- Prometheus 例: [http://\[server-ip\]:8080/metrics](http://[server-ip]:8080/metrics)

配置

属性	类型	默认值	说明	备注
<code>logger.level</code>	String	<code>:debug</code>	日志级别	支持的级别有 <code>:warn</code> , <code>:error</code> , <code>:info</code> , <code>:debug</code>
<code>logger.backends</code>	List	<code>[:console]</code>	日志后端	支持的有 <code>SS7.Web.LoggerBackend</code> , <code>Web.LoggerBackend</code>
<code>logger.default_formatter.format</code>	String	-	日志格式	支持的有 <code>%{date} %{time} %{level} %message\n</code>
<code>logger.default_formatter.metadata</code>	List	<code>[:]</code>	日志元数据	支持的有 <code>:error_code</code> , <code>:file</code>
<code>logger.default_formatter.truncate</code>	Integer	<code>:infinity</code>	日志截断	支持的有 <code>:infinity</code>

配置

```
config :logger,
  level: :debug,
  backends: [:console, SS7.Web.LoggerBackend]
```

```
config :logger, :default_formatter,
  format: "[%{date} %{time} %{level} %message\n",
  metadata: [:error_code, :file],
  truncate: :infinity
```

Configuration

Mnesia Configuration

Parameter	Default	Value	Unit	Description
<code>mnesia_storage_type</code>		<code>:disc_copies</code>		Mnesia storage type <code>:disc_copies</code> (disk) <code>:ram_copies</code> (RAM)

Example:

```
config :omniss7,  
  mnesia_storage_type: :disc_copies # disk  
  # mnesia_storage_type: :ram_copies # RAM
```

Options:

- `:disc_copies` - disk storage (default)
- `:ram_copies` - RAM storage

Mnesia Settings:

- `m3ua_peer` - M3UA peer
- `m3ua_route` - M3UA route
- `m3ua_gt_route` - M3UA gateway route

Path: `Mnesia.{node_name}/`

Summary

Configuration summary

□□

□	□□	□□□□
MAP □□□□ 10 □	□□ MAP □□□ 10 □□□□	□□□□□
ISD □□□ 10 □	□□ ISD □□□ 10 □□□□	□□□□□

HTTP □□□

□	□□	□□□□
HTTP IP: 0.0.0.0	□□/Swagger □□□□□□□□□□	□□□□□
HTTP □□: 8080	□□/Swagger □□□□□ 8080 □□□	□□□□□

□□□□

□	□□	□□□□
□□□□□ 5 □□	SMSc □ 5 □□□□□□□□	□□□□□

Web UI □□□□

□□	□□
□□□□	5 □
□□□□	2 □

□□□□

□□ **HLR** □□

```
config :omniss7,  
  map_client_enabled: true,  
  hlr_mode_enabled: true,  
  smsc_mode_enabled: false,  
  
  hlr_api_base_url: "https://10.180.2.140:8443",  
  hlr_service_center_gt_address: "55512341111",  
  smsc_service_center_gt_address: "55512341112",  
  
  map_client_m3ua: %{  
    mode: "ASP",  
    callback: {MapClient, :handle_payload, []},  
    process_name: :hlr_client_asp,  
    local_ip: {10, 179, 4, 11},  
    local_port: 2905,  
    remote_ip: {10, 179, 4, 10},  
    remote_port: 2905,  
    routing_context: 1  
  }
```

☐☐ SMSG ☐☐

```
config :omniss7,  
  map_client_enabled: true,  
  hlr_mode_enabled: false,  
  smsc_mode_enabled: true,  
  
  smsc_api_base_url: "https://10.179.3.219:8443",  
  smsc_name: "ipsmgw",  
  smsc_service_center_gt_address: "55512341112",  
  
  auto_flush_enabled: true,  
  auto_flush_interval: 10_000,  
  auto_flush_dest_smsc: "ipsmgw",  
  auto_flush_tps: 10,  
  
  map_client_m3ua: %{  
    mode: "ASP",  
    callback: {MapClient, :handle_payload, []},  
    process_name: :stp_client_asp,  
    local_ip: {10, 179, 4, 12},  
    local_port: 2905,  
    remote_ip: {10, 179, 4, 10},  
    remote_port: 2905,  
    routing_context: 1  
  }
```

STP

```
config :omniss7,  
  map_client_enabled: true,  
  hlr_mode_enabled: false,  
  smsc_mode_enabled: false,  
  
  enable_gt_routing: true,  
  mnesia_storage_type: :disc_copies,  
  
  sctp_handler: %{  
    enabled: true,  
    local_ip: {10, 179, 4, 10},  
    local_port: 2905,  
    point_code: 100  
  },  
  
  map_client_m3ua: %{  
    mode: "ASP",  
    callback: {MapClient, :handle_payload, []},  
    process_name: :stp_client_asp,  
    local_ip: {10, 179, 4, 10},  
    local_port: 2906,  
    remote_ip: {10, 179, 4, 11},  
    remote_port: 2905,  
    routing_context: 1  
  }  
}
```

□□

75+

□□□:

- □□□□□ 5 □□□
- HLR □□□ 17 □□□
- SMSc □□□ 8 □□□

- STP 节点 5+ 节点名称 m3ua_peers|m3ua_routes|m3ua_gt_routes 节点
- CAMEL GW 节点 14 节点
- 节点 NAT 2 节点
- M3UA 节点 8 节点
- 节点名称 Web API 节点 23 节点
- 节点 1 节点

节点名称:

HLR 节点:

- hlr_api_base_url
- hlr_service_center_gt_address
- smsc_service_center_gt_address
- 节点 map_client_m3ua.* 节点 8 节点

SMSc 节点:

- smsc_api_base_url
- smsc_service_center_gt_address
- auto_flush_dest_smsc 节点名称
- 节点 map_client_m3ua.* 节点 8 节点

STP 节点:

- sctp_handler.point_code 节点名称 SCTP 节点名称
- sctp_handler.local_ip
- sctp_handler.local_port

CAMEL GW 节点:

- cgrates_url 节点名称 CGrateS
- 节点 cap_client_m3ua.* 节点 8 节点

节点名称:

- license_client.license_server_api_urls
- license_client.licensee

□□□□

- **HLR** □□ - HLR □□□□
- **SMSc** □□ - SMSc □□□□
- **STP** □□ - STP □□□□
- **API** □□ - REST API □□
- **USSD** □□□□ - USSD □□□□ HTTP □□□□
- **Web UI** □□ - Web □□□□

GT NAT

GT

GT NAT OmniSS7 GT GT
GT GT
GT GT

GT NAT

GT NAT

GT

- called_prefix calling_prefix
-
- weight =
- -
 -
 -

GT

GT

- weight =
- response_gt GT

GT

- calling_prefix GT
- called_prefix GT

項目	型別	範囲	説明
<code>gt_nat_enabled</code>	bool	bool	GT NAT を有効にするかどうか
<code>gt_nat_rules</code>	array	array of array	GT NAT のルール

例

設定例

```
%{
  calling_prefix: "8772",      # 発信元 GT
  called_prefix: "555",      # 着信元 GT
  weight: 10,                # 優先度 =
  response_gt: "55512341112" # 応答 GT
}
```

説明

- `calling_prefix`** 発信元 GT
 - `String.starts_with?/2` を参照
 - 空文字列 "" は nil として扱われる
 - 必須
- `called_prefix`** 着信元 GT
 - `String.starts_with?/2` を参照
 - 空文字列 "" は nil として扱われる
 - 必須
- `weight`**
 - 優先度 =
 - 0
 - 必須

- **response_gt**
 - E.164
 - GT

calling_prefix **called_prefix**

1.
 - **calling_prefix** GT
 - **called_prefix** GT
 -
 -

2.
 - -
 - - =

- 3.

```

# []
gt_nat_rules: [
  # [] 1[] - []
  %{calling_prefix: "8772", called_prefix: "555", weight: 1,
response_gt: "11111"},

  # [] 10[] - []
  %{calling_prefix: "8772", weight: 10, response_gt: "22222"}, #
[]
  %{called_prefix: "555", weight: 10, response_gt: "33333"}, #
[]

  # [] 100[] - []
  %{weight: 100, response_gt: "44444"} # []
]

# []
# []"877234567"[]"555123" -> "111111"[] 1[]
# []"877234567"[]"999999" -> "222222"[] 10[]
# []"999999999"[]"555123" -> "333333"[] 10[]
# []"999999999"[]"888888" -> "444444"[] 100[]

```

[]

[] **1**[][]

[][] SMS[] GT []

```

config :omniss7,
  gt_nat_enabled: true,

  # SMS Sc GT NAT
  smsc_service_center_gt_address: "5551000",

  # GT NAT
  gt_nat_rules: [
    # A 4412 GT 5551001
    %{calling_prefix: "4412", weight: 10, response_gt: "5551001"},

    # B 4413 GT 5551002
    %{calling_prefix: "4413", weight: 10, response_gt: "5551002"},

    # SMS Sc GT
    %{weight: 100, response_gt: "5551000"}
  ]

```

GT NAT

```

44121234567 SRI-for-SM
  GT5551001 A
  GT44121234567 A GT

GT NAT
"44121234567" "4412"
  response_gt "5551001"

SRI-for-SM 44121234567
  GT44121234567
  GT5551001 NAT'd
  networkNode-Number5551001 MAP

```

2 GT HLR

HLR GT

```

config :omniss7,
  gt_nat_enabled: true,
  hlr_service_center_gt_address: "555000", # HLR GT

  gt_nat_rules: [
    # VLR 5551
    %{calling_prefix: "5551", weight: 10, response_gt: "555100"},

    # VLR 5552
    %{calling_prefix: "5552", weight: 10, response_gt: "555200"},

    # VLR 5553
    %{calling_prefix: "5553", weight: 10, response_gt: "555300"},

    #
    %{weight: 100, response_gt: "555000"}
  ]

```

3

GT GT

```

config :omniss7,
  gt_nat_enabled: true,
  hlr_service_center_gt_address: "123456789", # GT

  gt_nat_rules: [
    #
    %{calling_prefix: "555", weight: 10, response_gt:
"987654321"}, # GT
    %{calling_prefix: "666", weight: 10, response_gt:
"987654321"}, # GT

    # GT
    %{weight: 100, response_gt: "123456789"} # GT
  ]

```

00 4 0000000000000000

00000000 GT 00000000000000000000 GT 000000 GT

```
config :omniss7,  
  gt_nat_enabled: true,  
  
  gt_nat_rules: [  
    # 00000000 SMS GT05551xxx00000000 GT  
    %{called_prefix: "5551", weight: 10, response_gt: "555100"},  
  
    # 0000000000 GT05552xxx00000000 GT  
    %{called_prefix: "5552", weight: 10, response_gt: "555200"},  
  
    # 0000000000 GT05553xxx00000000 GT  
    %{called_prefix: "5553", weight: 10, response_gt: "555300"},  
  
    # 00000  
    %{weight: 100, response_gt: "555000"}  
  ]
```

000000

```
00000000 GT05551000000 SMS GT  
00 GT044123456700000000  
  
GT NAT 000  
00 GT "555100" 0000 "5551"  
000 response_gt "555100"  
  
00000000 GT055510000000000000
```

00 5 0000000 + 00000000000000

00000000000000000000 GT00000000000000

```

config :omniss7,
  gt_nat_enabled: true,

  gt_nat_rules: [
    # 0000 A 0000 SMS GT - 00000000 10
    %{calling_prefix: "4412", called_prefix: "5551", weight: 1,
response_gt: "555101"},

    # 0000 B 0000 SMS GT - 00000000 10
    %{calling_prefix: "4413", called_prefix: "5551", weight: 1,
response_gt: "555102"},

    # 00000000 SMS GT - 0000000000 100
    %{called_prefix: "5551", weight: 10, response_gt: "555100"},

    # 0000 A 0000 GT - 00000000 100
    %{calling_prefix: "4412", weight: 10, response_gt: "555200"},

    # 0000 - 00000000 1000
    %{weight: 100, response_gt: "555000"}
  ]

```

00000

```

# 0000 A 00 SMS GT
000"441234567"0000"555100"
→ 0000 1 00000000→ "555101"

# 0000 A 0000 GT
000"441234567"0000"555200"
→ 0000 10 00000000→ "555200"

# 00000000 SMS GT
000"999999999"0000"555100"
→ 0000 10 00000000→ "555100"

# 0000000000 GT
000"999999999"0000"555200"
→ 0000 100 000 → "555000"

```

□□□□

GT NAT □□□ OmniSS7 □□□□□□□□

HLR □□

GT NAT □□□

- UpdateLocation □□□□□□□□ HLR GT□
- InsertSubscriberData □□□□□□□□□□ HLR GT□
- SendAuthenticationInfo □□
- □□□□□□

□□ HLR □□□□□□□□□□□□ HLR □□□□□

□□□

```
config :omniss7,  
  hlr_mode_enabled: true,  
  hlr_service_center_gt_address: "5551234567", # □□ HLR GT  
  
  gt_nat_enabled: true,  
  gt_nat_rules: [  
    %{calling_prefix: "331", weight: 10, response_gt:  
"5551234568"}, # □□  
    %{calling_prefix: "44", weight: 10, response_gt:  
"5551234569"}, # □□  
    %{weight: 100, response_gt: "5551234567"} # □□□□  
  ]
```

SMSc □□

GT NAT □□□

- SRI-for-SM □□□ networkNode-Number □□□ - □□ SRI-for-SM □□□□
- MT-ForwardSM □□

□□ SMSc □□□□□□□□□□□□ SMSc □□□□□

□□□

```
config :omniss7,  
  smsc_mode_enabled: true,  
  smsc_service_center_gt_address: "5559999", # □□ SSMSc GT  
  
  gt_nat_enabled: true,  
  gt_nat_rules: [  
    {%calling_prefix: "1", weight: 10, response_gt: "5559991"},  
# □□  
    {%calling_prefix: "44", weight: 10, response_gt: "5559992"},  
# □□  
    {%calling_prefix: "86", weight: 10, response_gt: "5559993"},  
# □□  
    {%weight: 100, response_gt: "5559999"} # □□□□□  
  ]
```

CAMEL □□□□

GT NAT □□□

- □□ SCCP □□□□gsmSCF GT □□□□□□
- CAMEL/CAP □□□□□InitialDP□EventReportBCSM □□
- RequestReportBCSMEvent □□
- ApplyCharging ◆◆◆◆□
- Continue □□

□□□

```

config :omniss7,
  camelgw_mode_enabled: true,
  camel_gsmSCF_gt_address: "55512341112", # [] gsmSCF GT

  gt_nat_enabled: true,
  gt_nat_rules: [
    %{calling_prefix: "555", weight: 10, response_gt:
"55512341111"}, # [] A
    %{calling_prefix: "666", weight: 10, response_gt:
"55512311555"}, # [] B
    %{weight: 100, response_gt: "55512341112"} # []
  ]

```

[] [] gsmSCF [] gsmSSF [] gsmSCF GT []
 []GT NAT [] gsmSSF [] GT

[][][][][]

[] **GT NAT** []

GT NAT []

```

# []
[info] GT NAT [SRI-for-SM []]: [] GT 877234567 -> [] GT
55512341112
[info] GT NAT [UpdateLocation ISD]: [] GT 331234567 -> [] GT
55512341111
[info] GT NAT [MAP BEGIN []]: [] GT 441234567 -> [] GT 55512311555

```

[][] NAT []

- "SRI-for-SM []" - [] SRI-for-SM []
- "UpdateLocation ISD" - [] InsertSubscriberData []
- "UpdateLocation END" - [] UpdateLocation END []
- "MAP BEGIN []" - [] MAP BEGIN []
- "ISD ACK" - ISD []

- "HLR 0000" - HLR 00000
- "CAMEL 00" - CAMEL/CAP 00000gsmSCF0

00

00000000 GT NAT 000

```
# 00 GT NAT 00
iex> GtNat.validate_config()
{:ok, [
  %{calling_prefix: "8772", weight: 10, response_gt:
"55512341112"},
  %{calling_prefix: "8773", weight: 10, response_gt:
"55512341111"}
]}

# 00000000
iex> GtNat.enabled?()
true

# 00000000
iex> GtNat.get_rules()
[
  %{calling_prefix: "8772", weight: 10, response_gt:
"55512341112"},
  %{calling_prefix: "8773", weight: 10, response_gt:
"55512341111"}
]
```

00 GT NAT

00000000 GT NAT 000

```

# 877234567 GT 55512341112called_gt [] nil[]
iex> GtNat.translate_response_gt("877234567", nil, "default_gt")
"55512341112"

# 877234567 GT 555123
iex> GtNat.translate_response_gt("877234567", "555123",
"default_gt")
"55512341112"

# 877234567 nil [] GT[]
iex> GtNat.translate_response_gt_with_logging("877234567", nil,
"default_gt", "test")
# []GT NAT [test]: [] GT 877234567 -> [] GT 55512341112
"55512341112"

# 877234567 GT[]
iex> GtNat.translate_response_gt_with_logging("877234567",
"555123", "default_gt", "test")
# []GT NAT [test]: [] GT 877234567, [] GT 555123 -> [] GT
55512341112
"55512341112"

# 9999999999
iex> GtNat.translate_response_gt("999999999", "888888",
"default_gt")
"default_gt"

```

□□□□

□□□**GT NAT** □□□□

□□ **1**□□□□□□

```

iex> Application.get_env(:omniss7, :gt_nat_enabled)
true # [] true

```

□□ **2**□□□□□□□□

```
iex> Application.get_env(:omniss7, :gt_nat_rules)
[%{calling_prefix: "8772", response_gt: "55512341112"}, ...] # []
[]
```

3 GT NAT

GT

GT

```
#
gt_nat_rules: [
  %{weight: 1, response_gt: "111111"}, #
  %{calling_prefix: "8772", weight: 10, response_gt: "222222"} #
]

#
gt_nat_rules: [
  %{calling_prefix: "8772", weight: 10, response_gt: "222222"}, #
  %{weight: 100, response_gt: "111111"} #
]
```

GT NAT

NAT'd GT

- SCCP GT
- SRI-for-SM networkNode-Number
- UpdateLocation ISD HLR GT
- UpdateLocation END

- ISD
- MAP BEGIN

GT NAT

GT NAT $O(n)$ n

- 100
-
-

- 10 $< 1\mu s$
- 50 $< 5\mu s$
- 100 $< 10\mu s$

100

- 10 $\approx 1\text{ KB}$
- 100 $\approx 10\text{ KB}$

□□□□

1. □□□□□□□□□□

```
gt_nat_rules: [  
  {%calling_prefix: "8772", weight: 10, response_gt: "111111"},  
  {%calling_prefix: "8773", weight: 10, response_gt: "222222"},  
  {%weight: 100, response_gt: "default_gt"} # □□□□□□□□□□  
]
```

2. □□□□□□□□□□

```
# □□□□□□□□□□□□□□  
{%calling_prefix: "331", weight: 10, response_gt: "..."} # □□  
{%calling_prefix: "44", weight: 10, response_gt: "..."} # □□  
  
# □□□□□□□□□□□□□□  
{%calling_prefix: "3", weight: 5, response_gt: "..."} # □□□□  
{%calling_prefix: "331", weight: 100, response_gt: "..."} # □□□□□  
□□□□□□□□□□
```

3. □□□□□□

```
gt_nat_rules: [  
  # □□□□ XYZ - □□□□□GT □□□4412xxxxxxx□  
  # □□ 10□□□□□□□□□□□□  
  {%calling_prefix: "4412", weight: 10, response_gt: "5551001"},  
  
  # □□□□ ABC - □□□□□GT □□□33123xxxxxxx□  
  # □□ 10□□□□□□□□□□□□  
  {%calling_prefix: "33123", weight: 10, response_gt: "5551002"}  
]
```

4. 配置

```
# 配置 iex 规则
iex> GtNat.translate_response_gt("44121234567", nil, "default")
"5551001" # 配置

# 配置 GT 规则
iex> GtNat.translate_response_gt("44121234567", "555123",
"default")
"5551001" # 配置
```

5. 配置

INFO 配置 GT NAT 配置

配置

STP 配置

GT NAT 配置 STP 配置 STP 配置 GT 配置 GT NAT 配置

配置 STP 配置 STP 配置

CAMEL 配置

GT NAT 配置 CAMEL/CAP 配置

SCCP 配置

- 配置 CAMEL 配置 GT
- 配置 gsmSSF GT 配置

配置

- `camel_gsmscf_gt_address` - 配置 gsmSCF GT 配置
- 配置 GT
- GT NAT 配置

3. 確認

```
gt_nat_enabled: true # true
```

4. 確認

```
tail -f log/omniss7.log | grep "GT NAT"
```

5. 確認

- 確認
- 確認 24 時間
- 確認 `gt_nat_enabled: false`

□□

確認

- 確認 "GT NAT" 確認
- 確認 `GtNat.validate_config()` 確認
- 確認
- 確認 OmniSS7 確認

□□

- HLR 確認 - HLR 確認
- SMSC 確認 - SMSc 確認
- STP 確認 - STP 確認
- 確認 - 確認

HLR

←

OmniSS7 (HLR/HSS) OmniHSS

OmniHSS

OmniSS7 HLR SS7 OmniHSS HSS

- OmniSS7 (HLR) SS7/MAP SCCP
- OmniHSS (HSS)

OmniHSS

OmniHSS

- IMSI MSISDN IMSI eSIM
- Milenage 3G/4G/5G COMP128 2G
- CS PS
- CAMEL
- API RESTful HTTP API CRM
-

OmniHSS OmniSS7 HTTPS API OmniHSS MAP UpdateLocation SendAuthenticationInfo SendRoutingInfo

OmniSS7 HLR OmniHSS OmniSS7 SS7/MAP OmniHSS

IMSI

OmniHSS IMSI MSISDN IMSI

- IMSI 12345678901234567890
- **eSIM** 123456789012345678901234567890
- MSISDN 12345678901234567890
- **SIM** 123456789012345678901234567890
- IMSI 12345678901234567890

IMSIs

- IMSI 123456789012345678901234567890
- IMSI 123456789012345678901234567890
- IMSI 123456789012345678901234567890
- OmniSS7 IMSI 123456789012345678901234567890
- IMSI 123456789012345678901234567890

IMSIs

```

MSISDN: +1-555-123-4567
├ IMSI 1: 310260123456789 (Milenage)
├ IMSI 2: 208011234567890 (Milenage)
└ IMSI 3: 440201234567891 (COMP128)
  
```

IMSIs 123456789012345678901234567890 OmniHSS IMSI 123456789012345678901234567890

□□

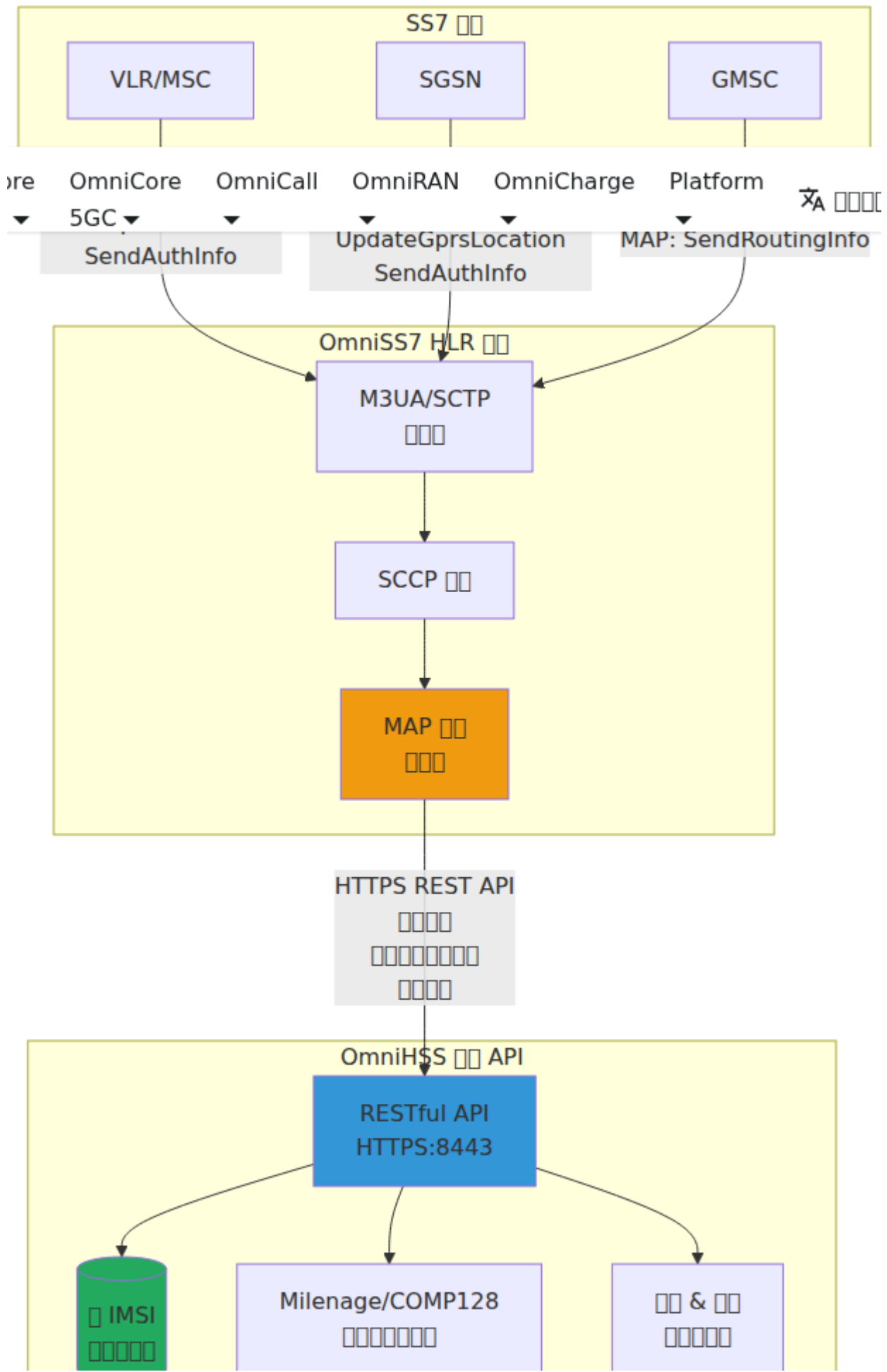
1. OmniHSS □□
2. □ IMSI □□
3. □□□ HLR □□□
4. □□ HLR □□
5. □□□□□
6. □□□□□□
7. □□□□
8. CAMEL □□
9. □□□□□□
10. HLR □□
 - □□□□□□
 - SendRoutingInfo (SRI)
 - UpdateLocation / ISD
 - SendRoutingInfoForSM
 - □□□□□□

HLR

HLR OmniSS7

-
-
- VLR
- SMS

HLR □□





HLR

OmniSS7 HLR HLR

HLR

OmniSS7 `config/runtime.exs` HLR

1. `config/runtime.exs`
2. 53-174
 - 1 STP 53-85
 - 2 HLR 87-123
 - 3 SMS Sc 125-174
3. #
4. HLR 87-123 #
- 5.
6. `iex -S mix`

HLR

HLR

```
config :omniss7,
  # 配置 - 配置 HLR 配置
  map_client_enabled: true,
  hlr_mode_enabled: true,
  smsc_mode_enabled: false,

  # OmniHSS 配置 API 配置
  hlr_api_base_url: "https://10.180.2.140:8443",

  # HLR 配置 GT 配置 SMS 配置
  hlr_service_center_gt_address: "1234567890",

  # MSISDN ↔ IMSI 配置
  # 配置MSISDN ↔ IMSI 配置
  hlr_imsi_plmn_prefix: "50557",
  hlr_msisdn_country_code: "61",
  hlr_msisdn_nsn_offset: 0,
  hlr_msisdn_nsn_length: 9,

  # InsertSubscriberData 配置
  # 配置:packetAndCircuit配置:packetOnly配置:circuitOnly
  isd_network_access_mode: :packetAndCircuit,

  # 配置 ISD #2配置
  isd_send_ss_data: true,

  # 配置 ISD #3配置
  isd_send_call_barring: true,

  # CAMEL 配置 SendRoutingInfo 配置
  # CAMEL 配置
  camel_service_key: 11_110,

  # CAMEL 配置
  # 配置:termAttemptAuthorized配置:tBusy配置:tNoAnswer配置:tAnswer
  camel_trigger_detection_point: :termAttemptAuthorized,

  # 配置 VLR 配置
  # 配置“配置”配置 VLR 配置
  # 配置 VLR 配置 SRI 配置
  # 配置 PRN 配置 MSRN
  home_vlr_prefixes: ["123456"],
```

```
# M3UA
# ASP MAP UpdateLocation SendAuthInfo
map_client_m3ua: %{
  mode: "ASP",
  callback: {MapClient, :handle_payload, []},
  process_name: :hlr_client_asp,
  # HLR
  local_ip: {10, 179, 4, 11},
  local_port: 2905,
  # STP
  remote_ip: {10, 179, 4, 10},
  remote_port: 2905,
  routing_context: 1
}
```

□□□□□□□□□□

□□□□□□□□□□□□□□□□ □□□□

参数	数据类型	默认值	说明
hlr_api_base_url	字符串	空字符串	Omni
hlr_service_center_gt_address	字符串	空字符串	核心网 HLR地址
smc_service_center_gt_address	字符串	空字符串	短信中心 SMS
hlr_smc_alert_gts	字符串	[]	短信中心 报警网关 SMS
hlr_alert_location_expiry_seconds	整数	172800	短信报警 有效期 秒
hlr_imsi_plmn_prefix	字符串	"50557"	MSISDN PLMN ↔ IMSI
hlr_msisdn_country_code	字符串	"61"	核心网 HLR IMSI
hlr_msisdn_nsn_offset	整数	0	核心网 HLR IMSI
hlr_msisdn_nsn_length	整数	9	核心网 MSISDN

属性名	数据类型	默认值	说明
isd_network_access_mode	枚举	:packetAndCircuit	接入模式
isd_send_ss_data	布尔	true	是否发送SS数据
isd_send_call_barring	布尔	true	是否发送呼叫限制
camel_service_key	枚举	11_110	发送CAM的密钥
camel_trigger_detection_point	枚举	:termAttemptAuthorized	CAM的触发检测点
home_vlr_prefixes	列表	["5551231"]	归属VLR前缀列表
local_ip	字符串		本地IP地址
local_port	整数	2905	本地端口
remote_ip	字符串		远程IP地址
remote_port	整数	2905	远程端口
routing_context	枚举	1	M3U路由上下文

HLR

hlr_mode_enabled: true Web UI

- SS7 -
- SS7 - MAP
- M3UA -
- HLR - HLR API + ← HLR
-
-

SMSc

- hlr_service_center_gt_address
- OmniHSS OmniHSS API hlr_api_base_url
- API OmniHSS API 5
- MAP MAP SRI UpdateLocation SendAuthInfo 10
- ISD UpdateLocation InsertSubscriberData (ISD) 10
- STP M3UA MAP
-
- Web UI Web UI
- API REST API Swagger UI API

OmniHSS OmniSS7 RESTful API

OmniHSS

OmniHSS

- IMSI
- Ki/OPc Milenage COMP128
- QoS
- VLR/MSC SGSN/GGSN
- CAMEL gsmSCF
- CLIP/CLIR
- /

OmniHSS Milenage COMP128 OmniSS7 **sendAuthenticationInfo** MAP

1. OmniSS7 MAP IMSI
2. OmniSS7 OmniHSS API
3. OmniHSS Ki OPc
4. OmniHSS RAND XRES CK IK AUTN
5. OmniSS7 MAP VLR/SGSN

OmniHSS API

OmniSS7 HTTPS REST API OmniHSS

```
config :omniss7,  
  hlr_api_base_url: "https://omnihss-server:8443"
```

OmniSS7 SS7 MAP OmniHSS

- IMSI MSISDN
- Ki/OPc
- UpdateLocation
-

□□□□

□□□□□□

□□□□ **updateLocation** MAP □□□□OmniSS7 □ OmniHSS □□□□□□ VLR □□□□□

1. **UpdateLocation** □□□□□□□□□□IMSI□□ VLR GT□□ MSC GT□
2. □□ **OmniHSS** □□□□□□□□□□□□
3. □ **OmniHSS** □□□□□□□□□□□ VLR/MSC □□
4. □□ **InsertSubscriberData (ISD)** □□□□□ VLR □□□□□
5. □□ **UpdateLocation** □□ □ VLR□□□□□ `hlr_service_center_gt_address` □
HLR GT□
6. □□ **alertServiceCenter** □□□□ SSMSc GT□□□□ `hlr_smsc_alert_gts` □□□□

□□□□ `hlr_service_center_gt_address` □□□□□□□□ UpdateLocation □□□□□□□ HLR □□□□
□□□□□□ VLR/MSC □□□□□□□□□□□□ HLR□

□□□□□□□□

□□□□ UpdateLocation □□□□HLR □□□□□□ **alertServiceCenter**□MAP □□□ 64□□□□□□□□
SSMc □□□□□□□□□□□□□□□□□□□□□□□□□□ **SSMc** □□□□□□□□□□□□□□

□□

□□□□□□□□ SSMSc □□□□□□□□

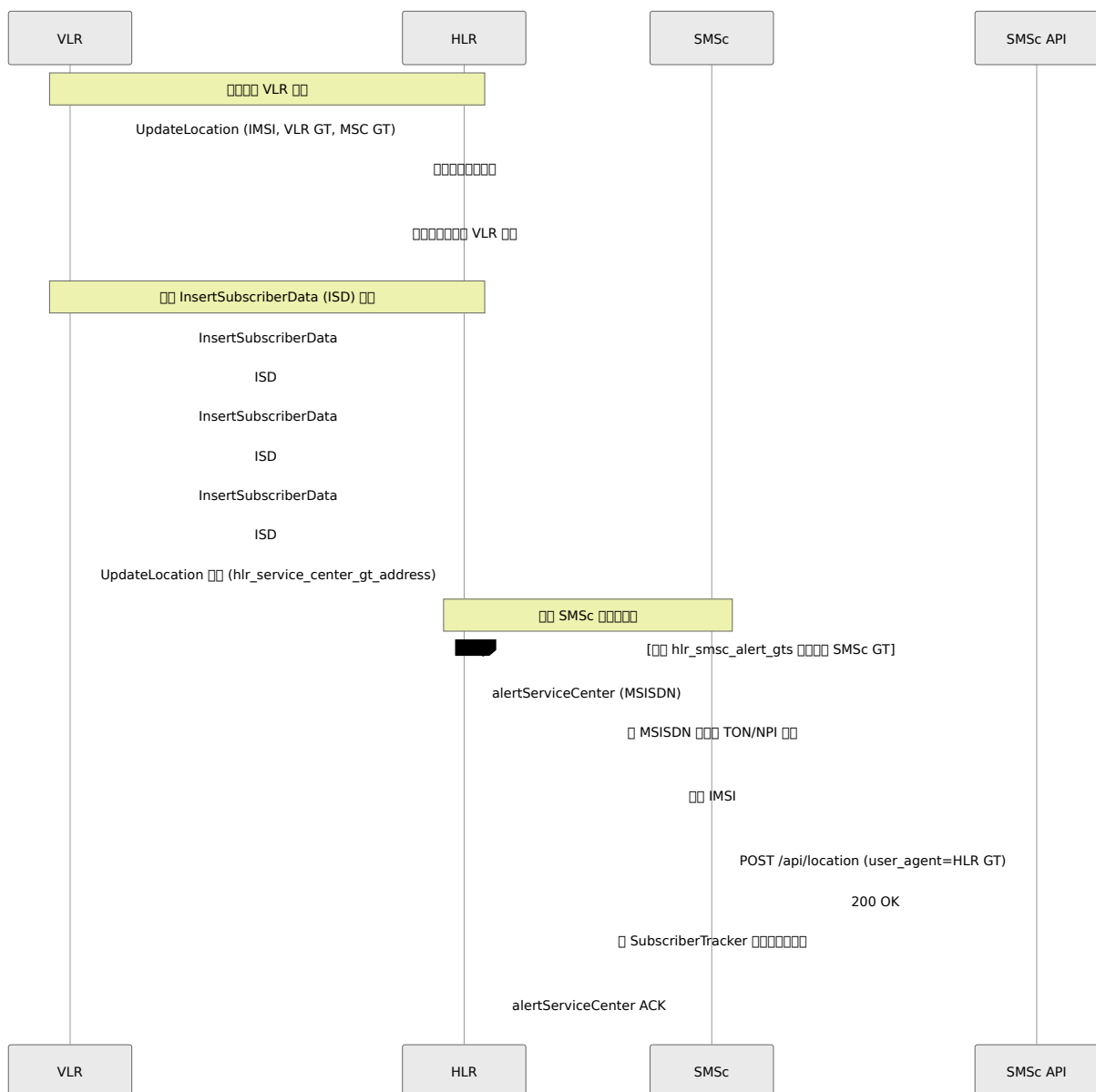
```

config :omniss7,
  # UpdateLocation alertServiceCenter SSMSc GT
  hlr_smsc_alert_gts: [
    "15559876543",
    "15559876544"
  ],

  # SSMSc alertServiceCenter 48
  hlr_alert_location_expiry_seconds: 172800

```

□□□



□□

UpdateLocation

1. HLR `hlr_smsc_alert_gts` SSMSc GT `alertServiceCenter`
2. MSISDN
3. HLR `hlr_service_center_gt_address` GT
4. SCCP SSN=6 HLR SSN=8 SSMSc

SSMSc

- **MSISDN** **TON/NPI** "19123123213" → "123123213"
- POST `/api/location`
- **API** `user_agent` **HLR GT** HLR
- `hlr_alert_location_expiry_seconds`
- SSMSc

Web UI `alertServiceCenter`

1. " "
2. " "
3. MSISDN SSMSc GT HLR GT
 - SSMSc GT `hlr_smsc_alert_gts`
 - HLR GT `hlr_service_center_gt_address`
4. " alertServiceCenter"

SSMSc UpdateLocation `phx-blur`

InsertSubscriberData (ISD)

UpdateLocation HLR **InsertSubscriberData (ISD)** VLR ISD

ISD

ISD

HLR 000000 3 0000 ISD 0000

1. ISD #1000000 - 00000000

- IMSI
- MSISDN
- 0000
- 000000serviceGranted
- 000000
- 000000
- 000000

2. ISD #20000 - 000000SS0000

- 000000000000000000000000
- 0000
- 0000
- 0000
- 00000000

3. ISD #30000 - 00000000

- 00000000BAOC
- 00000000BOIC
- 000000

0000

```
# InsertSubscriberData 00
# 00000000:packetAndCircuit0:packetOnly 0 :circuitOnly
isd_network_access_mode: :packetAndCircuit,

# 00 ISD #2000000000
isd_send_ss_data: true,

# 00 ISD #3000000000
isd_send_call_barring: true,
```

□□□□□□

isd_network_access_mode □□□□□□□□□□□□□□□□

□	□□	□□
:packetAndCircuit	□□□□□□□□□□GPRS/LTE□□□□□□□□□□□□	□□ - □□□□□□□
:packetOnly	□□□□□□□□□□/LTE□	□□□□□□□□□□
:circuitOnly	□□□□□□□□□□/SMS□	□□□□□□□□□□

□□ **ISD** □□

□□□□□□□□□□□□□□□□ ISD □□□□

□□□□ **ISD**□□□□ - □□□□□□□□

```
isd_send_ss_data: true,  
isd_send_call_barring: true,
```

□□□□□□□□□□□□□□□□

```
isd_send_ss_data: false,  
isd_send_call_barring: false,
```

□□□□ + □□□□□□□□□□□□□□

```
isd_send_ss_data: true,  
isd_send_call_barring: false,
```

ISD □□□□

□□□□ UpdateLocation □□

```

VLR → HLR: UpdateLocation ( )
HLR → VLR: InsertSubscriberData #1 ( ) - 
VLR → HLR: ISD #1 ACK ( )
HLR → VLR: InsertSubscriberData #2 ( ) - SS [ ]
VLR → HLR: ISD #2 ACK ( )
HLR → VLR: InsertSubscriberData #3 ( ) - [ ]
VLR → HLR: ISD #3 ACK ( )
HLR → VLR: UpdateLocation ( )

```

`isd_send_ss_data` `isd_send_call_barring` `false` ISD
 UpdateLocation

- `:packetAndCircuit` ISD
- `/M2M` `:packetOnly` SS
- VLR - `isd_send_ss_data`
- ISD

CAMEL

CAMEL `SendRoutingInfo`

GMSC MSC `SendRoutingInfo` HLR GMSC CAMEL

CAMEL

CAMEL

CAMEL GSM/UMTS

-
-
- VPN

- 0000
- 0000000000000000
- 00000000

0000

```
# CAMEL 00000 SendRoutingInfo 000
# CAMEL 000000000000
camel_service_key: 11_110,

# CAMEL 00000
# 000:termAttemptAuthorized:tBusy:tNoAnswer:tAnswer
camel_trigger_detection_point: :termAttemptAuthorized,
```

0000

camel_service_key 000 gsmSCF0000000000000000 CAMEL 00000000000000000000
0000

0000	0000
11_110	0000000000000000
100	00000000
200	??000000000000
300	00000000VPN
000	00000000

00000

```
# 00000000000000000000
camel_service_key: 11_110,
```

```
# 00 VPN 00
camel_service_key: 300,
```

000000

camel_trigger_detection_point 00000000000000000000 CAMEL 000

000	00	0000
:termAttemptAuthorized	0000000000000000	000000000000
:tBusy	0000	000000
:tNoAnswer	000000	00000000
:tAnswer	0000	0000000000

000000

000000000000000000000000

```
camel_trigger_detection_point: :termAttemptAuthorized,
```

000000000000000000000000

```
camel_trigger_detection_point: :tBusy,
```

000000000000000000000000

```
camel_trigger_detection_point: :tAnswer,
```

0 CAMEL 0 SRI 00

SendRoutingInfo CAMEL

```
GMSC → HLR: SendRoutingInfo ( )
HLR → GMSC: SRI ( )
- IMSI
- VLR
-
- CAMEL
* 11_110
* gsmSCF <>
* termAttemptAuthorized
* continueCall
```

GMSC gsmSCF CAMEL

- gsmSCF
- :termAttemptAuthorized
- 11_110
- defaultCallHandling: :continueCall gsmSCF

VLR VLR

HLR SendRoutingInfo SRI " " VLR VLR

VLR

- VLR CAMEL SRI
- VLR PRN MSRN SRI


```
GMSC → HLR: SendRoutingInfo (MSISDN: "1234567890")
HLR  API 
HLR  VLR "49170123456"
HLR  VLR "555123"  → 
HLR → MSC: ProvideRoamingNumber (PRN):
  - MSISDN: "1234567890"
  - IMSI: "999999876543210"
  - MSC  : "49170123456"
  - GMSC  : "5551234501"
MSC → HLR: PRN  MSRN: "49170999888777"
HLR → GMSC: SRI 
  - IMSI
  - VLR  : "49170123456"
  -     (MSRN): "49170999888777"
```

API

SRI

```
%{
  imsi: "999999876543210",
  extendedRoutingInfo: {
    :camelRoutingInfo, %{
      gsmcCamelSubscriptionInfo: %{
        "t-CSI": %{
          serviceKey: 11_110,
          "gsmSCF-Address": "5551234501",
          defaultCallHandling: :continueCall,
          "t-BcsmTriggerDetectionPoint": :termAttemptAuthorized
        }
      }
    }
  },
  subscriberInfo: %{
    locationInformation: %{"vlr-number": "5551234567"},
    subscriberState: {:notProvidedFromVLR, :NULL}
  }
}
```

SRI

```

%{
  imsi: "999999876543210",
  extendedRoutingInfo: {
    :routingInfo, %{
      roamingNumber: "49170999888777" # PRN MSRN
    }
  },
  subscriberInfo: %{
    locationInformation: %{"vlr-number": "49170123456"},
    subscriberState: {:notProvidedFromVLR, :NULL}
  }
}

```

PRN

PRN

MSC/VLR PRN

MSISDN	SRI	
IMSI	HLR API	IMSI
MSC	HLR API	MSC <code>serving_msc</code>
GMSC	SRI	SRI GMSC
CAMEL		GMSC CAMEL

PRN

HLR PRN

- **MSRN**

□□□□

- □□ PRN □□ → □ SRI □□□□□□ 27□□□□□□
- □□ PRN □□ → □ SRI □□□□□□ 27□□□□□□
- □□□□□□ MSRN → □ SRI □□□□□□ 27□□□□□□

□□□□

□□□□□□□□

```
# □□ VLR □□□ "555123" □□□□□□  
home_vlr_prefixes: ["555123"],
```

- VLR 5551234567 → □□□CAMEL □□□
- VLR 5551235001 → □□□CAMEL □□□
- VLR 49170123456 → □□□PRN + MSRN □□□

□□□□□□

```
# □□□□□□□□□□□□  
home_vlr_prefixes: ["555123", "555124", "555125"],
```

- VLR 5551234567 → □□□□□ 1□
- VLR 5552341234 → □□□□□ 2□
- VLR 5553411111 → □□□□□ 3□
- VLR 44201234567 → □□□□□□

□□□□

□□□□ PRN □□□□□□□□□□□□□□ VLR □□□□□□

```
# □□ VLR □□□□□□□□□□ PRN □□□□  
home_vlr_prefixes: [],
```

□□□□

- □□□□□□□□□□□□□□□□ VLR□□□□□□□□ + □□□□□
- □□□□□□□□□□□□ VLR □□□□□□□□□□□□
- □□□□□□□□□□□□□□□□ PRN
- □□□□□□□□□□□□□□□□□□□□
- □□□□□ PRN □□□□□□□□□□□□□□□□□□

□□□□

□□□□□□□□□□□□

- □□□□□□ `home_vlr_prefixes` □□□□ VLR □□□□□
- □□□□□□□□□□□□ VLR □□□□□□□□□□

□□□PRN □□□□

- □□□□□□□□□□□□ MSC/VLR □□□□□□□□
- □□□□□□□□□□□□ MSC □□□ M3UA/SCCP □□

□□□SRI □□□□ MSRN □□

- □□□□□□□□□□□□ PRN □□□□□□□□□□□□
- □□□□□□□□ PRN □□□□□□□□□□□□ `extract_msrn_from_prn/1`

HLR □□

□□□ MAP □□

- `updateLocation`□□□□□ 2□ - □□ VLR □□
- `sendAuthenticationInfo`□□□□□ 56□ - □□□□□□□□
- `sendRoutingInfo`□□□□□ 22□ - □□□ CAMEL □□□□□ MSRN
- `sendRoutingInfoForSM`□□□□□ 45□ - □ SMS □□ MSC GT
- `cancelLocation`□□□□□ 3□ - □□ VLR □□

- `insertSubscriberData` 7 -

HLR

SendRoutingInfo (SRI)

HLR VLR

CAMEL

VLR `home_vlr_prefixes`

項目	API	説明
IMSI	OmniHSS API	OmniHSS から取得した IMSI
VLR 番号	OmniHSS API	VLR 番号 <code>circuit_session.assigned_vlr</code>
フラグ	なし	<code>notProvidedFromVLR</code>
extendedRoutingInfo	-	<code>camelRoutingInfo</code>
gsmSCF 番号	OmniHSS API	MSC 番号 <code>circuit_session.assigned_msc</code>
フラグ	runtime.exs	CAMEL サービスキー <code>camel_service_key</code>
フラグ	runtime.exs	CAMEL トリガー検出ポイント <code>camel_trigger_detection_point</code>
CAMEL サービス	なし	CAMEL サービス
フラグ	なし	gsmSCF サービス

MSRN 番号

VLR 番号 home_vlr_prefixes

フラグ

필드명	API	설명
IMSI	OmniHSS API	OmniHSS 에서 받은 IMSI
VLR 번호	OmniHSS API	VLR 번호 [circuit_session.assigned_vlr]
상태		[notProvidedFromVLR]
extendedRoutingInfo	-	[routingInfo]
원래의 MSRN	PRN 번호	원래의 MSRN

참고사항

1. OmniSS7 에서 SendRoutingInfo
2. OmniSS7 에서 OmniHSS API
3. OmniSS7 에서 VLR 번호 home_vlr_prefixes

원래의 VLR 번호
→ CAMEL 번호

원래의 VLR 번호
→ MSC 번호 PRN
→ PRN 번호 MSRN
→ 원래의 MSRN 번호

참고사항

- OmniSS7 에서 OmniHSS
- OmniHSS 에서 IMSI VLR/MSC
- OmniSS7 에서 MAP

참고사항

```
# runtime.exs
home_vlr_prefixes: ["555123"], # VLR
```

- serving_vlr serving_msc null 27
- 1
- PRN 27
- PRN 27

UpdateLocation InsertSubscriberData

VLR

UpdateLocation

HLR	runtime.exs	HLR hlr_service_center_gt_address	"5551234568"
TCAP		ISD	END

InsertSubscriberData #1

Field	Value	Operation	Result
IMSI		UpdateLocation	"999999876543210"
MSISDN	OmniHSS API	OmniHSS	"555123456"
			"\n" (0x0A)
			:serviceGranted
			[<31>]
			[<17>, "!", "\"]
	runtime.exs	/ isd_network_access_mode	:packetAndCircuit

InsertSubscriberData #2 -

Field	Value	Operation	Result
SS			isd_send_ss_data: true

InsertSubscriberData #3 -

ISD	ISD	ISD	ISD
ISD #1	ISD	ISD	isd_send_call_barring: true
BAOC	ISD	ISD SS 146	ISD
BOIC	ISD	ISD SS 147	ISD
ISD	ISD	ISD	ISD

ISD ISD

- ISD #1 ISD - ISD
- ISD #2 ISD isd_send_ss_data: true ISD
- ISD #3 ISD isd_send_call_barring: true ISD

SendRoutingInfoForSM (SRI-for-SM) ISD

ISD SMS ISD MSC/SMSC ISD SMSc ISD SMS ISD HLR ISD SRI-for-SM ISD

ISD

ISD	ISD	ISD	ISD	ISD
IMSI	ISD	MSISDN ISD IMSI	PLMN_PREFIX + zero_padded_MSISDN	"00100
ISD ISD	runtime.exs	ISD SMS ISD SMSC GT ISD	smsc_service_center_gt_address	"55512

ISD runtime.exs ISD

```
# SRI-for-SM SRI-for-SM
# SSMSc MT-ForwardSM
smsc_service_center_gt_address: "5551234567", #

# MSISDN ↔ IMSI
# PLMN MCC001 = MNC01 =
hlr_imsi_plmn_prefix: "001001", #
```

MSISDN ↔ IMSI

OmniSS7 SRI-for-SM IMSI

- `hlr_imsi_plmn_prefix` IMSI MCC+MNC "50557"
MCC=505 MNC=57
- `hlr_msisdn_country_code` IMSI→MSISDN "61"
"1"/
- `hlr_msisdn_nsn_offset` MSISDN NSN 0
MSISDN
- `hlr_msisdn_nsn_length` MSISDN NSN

MSISDN ↔ IMSI

MSISDN IMSI

SendRoutingInfoForSM SRI-for-SM HLR IMSI
SSMSc MSISDN

- SSMSc MSISDN SRI-for-SM "5551234567"
- HLR IMSI
- HLR SRI-for-SM IMSI
- SSMSc MT-ForwardSM MSC/VLR IMSI

OmniSS7 - IMSI

OmniSS7 000000000000 MSISDN 0 IMSI 0000000000000000000000 IMSI0000000000000000

1. 000000 HLR 0000000000 IMSI 00000000 SS7 000000 SRI-for-SM 000000
2. 000000 SRI-for-SM 0000000000 HLR 000000 IMSI 00 - IMSI 000 MSISDN 000000

000000

MSISDN 000000 IMSI 00000000MCC+MNC 00000000

```
IMSI = PLMN_PREFIX + zero_padded_MSISDN
```

0000

- **PLMN_PREFIX** MCC + MNC 0000 "001001" 00000000
- **MSISDN** 000000000000
- 000000000000 IMSI 000 15 0

000000

```
# 00
plmn_prefix = "001001" # MCC 001 + MNC 01

# 000000 SRI-for-SM 000 MSISDN TBCD 000
msisdn = "555123456" # 9 000

# 00 1000000000000000
subscriber_digits = 15 - String.length("001001") # = 9 000

# 00 2000000 MSISDN 00000000
padded_msisdn = String.pad_leading("555123456", 9, "0") # =
"555123456"0000000

# 00 3000000 PLMN 00 + 000 MSISDN
imsi = "001001" <> "555123456" # = "001001555123456"000 15 00
```

000000

MSISDN	PLMN		MSISDN	IMSI	
"555123456"	"001001" (6)	9	"555123456"	"001001555123456"	
"99"	"001001" (6)	9	"000000099"	"001001000000099"	
"999999999"	"001001" (6)	9	"999999999"	"001001999999999"	
"91123456789"	"001001" (6)	9	"555123456"	"001001555123456"	

MSISDN

- MSISDN "99" → "000000099"
- MSISDN "91123456789" → "555123456"
- IMSI 15

IMSI → MSISDN

SMSc IMSI MSISDN

```
# SRI-for-SM IMSI
imsi = "001001555123456"

# 1 PLMN
plmn_prefix = "001001"
subscriber_portion = String.slice(imsi, 6, 9) # = "555123456"

# 2 MSISDN
msisdn = String.replace_leading(subscriber_portion, "0", "") # =
"555123456"
```

Table 1

IMSI	PLMN	Subscriber Portion	MSISDN	MSISDN
"001001555123456"	"001001"	"555123456"	"555123456"	"555123456"
"001001000000099"	"001001"	"000000099"	"99"	"99"
"001001999999999"	"001001"	"999999999"	"999999999"	"999999999"

Table 2

- MSISDN IMSI
- IMSI MSISDN
- hlr_imsi_plmn_prefix
- IMSI
- HLR
- 15 IMSI 15

MSISDN

HLR SRI-for-SM MSISDN TBCD

1. **TBCD** TBCD TON/NPI "91"
- 2.
- 3.

4. PLMN ID + MSISDN

Scenario

SRI-for-SM IMSI HLR IMSI
UpdateLocation SendAuthenticationInfo

Scenario

1. SSMc → HLR: SRI-for-SM
- MSISDN TBCD "91123456789" TON/NPI
2. HLR
- TBCD "91123456789"
- "91123456789" 11
- 9 "555123456" 9
- PLMN "001001" + "555123456" = "001001555123456"
- SMSC GT "5551234567"
3. HLR → SSMc: SRI-for-SM
- IMSI "001001555123456" 15
- "5551234567" MT-ForwardSM
4. SSMc MT-ForwardSM "5551234567" IMSIN "001001555123456"

Scenario

runtime.exs

```
# PLMN MCC001 = MNC01 =  
hlr_imsi_plmn_prefix: "001001",  
  
# NSN MSISDN  
hlr_msisdn_country_code: "1", # IMSI→MSISDN  
hlr_msisdn_nsn_offset: 1, # 1  
hlr_msisdn_nsn_length: 10 # 10 NSN
```

NSN

MSISDN "68988000088" NSN "88000088"

- `hlr_msisdn_nsn_offset` NSN
- `hlr_msisdn_nsn_length` MSISDN NSN

		MSISDN	nsn_offset	nsn_length	NSN
1 CC	"9"	"95551234567"	1	10	"5551234567"
2 CC	"99"	"99412345678"	2	9	"412345678"
3 CC	"999"	"99988000088"	3	8	"88000088"

1. MSISDN → IMSI

```
MSISDN: "99988000088"  
NSN: String.slice("99988000088", 3, 8) = "88000088"  
NSN: "088000088"  
IMSI: "547050" + "088000088" = "547050088000088"
```

2. IMSI → MSISDN

```
IMSI: "547050088000088"  
NSN: "088000088"  
NSN: "88000088"  
MSISDN: "+999" + "88000088" = "+99988000088"
```

API Overview - SRI-for-SM

API Overview

Overview

API	Component	Details
OmniHSS API	OmniHSS Component	IMSI, MSISDN, circuit_session, VLR/MSC
runtime.exs	OmniSS7 Component	smc_service_center_gt_address, camel_service_key, isd_network_access_mode
API	Component	SS Component
API	MAP Component	UpdateLocation, IMSI, SRI, MSISDN
API	Component	SRI-for-SM, IMSI, hlr_imsi_prefix + NSN

Overview

runtime.exs Overview

- hlr_service_center_gt_address - UpdateLocation
- smc_service_center_gt_address - SRI-for-SM, MT-ForwardSM

runtime.exs Configuration

- camel_service_key - 11_110
- camel_trigger_detection_point - :termAttemptAuthorized

- `isd_network_access_mode` - `packetAndCircuit`
- `isd_send_ss_data` - `true`
- `isd_send_call_barring` - `true`
- `hlr_imsi_plmn_prefix` - `"001001"` MSISDN↔IMSI PLMN

OmniHSS

OmniHSS REST API

- IMSI MSISDN
- VLR/MSC
-
-

OmniSS7

- ←
-
- MAP
-
-

OmniHSS OmniHSS OmniHSS IMSI

OmniSS7 Omnitouch

MAP

←

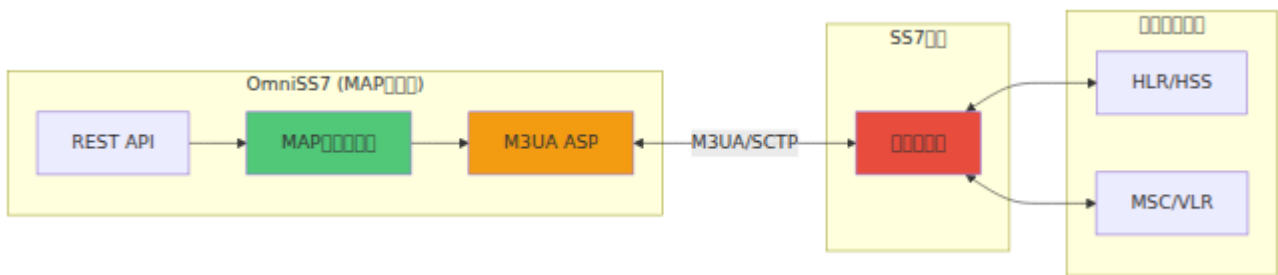
OmniSS7 MAP

1. MAP
 2. MAP
 3. MAP
 4. API
 - 5.
 - 6.
-

MAP

MAP OmniSS7** ASP M3UA STP SGP / MAP
**

- HLR SRI SRI-for-SM
-
- PRN



MAP

config/runtime.exs MAP M3UA

□□□□

```
config :omniss7,  
  # □□MAP□□□□□  
  map_client_enabled: true,  
  
  # MAP□□□□M3UA□□□□□ASP□□□□□STP/SGP□  
  map_client_m3ua: %{\br/>    mode: "ASP", # M3UA□□□"ASP"□□□□□□"SGP"□□□  
  }  
  □□  
  callback: {MapClient, :handle_payload, []}, # □□□□□□□□□  
  process_name: :map_client_asp, # □□□□□□□□  
  local_ip: {10, 0, 0, 100}, # □□IP□□  
  local_port: 2905, # □□SCTP□□  
  remote_ip: {10, 0, 0, 1}, # □□STP/SGP IP  
  remote_port: 2905, # □□STP/SGP□□  
  routing_context: 1 # M3UA□□□□□  
}
```

□□□□□□

```
config :omniss7,  
  # □□□□MAP□□□  
  map_client_enabled: true,  
  
  # □□M3UA□□  
  map_client_m3ua: %{:mode: "ASP",  
    callback: {MapClient, :handle_payload, []},  
    process_name: :map_client_asp,  
    local_ip: {10, 0, 0, 100},  
    local_port: 2905,  
    remote_ip: {10, 0, 0, 1},          # □□STP IP  
    remote_port: 2905,  
    routing_context: 1  
  }  
  
config :control_panel,  
  web: %{:listen_ip: "0.0.0.0",  
    port: 443,  
    hostname: "ss7-gateway.example.com",  
    enable_tls: true,  
    tls_cert: "/etc/ssl/certs/gateway.crt",  
    tls_key: "/etc/ssl/private/gateway.key"  
  }
```

MAP

1. SM SRI-for-SM

HLR SMS MSC HLR SRI-for-SM HLR SRI-for-SM

API `POST /api/sri-for-sm`

Request

```
{
  "msisdn": "447712345678",
  "serviceCenter": "447999123456"
}
```

Response

```
{
  "result": {
    "imsi": "234509876543210",
    "locationInfoWithLMSI": {
      "networkNode-Number": "447999555111"
    }
  }
}
```

cURL

```
curl -X POST http://localhost/api/sri-for-sm \
-H "Content-Type: application/json" \
-d '{
  "msisdn": "447712345678",
  "serviceCenter": "447999123456"
}'
```

2. SRI

HLR

API `POST /api/sri`

Request

```
{
  "msisdn": "447712345678",
  "gmsc": "447999123456"
}
```

□□□

```
{
  "result": {
    "imsi": "234509876543210",
    "extendedRoutingInfo": {
      "routingInfo": {
        "roamingNumber": "447999555222"
      }
    }
  }
}
```

3. □□□□□□□□PRN□

□□□□MSC□□□□□□□□MSRN□□

API□□□ `POST /api/prn`

□□□

```
{
  "msisdn": "447712345678",
  "gmsc": "447999123456",
  "msc_number": "447999555111",
  "imsi": "234509876543210"
}
```

4. HLR Update

HLR Update

API POST /api/send-auth-info

Request

```
{
  "imsi": "234509876543210",
  "vectors": 5
}
```

Response

```
{
  "result": {
    "authenticationSetList": [
      {
        "rand": "0123456789ABCDEF0123456789ABCDEF",
        "xres": "ABCDEF0123456789",
        "ck": "0123456789ABCDEF0123456789ABCDEF",
        "ik": "FEDCBA9876543210FEDCBA9876543210",
        "autn": "0123456789ABCDEF0123456789ABCDEF"
      }
    ]
  }
}
```

5. HLR Location Update

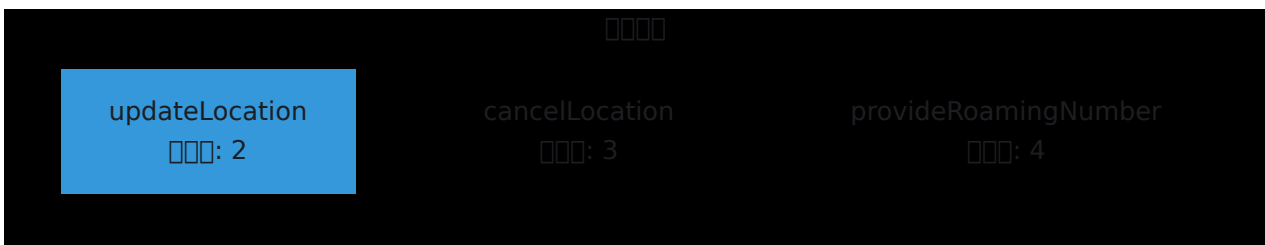
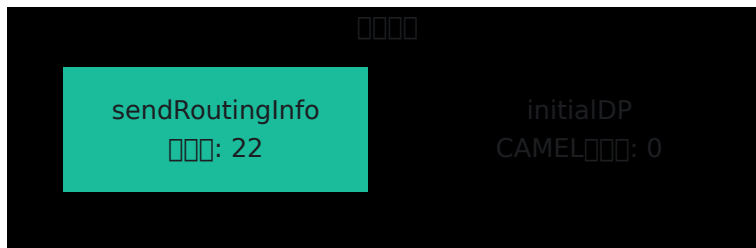
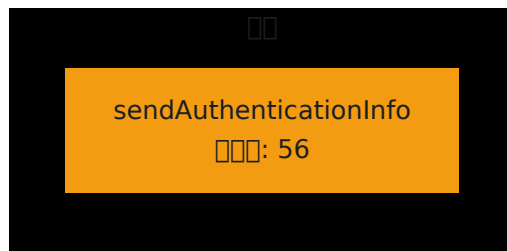
HLR Location Update

API POST /api/updateLocation

Request

```
{  
  "imsi": "234509876543210",  
  "vlr": "447999555111"  
}
```

MAP



API

Swagger UI

Swagger UI는 API를 시각적으로 보여주고 테스트할 수 있는 도구입니다. SS7 관련 API를 관리하는 데 유용합니다.

Swagger UI

1. `http://your-server/swagger`에 접속합니다.
2. API 목록을 확인합니다.
3. 필요한 API를 선택합니다.

예시:

1. `/api/sri-for-sm` API를 선택합니다.
2. "시작" 버튼을 클릭합니다.
3. API 호출을 시도합니다.
4. "시작" 버튼을 클릭합니다.
5. 결과를 확인합니다.

API 응답

- **200** - 성공적으로 응답합니다.
- **400** - 잘못된 요청입니다.
- **504** - SS7 관련 10초 이상 지연입니다.

MAP

MAP

MAP

- `map_requests_total` - MAP 요청 수
 - `operation`, `sri`, `sri_for_sm`, `prn`, `authentication_info`

- `map_request_errors_total` - MAP
 - `operation`
- `map_request_duration_milliseconds` - MAP
 - `operation`
- `map_pending_requests` - MAP

Prometheus

```
#           SRI-for-SM  
increase(map_requests_total{operation="sri_for_sm"}[1h])

# SRI          
rate(map_request_duration_milliseconds_sum{operation="sri"}[5m]) /
rate(map_request_duration_milliseconds_count{operation="sri"}[5m])

#   MAP          
sum(rate(map_request_errors_total[5m])) by (operation)

#           
map_pending_requests
```

MAP

- API 504
- HLR/MS

1. M3UA

```
# IEx
:sys.get_state(:map_client_asp)
```

2. STP
3. SCCP
4. SCCP

SCCP

- API SCCP
- "SCCP unitdata service"

SCCP

- STP
- HLR SSN 6
-

- STP
-
-

- ←
- - Web UI API
- STP -
- -

- 0000 - 0000

OmniSS7 Omnitouch 000000

OmniSS7 (SMSc) Overview

← Introduction

OmniSS7 provides a **SMSc (SMS Gateway)** service that enables **OmniMessage** to send and receive SMS messages.

OmniMessage

OmniSS7 SMSc connects **SS7** networks to **OmniMessage** via **OmniMessage** and **SMS** services.

- **OmniSS7 (SMSc)** connects **SS7/MAP** to **SCCP**.
- **OmniMessage (SMS)** connects to **OmniMessage**.

OmniMessage

OmniMessage provides **SMS** services.

- **OmniMessage** provides **SMS** services.
- **OmniMessage** provides **(DLR)** services.
- **SMSc** connects **OmniMessage** to **SS7**.
- **OmniMessage** provides **TPS** services.
- **API** provides **RESTful HTTP API** services.
- **OmniMessage** provides **SMS** services.

OmniMessage provides **OmniMessage** via **OmniSS7** and **HTTPS API** services.

OmniSS7 **SMSc** connects **OmniMessage** to **SS7/MAP** via **OmniMessage** and **OmniMessage** services.



1. OmniMessage
2. SMS
3. SMSc
4. HTTP API
- 5.
- 6.
- 7.
8. SMSc
9. VLR
- 10.
- 11.
- 12.

SMS

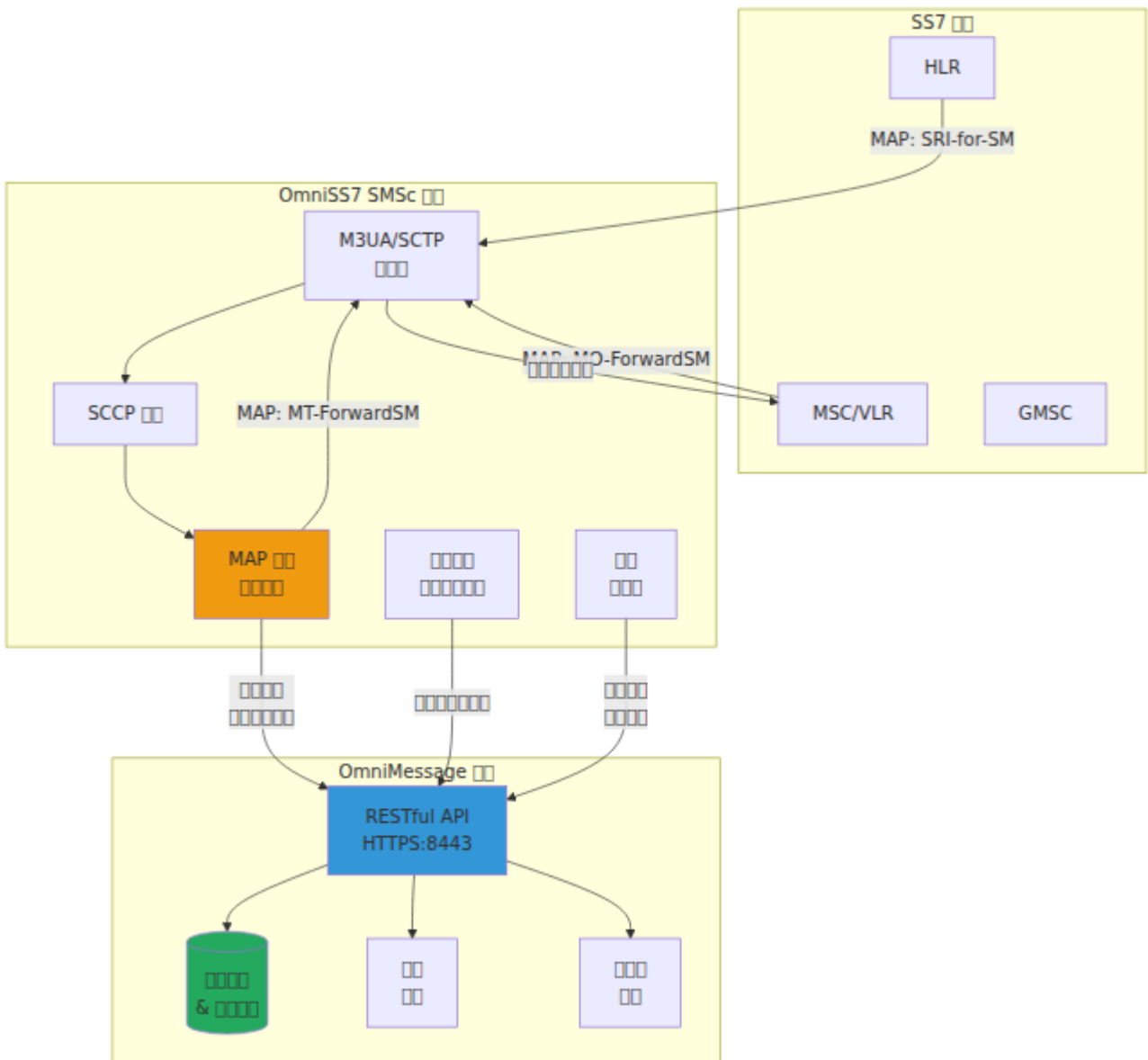
OmniSS7 SS7

OmniMessage

SMS OmniSS7 SMSc

- **MT-SMS**
- **MO-SMS**
-
- SMS
-

SMS



SS7 SMSc

OmniSS7 is a SMSc component that connects to the SS7 network.

SS7 SMSc

OmniSS7 is configured in `config/runtime.exs` to connect to the SS7 network.

1. `config/runtime.exs`

2. 00 00000000 53-204 000
 - 00 1 0STP 0000 53-95 00
 - 00 2 0HLR 0000 97-142 00
 - 00 3 0SMSc 0000 144-204 00
3. 000 0000000000000000 #
4. 0000 SMSc 00000 144-204 0000 #
5. 00000000 0000
6. 00 00000 `iex -S mix`

SMSc 0000

000 SMSc 00000000

```

config :omniss7,
  # STP + SMSc
  # map_client_enabled true SMSc
  map_client_enabled: true,
  hlr_mode_enabled: false,
  smsc_mode_enabled: true,

  # OmniMessage API
  smsc_api_base_url: "https://10.179.3.219:8443",
  # SMSc
  smsc_name: "ipsmgw",
  # GT
  smsc_service_center_gt_address: "5551234567",

  # SMS
  auto_flush_enabled: true,
  auto_flush_interval: 10_000,
  auto_flush_dest_smsc: "ipsmgw",
  auto_flush_tps: 10,

  # M3UA
  # ASP / MAP SMS
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :stp_client_asp,
    # SMSc
    local_ip: {10, 179, 4, 12},
    local_port: 2905,
    # STP
    remote_ip: {10, 179, 4, 10},
    remote_port: 2905,
    routing_context: 1
  }

config :control_panel,
  use_additional_pages: [
    {SS7.Web.EventsLive, "/events", "SS7"},
    {SS7.Web.TestClientLive, "/client", "SS7"},
    {SS7.Web.M3UAStatusLive, "/m3ua", "M3UA"},
    {SS7.Web.RoutingLive, "/routing", ""},
    {SS7.Web.RoutingTestLive, "/routing_test", ""},
    {SS7.Web.SmscLinksLive, "/smsc_links", "SMSc"}
  ]

```

```
],
  page_order: ["/events", "/client", "/m3ua", "/routing",
"/routing_test", "/smsc_links", "/application", "/configuration"]
```

□□□□□□□□

□□□□□□□□□□□□□□□□ □□□□

参数	数据类型	默认值	说明
<code>smsc_api_base_url</code>	字符串	空字符串	OmniMessage API 地址
<code>smsc_name</code>	字符串	"{hostname}_SMSc"	SMSc 名称
<code>smsc_service_center_gt_address</code>	字符串	空字符串	SMSc 地址
<code>auto_flush_enabled</code>	布尔值	true	是否启用自动刷新
<code>auto_flush_interval</code>	整数	10_000	自动刷新的间隔
<code>auto_flush_dest_smsc</code>	字符串	空字符串	SMSc 地址
<code>auto_flush_tps</code>	整数	10	每秒刷新次数
<code>local_ip</code>	字符串	空字符串	SMSc 本地 IP
<code>local_port</code>	整数	2905	SCTP 端口
<code>remote_ip</code>	字符串	空字符串	SS7 STP IP

Property	Type	Value	Description
remote_port	Integer	2905	SCTP Port
routing_context	Integer	1	M3UA Routing Context ID

SS7 SCS Configuration

map_client_enabled: true map_client_enabled: true Web UI

- SS7 -
- SS7 - MAP
- M3UA -
- STP -
- STP -
- SMS Sc API + SMS ← SMS Sc
-
-

HLR Configuration

Configuration

- SMS Sc map_client_enabled: true
- OmniMessage OmniMessage API smsc_api_base_url
- 5 SMS.FrontendRegistry OmniMessage
- API OmniMessage API 5
- MAP SRI-for-SM 10 MT-ForwardSM 30 2G/CS VLR
- SMS
- STP M3UA / MAP SMS
-

- **Web UI** Web UI
 - **API** REST API Swagger UI API
-

HTTP API

OmniMessage

OmniSS7 HTTPS REST API OmniMessage

```
config :omniss7,  
  # OmniMessage API URL  
  smsc_api_base_url: "https://10.5.198.200:8443",  
  # SMSC hostname_SMSC  
  smsc_name: "omni-smsc01",  
  # GT  
  smsc_service_center_gt_address: "5551234567"
```

項目	種別	単位	値	注
smsc_api_base_url	文字列		"https://localhost:8443"	C A
smsc_name	文字列		"{hostname}_SMSc"	[S
smsc_service_center_gt_address	文字列		"5551234567"	[C F S

設定

OmniMessage 5 SMS.FrontendRegistry OmniMessage

-
-
-
- SMS

設定

- 5
- smsc_mode_enabled: true

設定

URI	Method	Request	Response
/api/frontends	POST	JSON	{ "frontend_name": "...", "frontend_type": "SMS", "hostname": "...", "uptime_seconds": ... }
/api/messages_raw	POST	JSON SMS	{ "source_msisdn": "...", "source_smsc": "...", "message_body": "... }
/api/messages	GET	JSON	URI: smsc: <smsc_name>
/api/messages/{id}	PATCH	JSON	{ "deliver_time": "...", "dest_smsc": "... }
/api/messages/{id}	PUT	JSON	{ "dest_smsc": null }
/api/locations	POST	JSON	{ "msisdn": "...", "imsi": "...", "location": "...", "ims_capable": true, "csfb": false, "expires": "...", "user_agent": "...", "ran_location": "...", "imei": "...", "registered": "... }
/api/events	POST	JSON	{ "message_id": ..., "name": "...", "description": "... }
/api/status	GET	JSON	-

API Endpoints

API endpoints return JSON responses.

- 返回HTTP 200-201 JSON 数据
- 返回HTTP 4xx/5xx 错误
- 返回ISO 8601 时间戳 "2025-10-21T12:34:56Z"
- 返回 ID

API 接口

SMS 接口

1. SMS.APIClient

API 接口 OmniMessage HTTP API

- `frontend_register/4` - OmniMessage
- `insert_message/3` - SMS Python 3
- `insert_location/9` - /
- `get_message_queue/2` - 消息队列
- `mark_dest_smsc/3` - 标记目的地
- `add_event/3` - 添加事件
- `flush_queue/2` - SRI-for-SM + MT-forwardSM
- `auto_flush/2` - 自动刷新

2. SMS.FrontendRegistry

注册接口

- 注册
- 5 注册
- 注册 `smc_name`
- 注册

3. SMS.Utils

SMS 工具

- `generate_tp_scts/0` - TPDU SMS
-

□□□□□□

□□ **SMS** □□□□□□□□

M3UA 00 SCTP 000

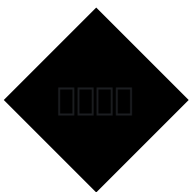
M3UA 00000

00 SCCP 0000

00 SCCP 00

00 TCAP/MAP 00

00 MAP 00



Forward-SM

00 SMS TPDU

000000



□□□□□□

POST □
/api/messages_raw

POST □ /api/events

□□ MAP □□

□□ **SMS** □□□□□□□□

□□□□ HSS/HLR API (`hlr_api_base_url`) □□IP-SM-GW □□□ MT-SMS □□□ □□□□□□□□□
SRI-for-SM □□□□□□□□□ HSS API □□□□□□□ IMSI □□□ VLR□**VLR** □□ □□□□□□□□□□
□□□□□□□□□ HSS □□□

M3UA □□ SCTP □□□

M3UA □□□□□

□□ SCCP □□□□

Core OmniCore OmniCall Omni
▼ 5GC ▼ ▼ ▼

□□ TCAP/MAP □□

□□ MAP □□

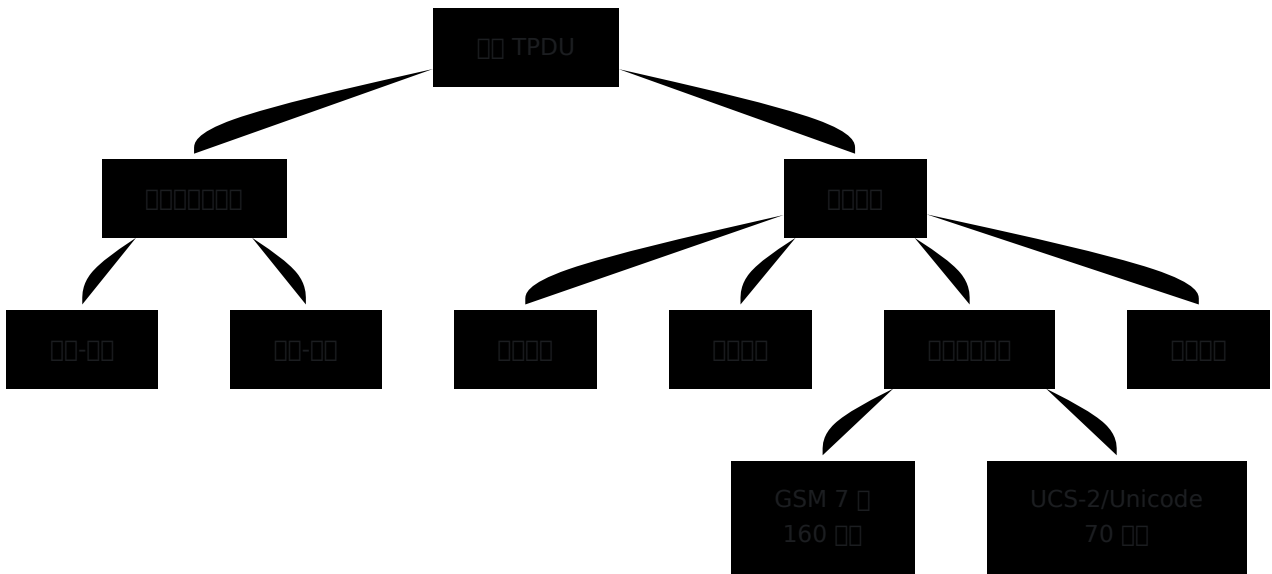
□□□□

Forward-SM

□□ SMS TPDU

□□□□□□

SMS TPDU



SMS-PP

SMS Sc HLR alertServiceCenter

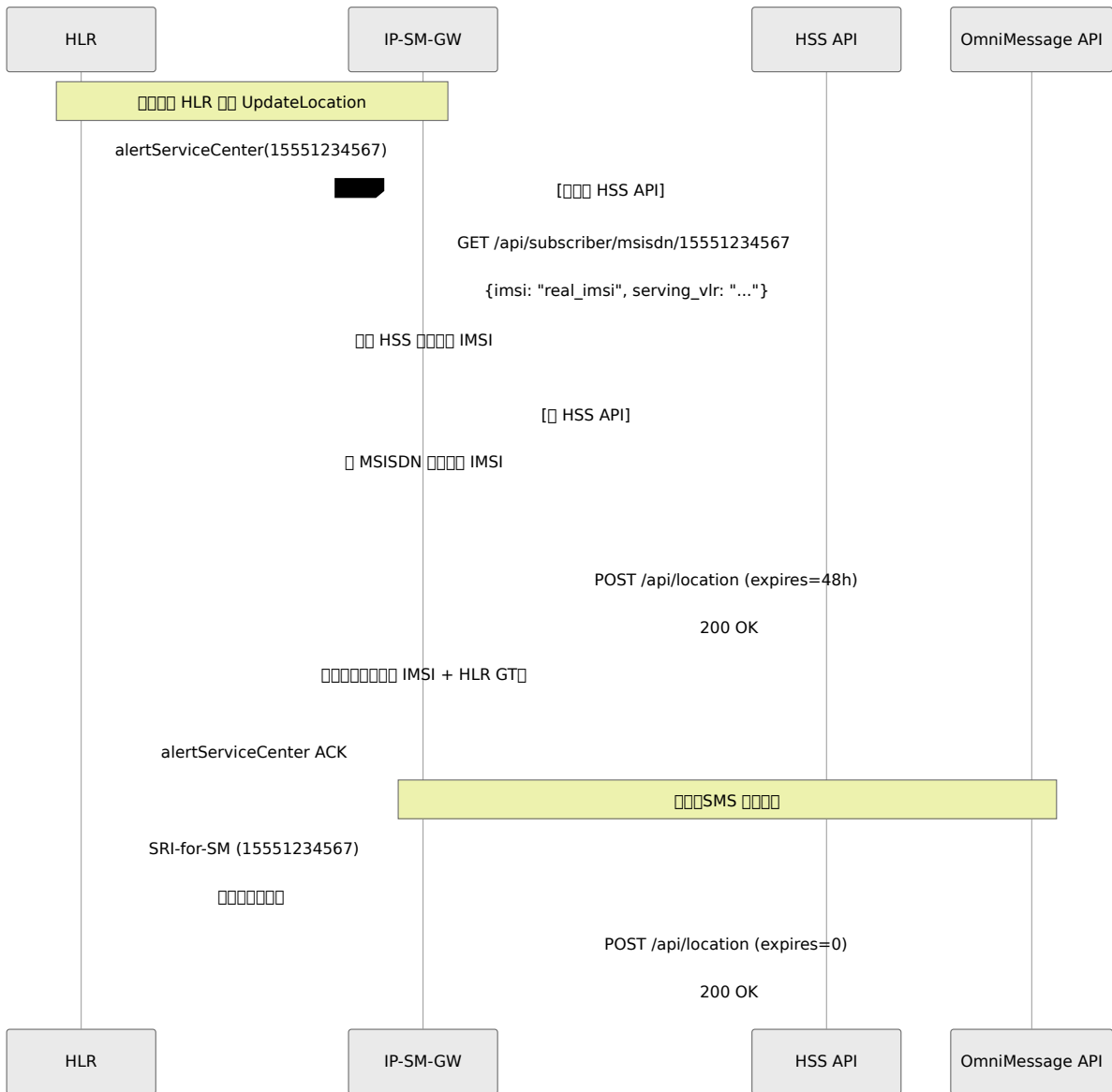
HLR alertServiceCenter HLR

alertServiceCenter

HLR UpdateLocation VLR/MSCHLR alertServiceCenter MAP 64 SMS Sc

HLR

```
config :omniss7,  
  # SMS Sc alertServiceCenter 48  
  hlr_alert_location_expiry_seconds: 172800
```

API

POST /api/location

```
{
  "msisdn": "15551234567",
  "imsi": "001010123456789",
  "location": "ipsmgw",
  "ims_capable": false,
  "csfb": true,
  "expires": "2025-11-01T12:00:00Z",
  "user_agent": "15551111111",
  "ran_location": "SS7",
  "imei": "",
  "registered": "2025-10-30T12:00:00Z"
}
```

user_agent alertServiceCenter HLR GT SCSMSc HLR

expires

SS7

SMS Sc SS7 MO→MT

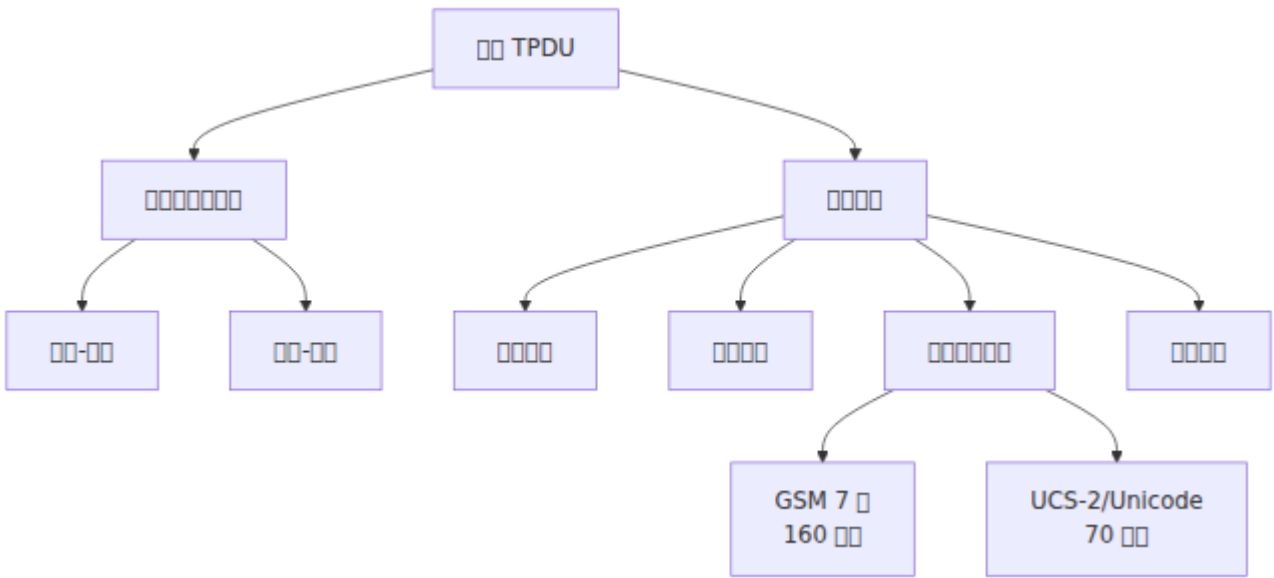
SS7

SMS Sc SS7 (MO) SMS source_smsc "SS7_MO_15551234567"

1. SS7 → source_smsc "SS7"
2. →
3. SS7 →

SS7

SS7 IP-SM-GW MSC MSC MO→MT



TPDU

TPDU SS7 TPDU

1. **VLR** TPDU VLR TPDU
2. **HSS API** TPDU `hlr_api_base_url` TPDU MSISDN TPDU HSS API TPDU HSS TPDU

TPDU TPDU

TPDU

TPDU SMSc TPDU `source_smsc` TPDU

- TPDU `source_smsc` TPDU **"SS7"** TPDU
 - TPDU
 - TPDU `"TPDU"` TPDU
 - TPDU PUT TPDU
 - TPDU
- TPDU `source_smsc` TPDU **"SS7"** TPDU
 - TPDU — TPDU MO→MT TPDU
 - TPDU HSS API / VLR TPDU

- `source_smsc` `"SS7"`
 - `source_smsc` Web API/SMPP `source_smsc` SS7 `source_smsc`

SMSC

`source_smsc`

Event	Source	Destination
SS7 MO-FSM	"SS7_M0_15551234567"	Phone - Phone
SS7 MO-FSM	"SS7_M0_15551234567"	Phone - Phone MO→MT
API/SMPP	"ipsmgw" & "api_gateway"	Phone
SMSc	"smsc-node-01"	Phone

Example

Example

```
{
  "message_id": 12345,
  "name": "SMS",
  "description": "SMS - source_smsc 'SS7_M0_15551234567' to 'SS7'
  phone number"
}
```

Example

- **Web UI** SS7 `/events`
- `source_smsc` API `events`
- `source_smsc`

- **HLR** 数据库 alertServiceCenter 的 HLR
- 数据库/数据库
- 数据库
- **Web UI** 数据库

数据库

数据库

数据库	数据库	数据库
msisdn	数据库	"15551234567"
imsi	数据库 IMSI 数据库 HSS 数据库	"001010123456789"
hlr_gt	数据库 alertServiceCenter 的 HLR GT	"15551111111"
messages_sent	数据库 MT-FSM 数据库	5
messages_received	数据库 MO-FSM 数据库	2
status	:active 或 :failed	:active
updated_at	数据库 Unix 数据库	1730246400
vlr_address	数据库 VLR GT 数据库	"14155550100"
vlr_cached_at	VLR 数据库 Unix 数据库	1730246400

- `messages_sent` `updated_at`

SRI-for-SM

- `status = :failed`

SRI-for-SM

- `status = :failed`

ETC

- ETS
- Web UI

Web UI - SMSc

`/smsc_subscribers`

SMSc `config/runtime.exs` SMSc

SMSc

1.

- MSISDN IMSI HLR GT VLR GT VLR
-
-
-
-

2.

-

- 0000000000
- 0000000000
- 00 HLR 000

3. 00

- 00000000000000000000
- 000000000000

0000

SMSc 000000		0003		
MSISDN	IMSI	HLR GT	VLR GT	Msg S/
15551234567	001010123456789	15551111111	14155550100	5/2
15559876543	001010987654321	15551111111	-	0/6
15551112222	001010111222233	15552222222	14155550200	3/1

000 0003 | 0002 | 0001 | 00 HLR02

API 00

00000000000000000000

1. **HSS API** `hlr_api_base_url` HSS API `hlr_api_base_url` VLR `hlr_api_base_url` IMSI `hlr_api_base_url` VLR `hlr_api_base_url`

2. **MO-ForwardSM** `hlr_api_base_url` MO-ForwardSM `hlr_api_base_url` SCCP `hlr_api_base_url` GT `hlr_api_base_url` VLR/MSC `hlr_api_base_url` VLR `hlr_api_base_url`

`hlr_api_base_url` MT `hlr_api_base_url` IMSI `hlr_api_base_url` VLR `hlr_api_base_url` TTL `hlr_api_base_url` HSS API `hlr_api_base_url`

`hlr_api_base_url`

`hlr_api_base_url` VLR `hlr_api_base_url`

- **MT-ForwardSM** `hlr_api_base_url` SCCP `hlr_api_base_url` VLR `hlr_api_base_url` VLR `hlr_api_base_url`
- **TTL** `hlr_api_base_url` `vlr_cache_ttl_seconds` `hlr_api_base_url` TTL `hlr_api_base_url`

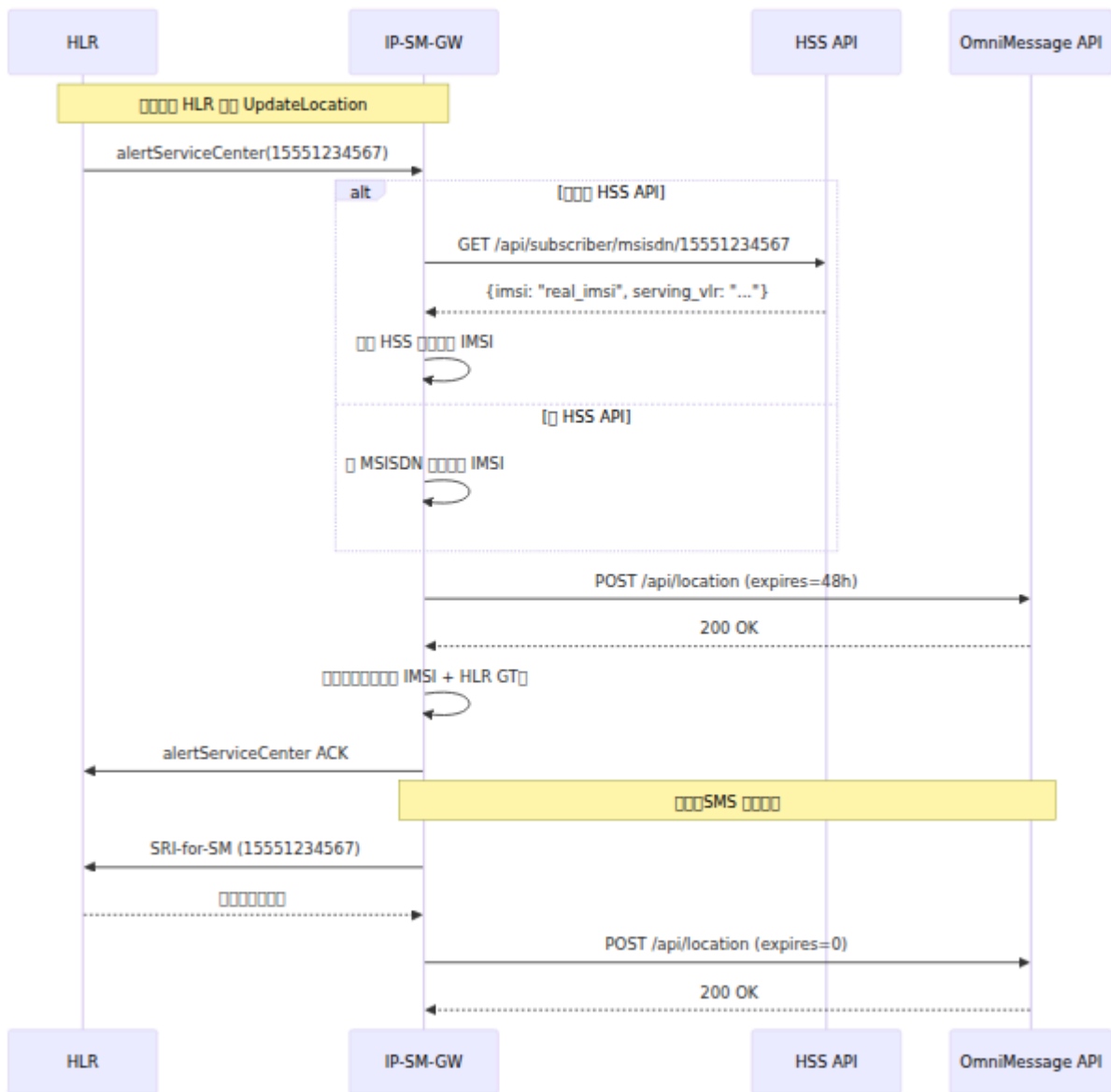
`hlr_api_base_url` alertServiceCenter `hlr_api_base_url` VLR `hlr_api_base_url` VLR `hlr_api_base_url`

`hlr_api_base_url`

```
config :omniss7,  
  # HSS/HLR API hlr_api_base_url VLR hlr_api_base_url  
  hlr_api_base_url: "https://10.179.2.140:8443",  
  
  # VLR hlr_api_base_url TTL hlr_api_base_url 3600 = 1 hlr_api_base_url  
  vlr_cache_ttl_seconds: 3600
```

<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>
<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>
<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>	<code>hlr_api_base_url</code>
<code>vlr_cache_ttl_seconds</code>	<code>vlr_cache_ttl_seconds</code>	<code>vlr_cache_ttl_seconds</code>	<code>vlr_cache_ttl_seconds</code>

□□□



□□□□ SMS □□

□□□□ □□□□□□□□□□ SMS □□□

□□□□□□□□□□ □□□□□□□□□□

- `smsc_delivery_duration_milliseconds` - 消息传递延迟

配置

```
# 队列深度
smsc_queue_depth

# 每5分钟的消息速率
rate(smsc_messages_delivered_total[5m]) /
(rate(smsc_messages_delivered_total[5m]) +
rate(smsc_messages_failed_total[5m]))

# 平均延迟
rate(smsc_delivery_duration_milliseconds_sum[5m]) /
rate(smsc_delivery_duration_milliseconds_count[5m])
```

消息传递 SMSG

消息传递流程

步骤

1. 消息生成
2. 消息排队
3. 消息传递
4. 通过 M3UA 消息传递
5. 通过 TPS 消息传递

消息传递性能

配置

- TPS 配置
- HLR 配置
- 消息传递

- 00000000

0000

- 00 auto_flush_tps
 - 00 HLR 000
 - 00000000
-

MT-forwardSM API

00 API 00 SMS

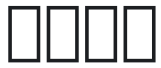
API 000 POST /api/MT-forwardSM

000

```
{  
  "imsi": "234509876543210",  
  "destination_serviceCentre": "447999555111",  
  "originating_serviceCenter": "447999123456",  
  "smsPDU":  
  "040B917477218345F600001570301857140C0BD4F29C0E9281C4E1F11A"  
}
```

000

```
{  
  "result": "success",  
  "message_id": "12345"  
}
```



OmniSS7

- ←
- HLR - HLR
 - SRI-for-SM - MSISDN IMSI
- - Web UI/API
- MAP - MAP
- -

OmniMessage **OmniMessage**
OmniMessage

OmniSS7 Omnitouch

M3UA & M2PA STP 簡介

← 簡介

OmniSS7 的 STP 簡介

目錄

1. 什麼是 STP?
 2. STP 的用途
 3. 傳統的 TDM 網絡
 4. 什麼是 STP
 5. 簡介
 6. M2PA 簡介
 - M3UA 與 M2PA 的關係
 - 什麼是 M2PA
 - SLTM 簡介
 - 什麼是 Web UI 與 M2PA
 - M2PA 簡介
 7. 簡介
 8. 簡介
 9. 簡介
 - 簡介
 - DROP 簡介 - 簡介
 10. 簡介
 11. 簡介
 12. 簡介
 13. M3UA 簡介
 - M3UA 簡介
 - IP 簡介
-

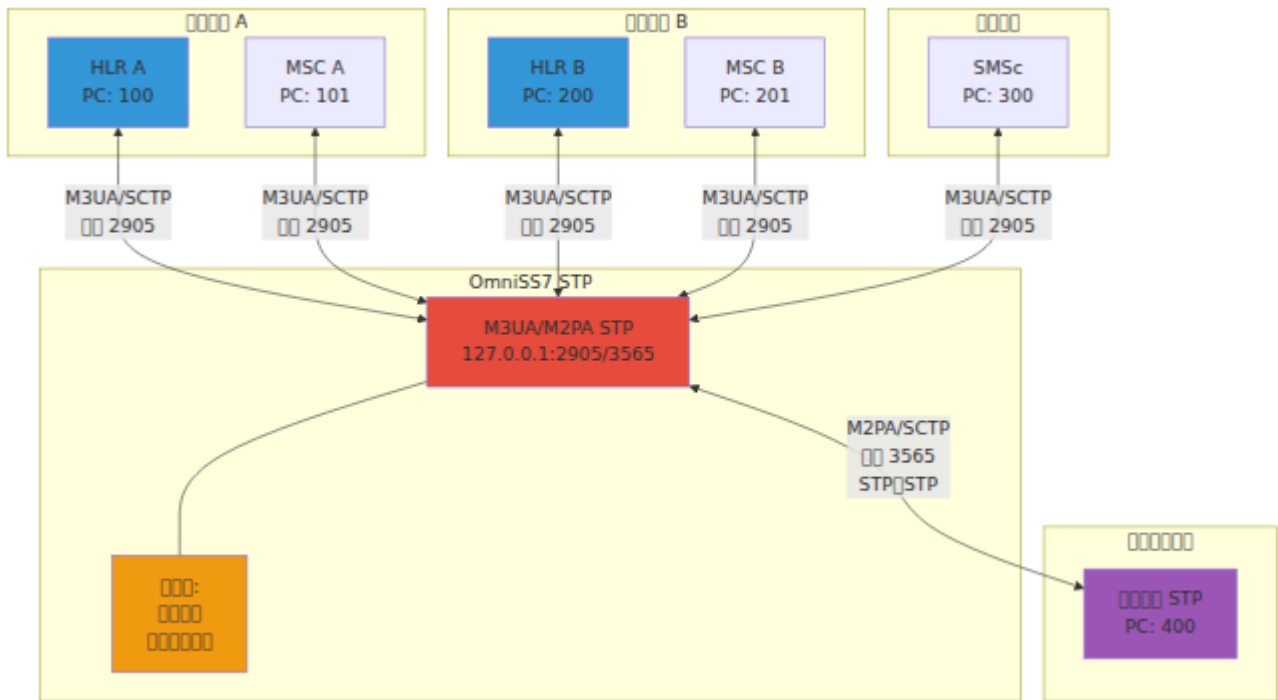
IP over SS7 (STP)?

IP over SS7 (STP) is a protocol that allows IP traffic to be transported over SS7 networks.

STP

- IP over SS7 (PC) is a protocol that allows IP traffic to be transported over SS7 networks.
- IP over SS7 (PC) is a protocol that allows IP traffic to be transported over SS7 networks.
- IP over SS7 (PC) is a protocol that allows IP traffic to be transported over SS7 networks.
- IP over SS7 (PC) is a protocol that allows IP traffic to be transported over SS7 networks.
- IP over SS7 (PC) is a protocol that allows IP traffic to be transported over SS7 networks.

STP



STP

ASP ()

- SGP/STP
-
- STP STP

SGP ()

- ASP
-
- STP

AS ()

- ASP


```

config :omniss7,
  # ASP
  map_client_m3ua: %{
    mode: "ASP",
    callback: {MapClient, :handle_payload, []},
    process_name: :sgw_connection,
    # (OmniSS7)
    local_ip: {10, 0, 0, 1},
    local_port: 0, #
    #
    remote_ip: {10, 0, 0, 100}, # SGW IP
    remote_port: 2905, # SGW M3UA
    routing_context: 1 # SGW
  },

  # STP
  peers: [
    %{
      peer_id: 1,
      name: "TDM_Gateway",
      role: :client, # OmniSS7
      local_ip: {10, 0, 0, 1},
      local_port: 0,
      remote_ip: {10, 0, 0, 100}, # SGW IP
      remote_port: 2905,
      routing_context: 1,
      point_code: 100, # SGW
      network_indicator: :international
    }
  ],

  # TDM
  m3ua_routes: [
    %{
      dest_pc: 100, # TDM MSC
      peer_id: 1, # SGW
      priority: 1,
      network_indicator: :international
    },
    %{
      dest_pc: 200, # TDM HLR
      peer_id: 1,
      priority: 1,

```

```

    network_indicator: :international
  }
]

```

SGW 配置

OmniSS7 配置

項目	設定	備考
IP	OmniSS7 の IP	外部 IP
ポート	OmniSS7 の SCTP ポート	2905
ネットワーク	SGW の AS	SGW ネットワーク
プロトコル	SGW と TDM 間のプロトコル	
アプリケーション	ASP	

接続

1 IP から TDM

OmniSS7 から SGW と TDM から HLR へ MAP (SRI-SM, PRN)

```

OmniSS7 (ASP) → SGW (SGP) → TDM → HLR
M3UA           MTP2           MTP3

```

2 OmniSS7 から IP から TDM から STP

OmniSS7 から IP から TDM から HLR

```

IP SMSC → OmniSS7 STP → SGW → TDM HLR
          ↓
          IP → HLR

```

3 SGW

```
peers: [  
  %  
    peer_id: 1,  
    name: "SGW_Primary",  
    role: :client,  
    remote_ip: {10, 0, 0, 100},  
    remote_port: 2905,  
    point_code: 100,  
    # ...  
  },  
  %  
    peer_id: 2,  
    name: "SGW_Backup",  
    role: :client,  
    remote_ip: {10, 0, 0, 101},  
    remote_port: 2905,  
    point_code: 100,  
    # ...  
  }  
],  
  
m3ua_routes: [  
  # SGW_Primary  
  %  
    dest_pc: 100, peer_id: 1, priority: 1, network_indicator:  
:international},  
  # SGW_Backup  
  %  
    dest_pc: 100, peer_id: 2, priority: 2, network_indicator:  
:international}  
]
```

M3UA STP

OmniSS7 STP STP

STP

OmniSS7 `config/runtime.exs` STP

1. `config/runtime.exs`
2. STP (53-174)
 - 1 STP (53-85)
 - 2 HLR (87-123)
 - 3 SMSc (125-174)
3. STP (#)
4. STP (53-85 #)
5. STP
6. `iex -S mix`

STP

STP

```
config :omniss7,  
  # STP - STP STP  
  map_client_enabled: true,  
  hlr_mode_enabled: false,  
  smsc_mode_enabled: false,  
  
  # M3UA STP  
  # ASP (M3UA) STP/SGW  
  map_client_m3ua: %{\br/>    mode: "ASP",  
    callback: {MapClient, :handle_payload, []},  
    process_name: :stp_client_asp,  
    # STP (M3UA)  
    local_ip: {10, 179, 4, 10},  
    local_port: 2905,  
    # STP/SGW STP  
    remote_ip: {10, 179, 4, 11},  
    remote_port: 2905,  
    routing_context: 1  
  }
```

STP/SGW

STP/SGW STP

項目	型別	デフォルト値	説明	設定値
<code>map_client_enabled</code>	ブール値	<code>true</code>	MAP クライアントを有効にする	<code>true</code>
<code>local_ip</code>	IP アドレス		ローカル IP アドレス (例: {10, 0, 0, 1})。サブネットマスク (例: [{10, 0, 0, 1}, {10, 0, 0, 2}])	{10, 179, 4, 10}
<code>local_port</code>	ポート番号	2905	ローカル SCTP ポート	2905
<code>remote_ip</code>	IP アドレス		リモート STP/SGW IP アドレス	{10, 179, 4, 11}
<code>remote_port</code>	ポート番号	2905	リモート SCTP ポート	2905
<code>routing_context</code>	整数	1	M3UA ルーティングコンテキスト ID	1
<code>enable_gt_routing</code>	ブール値	<code>false</code>	グローバルルーティングを有効にする (PC 経由)	<code>true</code>

ローカル IP アドレス `local_ip` とリモート IP アドレス `remote_ip` は、SCTP 接続を確立するために必要です。
SCTP 接続

STP 設定

`map_client_enabled: true` Web UI を有効にする

- **SS7** 設定 - 設定
- **SS7** ネットワーク - MAP ネットワーク
- **M3UA** - ネットワーク
- ネットワーク - ネットワーク ← STP ネットワーク
- ネットワーク - ネットワーク ← STP ネットワーク
- ネットワーク - ネットワーク
- ネットワーク - ネットワーク

HLR 和 SMSc 的部署

部署

- 部署 SCTP 和 (IP 为 132) 的
- 部署 M3UA 和 2905
- 部署
- 部署 Web UI 和 API 部署 **Mnesia** 和 部署
- 部署 `runtime.exs` 部署 Mnesia 部署
- 部署
- **Web UI** 部署 **Web UI** 部署 Web 部署
- **API** 部署 **API** 部署 REST API 部署 Swagger UI 部署

部署 STP

部署 `map_client_enabled: true` 部署 STP 部署 **M3UA STP** 部署

部署 STP

部署 `config/runtime.exs`

```
config :omniss7,  
  sctp_handler: %{  
    enabled: true,  
    local_ip: {127, 0, 0, 1}, # 本地 IP  
    local_port: 2905, # 本地端口  
    point_code: 100 # STP 点代码  
  }
```

STP 配置

参数	类型	默认值	描述	范围
<code>enabled</code>	布尔	<code>false</code>	是否启用 STP	<code>true</code>
<code>local_ip</code>	IP	<code>{127, 0, 0, 1}</code>	本地 IP	<code>{0, 0, 0, 0}</code>
<code>local_port</code>	端口	<code>2905</code>	本地端口	<code>2905</code>
<code>point_code</code>	点码	<code>00</code>	STP 点码 SS7	<code>100</code>

配置 STP

- 配置 M3UA 和 MAP 协议
- 配置 STP 协议
- 配置 STP 协议 HLR、MSC 和 SMSC

配置 STP 协议 `map_client_m3ua` 和 `sctp_handler`

配置 (Mnesia)

配置 Mnesia 数据库

配置

1. **Runtime.exs** 配置 `config/runtime.exs` 中的 `peers` 配置
包括 `m3ua_peers`、`m3ua_routes` 和 `m3ua_gt_routes`
2. **Web UI** 配置 `Web UI` 配置 Mnesia
3. 配置 `runtime.exs` 中的 Mnesia 配置
4. 配置 Web UI 配置

Mnesia 配置

配置 Mnesia 存储类型

```
config :omniss7,  
  mnesia_storage_type: :disc_copies # 配置 :ram_copies 配置
```

配置项	默认值	配置项	默认值
<code>:disc_copies</code>	配置项	配置项	配置项
<code>:ram_copies</code>	配置项	配置项	配置项

配置 `:disc_copies`

Mnesia 部署

Mnesia 部署 Mnesia 配置

- 配置 `Mnesia.{node_name}/` 配置 `Mnesia.nonnode@nohost/`
- 配置 `m3ua_peer` `m3ua_route` `m3ua_gt_route`

配置

配置

- Runtime.exs** - 配置
- Web UI** - 配置 [Web UI](#)
- REST API** - 配置 [API](#)

配置 `runtime.exs` 配置 Web UI

配置 M3UA

配置 M3UA 配置 STP HLR MSC SMSC 配置 `config/runtime.exs`

□□□□□□□□

| □□□□□□ peers □□□□□□□□□□ m3ua_peers □□□□□□□□□□□□□□

```

config :omniss7,
  peers: [
    # STP STP:client
    %{
      peer_id: 1,
      name: "Partner_STP_West",
      role: :client,
      local_ip: {10, 0, 0, 1},
      local_port: 0,
      remote_ip: {10, 0, 0, 10},
      remote_port: 2905,
      routing_context: 1,
      point_code: 100,
      network_indicator: :international
    },
    # HLR STP:client
    %{
      peer_id: 2,
      name: "Local_HLR",
      role: :client,
      local_ip: {10, 0, 0, 1},
      local_port: 0,
      remote_ip: {10, 0, 0, 20},
      remote_port: 2905,
      routing_context: 2,
      point_code: 200,
      network_indicator: :international
    },
    # MSC STP:server
    # :server STP
    %{
      peer_id: 3,
      name: "Remote_MSC",
      role: :server,
      remote_ip: {10, 0, 0, 30},
      remote_port: 2905,
      routing_context: 3,
      point_code: 300,
      network_indicator: :international
    }
  ]

```

```
},  
  
#   
%{  
  peer_id: 4,  
  name: "Dynamic_Client",  
  role: :server,  
  remote_ip: {10, 0, 0, 40},      #   
  remote_port: 0,                # 0 =   
  routing_context: 4,  
  point_code: 400,  
  network_indicator: :international  
}  
]
```


Configuration

role: server remote_port

- remote_port: 2905
 -
 -
- remote_port: 0
 -
 - IP
 -

Example

```
# 10.5.198.200:2905
%{
  peer_id: 1,
  name: "Strict_Peer",
  role: :server,
  remote_ip: {10, 5, 198, 200},
  remote_port: 2905,
  # ...
}

# 10.5.198.200
%{
  peer_id: 2,
  name: "Flexible_Peer",
  role: :server,
  remote_ip: {10, 5, 198, 200},
  remote_port: 0, #
  # ...
}
```

M2PA

OmniSS7 M3UA M2PA SS7

M2PA

M2PA MTP2 IETF RFC 4165 IP SCTP SS7 MTP3

M3UA M2PA

	M3UA	M2PA
	(ASP/SGW)	
	SS7 IP	
	(ASPUP/ASPAC)	MTP2
		24 BSN/FSN
	SS7 IP STP	
RFC	RFC 4666	RFC 4165

M3UA M2PA

M3UA

M3UA

- **STP**
- SS7 IP
- HLR MSC SMSC STP

- **SGW** (Service Gateway) 提供 IP 地址
- 提供 DNS 服務 / 代理
- 提供 NAT 服務 (RFC 4666)

IP M3UA 與 HLR、MSC、SMSC、VLR 等設備的 STP

IP M2PA 與 HLR、MSC、SMSC、VLR 等設備

M2PA 與 HLR、MSC、SMSC、VLR 等設備

- **STP** 與 **STP** 設備的 STP 協議
- 與 **TDM** 設備的 M2PA 與 SS7 TDM 協議
- 與 **MTP2** 設備的 MTP2 協議
- 與 **M2PA** 設備的 M2PA 協議

IP M2PA 與 HLR、MSC、SMSC、VLR 等設備的 STP - IP M3UA、M2PA 與 STP 協議

IP M2PA 協議

M2PA 與 M3UA 協議的 `protocol` 協議

M2PA 協議

IP M2PA 協議的 `config/runtime.exs` 與 `peers` 與 M3UA 與 M2PA 協議的 `protocol` 協議

M2PA 協議

項目	値	説明
protocol	:m2pa	M2PA 通信プロトコル :m3ua
role	:client :server	クライアント/サーバー
local_port		SCTP 経由で M2PA 通信 3565
remote_port		SCTP 経由で M2PA 通信 3565
point_code		
adjacent_point_code		隣接 M2PA 通信
send_slm		SLTM 送信 SLTM
network_indicator		:international :national -

M2PA 通信ポート **3565** 経由で M3UA 通信 2905

SLTM 通信

SLTM 通信は SLTA 通信と MTP3 通信で M2PA 通信 READY 状態
ITU-T Q.707 規定の SLTM 通信 SLTA 通信

SLTM 通信

send_slm 通信 SLTM 通信

<code>send_slm</code> □	□□
<code>true</code>	□□□□□□□□ READY □□□ SLTM□□□ SLTA
<code>false</code>	□□□□□□□□□□ SLTM□□□ SLTA
□□□□□□□□	□□ Q.707□SCTP □□□□□ SLTM□SCTP □□□□□

□□□□□□ **Q.707**□□

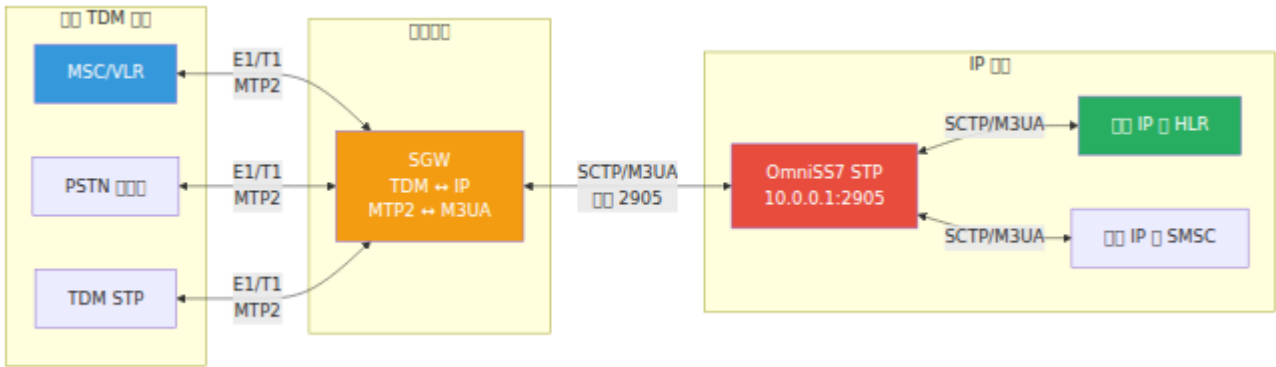
- □□ `initiate_connection: true`□SCTP □□□/□□□□ → □□□□□□□□□□ SLTM
- □□ `initiate_connection: false`□SCTP □□□/□□□□ → □□□□ SLTM

□□□□ **SLTM** □□□

□□□□□□□□ Q.707 □□□□□□□□□□□□□□□□

```
# □□□□□□□□□□□□□□ SLTM□□□□ SCTP □□□□
%{
  peer_id: 100,
  name: "Partner_STP",
  protocol: :m2pa,
  role: :client,
  local_ip: {10, 0, 0, 1},
  local_port: 3565,
  remote_ip: {10, 0, 0, 2},
  remote_port: 3565,
  point_code: 7415,
  adjacent_point_code: 15528,
  network_indicator: :international,
  send_slm: true # □□□□□□□□ SLTM□□□□□□□□ SCTP □□□□
}
```

SLTM □□□



SLTM

READY

SLTM

"waiting for peer to send SLTM"

`send_sltm: true`

SLTM **SLTA**

1. `adjacent_point_code` SLTM DPC
2. `:international` `:national`
- 3.

`adjacent_point_code` STP `sctp_handler.point_code`

M2PA

M2PA

1. **Down** -
2. **Alignment** - 1
3. **Proving** - 2
4. **Ready** -

□□□□□□□□□□□□□□□□□□□□

□□ **Web UI** □□ **M2PA** □□□

Web UI □ □□ □□□□□ M2PA □□□□□□□□□□

1. □□ □□□□□
2. □□ "□□□□" □□□
3. □□ "□□□□□□□"
4. □□ "M2PA (RFC 4165)" □□□□□□□□
5. □□ □□□□□□
 - □□□□□□□□□□□□□□
 - □□□□M2PA
 - □□□□□□□□□□
 - □□□□□□ PC□
 - □□/□□ IP □□
 - □□/□□□□□□□□ M2PA □ 3565□
 - □□□□□□□□□□□□
6. □□ "□□□□□□"

□□□□□□□□□□□□□□□□□□

- □□ - M3UA □□□
- □□ - M2PA □□□

M2PA □□□□

M2PA □□□□ OmniSS7 □□□□□□□□□□

- □□□□□□ M2PA □ M3UA □□
- □□□□□□□□ M2PA □□□□□□□□
- □□□□□□□□ M2PA □ M3UA □□□□□□□□□□□□□□
- □□□□□□□□□□ M2PA □□□□□□□□ M3UA□□□□□□

M2PA

M2PA Prometheus

- `m2pa_messages_sent_total` - MTP3
- `m2pa_messages_received_total` - MTP3
- `m2pa_bytes_sent_total` - M2PA
- `m2pa_bytes_received_total` - M2PA

`link_name` `point_code` `adjacent_pc`

- `m2pa_link_state_changes_total` - DOWN → ALIGNMENT → PROVING → READY
 - `link_name` `from_state` `to_state`

- `m2pa_errors_total` -
 - `decode_error` - M2PA
 - `encode_error` - M2PA
 - `sctp_send_error` - SCTP
 - `link_name` `error_type`

- Prometheus `http://your-server:8080/metrics`
-

M2PA

1. 3565 M2PA
- 2.
3. SCTP IP 132
- 4.

5. `SocketHandler` 클래스를 구현합니다.
6. `SocketHandler` 클래스를 `M2PA` 클래스에 등록합니다.

M2PA 클래스

M2PA 클래스는 `SCTP.SocketHandler` 클래스를 상속받아 `M2PA` 클래스의 `SCTP.SocketHandler` 인터페이스를 구현합니다.

예제

- `SCTP.SocketHandler` 인터페이스
- `M2PA` 클래스

예제

```

# Sctp.SocketHandler
sctp_handler: %{
  enabled: true,
  local_ip: {10, 179, 4, 10},
  local_port: 3565,
  point_code: 100
}

# M2PA
peers: [
  %{
    peer_id: 1,
    name: "M2PA_Link_1",
    protocol: :m2pa,
    role: :client,
    local_ip: {10, 179, 4, 10},
    local_port: 3565,
    remote_ip: {10, 179, 4, 20},
    remote_port: 3565,
    point_code: 100,
    adjacent_point_code: 200
  },
  %{
    peer_id: 2,
    name: "M2PA_Link_2",
    protocol: :m2pa,
    role: :client,
    local_ip: {10, 179, 4, 10},
    local_port: 3565, #
    remote_ip: {10, 179, 4, 30},
    remote_port: 3565,
    point_code: 100,
    adjacent_point_code: 300
  }
]

```

SCTP

- unordered
- PPID 5 M2PA RFC 4165

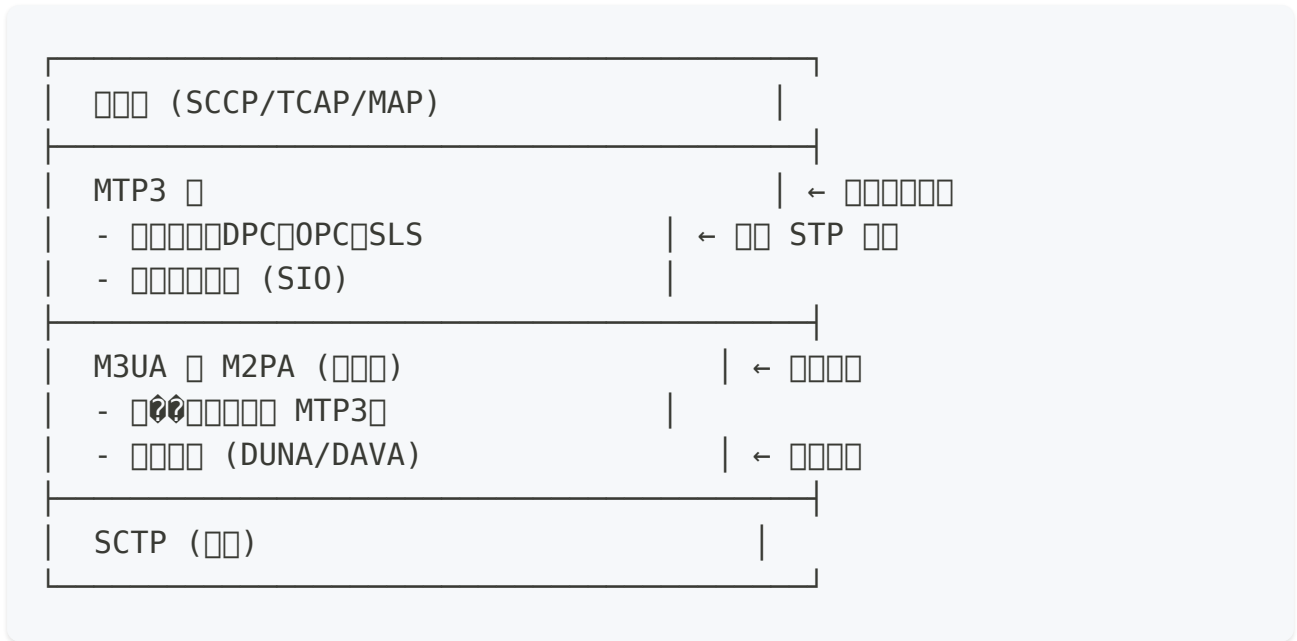
SS7

SS7 MTP3 DPC OPC

SS7

SS7

SS7



SS7

1. MTP3

- MTP3 DPC OPC
- M3UA 528
- M2PA
- STP DPC
-

2. M3UA

- M3UA DUNA DAVA SCON DUPU
-

- 000000000000/000
- 0000000000

STP 00000000

- 00 **M3UA DATA** 000STP 0000000000 5280000 MTP3 0000000 MTP3 0000
0DPC0OPC0SLS00MTP3 00 DPC 00000000
- 00 **M2PA** 0000000STP 0 M2PA 000000000 MTP3 0000000 MTP3 0000000
DPC0
- **M3UA** 000000000000DUNA0DAVA0SCON00 M3UA 00000000000000000000
0000000000

00000000

00000000 `config/runtime.exe` 0

```

config :omniss7,
  m3ua_routes: [
    # 100000000 PC 100 10000 1000000 STP
    %{
      dest_pc: 100, # 10000
      peer_id: 1, # 1000000
      priority: 1, # 1000000000000000000
      network_indicator: :international
      # mask: 14 # 1000000 140000000
    },

    # 100000000 PC 200 10000 20000 HLR
    %{
      dest_pc: 200,
      peer_id: 2,
      priority: 1,
      network_indicator: :international
    },

    # 100000000 PC 300 100000000
    %{
      dest_pc: 300,
      peer_id: 3, # 10000
      priority: 1,
      network_indicator: :international
    },
    %{
      dest_pc: 300,
      peer_id: 4, # 1000000000000000000
      priority: 2,
      network_indicator: :international
    }
  ]

```

mask 14 mask

10000

1. STP M3UA DATA M2PA
2. STP M3UA M2PA **MTP3**
3. STP MTP3 **(DPC)**

4. 00000000 DPC000000
5. 000000000000 000000000000000000 00000000
6. 0 MTP3 00000 M3UA DATA 0 M2PA 0000000000000000
7. 000000000000
8. 00000000000000000000000000000000

0000

000 14 00000 0-163830000000000000000000 /14 00000000000 0000 0000 000
0 0000

0000

0000000000 PC 000000 DPC 00000000 00000000000000000000000000000000

000000

階	階数	階名
/14	1 PC	1階
/13	2 PCs	2階
/12	4 PCs	3階
/11	8 PCs	4階
/10	16 PCs	5階
/9	32 PCs	6階
/8	64 PCs	7階
/7	128 PCs	8階
/6	256 PCs	9階
/5	512 PCs	10階
/4	1,024 PCs	11階
/3	2,048 PCs	12階
/2	4,096 PCs	13階
/1	8,192 PCs	14階
/0	16,384 PCs	15階

階数

階名 mask 階数 階名

階名 1階

```

# 10.0.0.0/24 PC
%{
  dest_pc: 1000,
  peer_id: 1,
  priority: 1,
  network_indicator: :international
}
# 10.0.0.14 - 10.0.0.1000

# 10.0.0.0/24
%{
  dest_pc: 1000,
  peer_id: 1,
  priority: 1,
  mask: 14, # 10.0.0.14
  network_indicator: :international
}
# 10.0.0.1000 PC 1000

```

2. 10.0.0.0/24

```

%{
  dest_pc: 1000,
  peer_id: 2,
  priority: 1,
  mask: 12, # 10.0.0.0/12
  network_indicator: :international
}
# 10.0.0.1000-1001-1002-1003

```

3. 10.0.0.0/24

```
%{
  dest_pc: 1000,
  peer_id: 3,
  priority: 1,
  mask: 8, # 64 PCs
  network_indicator: :international
}
# PC 1000-106364
```

4

```
%{
  dest_pc: 0,
  peer_id: 4,
  priority: 10, # 
  mask: 0, # PCs
  network_indicator: :international
}
# 0-16383
# /
```

```

config :omniss7,
  m3ua_routes: [
    # PC 1000
    %{
      dest_pc: 1000,
      peer_id: 1,
      priority: 1,
      network_indicator: :international
      # 14
    },

    # PCs 1000-1063
    %{
      dest_pc: 1000,
      peer_id: 2,
      priority: 1,
      mask: 8, # 64 PCs
      network_indicator: :international
    },

    # / PCs
    %{
      dest_pc: 0,
      peer_id: 3,
      priority: 10, #
      mask: 0, # PCs
      network_indicator: :international
    }
  ]

```

DPC 1000

1. /14 PC 1000 -
2. /8 PC 1000-1063 -
3. /0 PCs -

DPC 1015

1. /14 PC 1000
2. /8 PC 1000-1063 -
3. /0 PCs -

DPC 5000

1. /14
2. /8
3. /0 PCs -

1. mask /14
2. /14 mask: 14
3. /0 /13
4. /0
5. ◆◆
- 6.

(GT)

IMS! / NAT

□□□□

- □□ GT □□□□ `config/runtime.exs` □□□ `enable_gt_routing: true`

GT 配置

```
config :omniss7,
  # 启用 GT 路由
  enable_gt_routing: true,

  m3ua_gt_routes: [
    # 44 路由 1
    %{
      gt_prefix: "44",           # 44
      peer_id: 1,               # 1
      priority: 1,             # 1
      description: "44"        # 44
    },

    # 1 路由 2
    %{
      gt_prefix: "1",
      peer_id: 2,
      priority: 1,
      description: "1"
    },

    # 447 路由 3
    %{
      gt_prefix: "447",        # 447
      peer_id: 3,
      priority: 1,
      description: "447"
    },

    # SSN 路由 4
    %{
      gt_prefix: "555",
      source_ssn: 8,          # 8 SSN = 8 SMSC
      peer_id: 4,
      dest_ssn: 6,           # 6 SSN 6 HLR
      priority: 1,
      description: "61 61 SMS 61"
    }
  ]
]
```

GT □□□□

GT □□□□□□□□□□□□□□

SCCP

GT SSN TT NPI
NAI

GT
?

GT + SSN + TT + NPI
+ NAI

?

- ：
1. GT
 2. SSN >
 3. TT >
 4. NPI >
 5. NAI >
 - 6.

- 消息源 source_ssn 为 SCCP 消息 SSN 标识符
- 消息源 source_ssn 为 nil 标识符 SSN 标识符

3. TT/NPI/NAI 标识符

- 消息源 source_tt 消息源 source_npi 消息源 source_nai 标识符
- nil 标识符

4. 消息源

- 消息源标识符
- 消息源 GT 标识符 → SSN → TT → NPI → NAI → 消息源

5. 消息目的地

- 消息源 dest_ssn 消息源 dest_tt 消息源 dest_npi 消息源 dest_nai STP 标识符

- 00000000000000000000

6. 00000000

- 00000000 GT 000STP 00000000 DPC 000000

00 GT 0000000000NPI 0 NAI

00 GT 000 SSN 0000STP 000000 SCCP 0000000000000000

- 0000 (**TT**)0000000000000000
- 00000000 (**NPI**)000000000000ISDN00000000
- 00000000 (**NAI**)000000000000000000000000

0000000000

000000000000000000000000

- source_tt 00000000000000000000
- source_npi 0000000000000000000000
- source_nai 000000000000000000000000
- nil 0 = 00000000000000000000

000000000000

000000000000000000000000

- dest_tt 000000000000
- dest_npi 0000000000000000
- dest_nai 0000000000000000
- nil 0 = 000000000000

00000000

00000000000000000000000000000000

1. 00 GT 0000
2. 000 SSN 000000 SSN
3. 000 TT 000000 TT
4. 000 NPI 000000 NPI
5. 000 NAI 000000 NAI
6. 00000000

0000

```

config :omniss7,
  enable_gt_routing: true,

m3ua_gt_routes: [
  # 1
  %{
    gt_prefix: "44",
    peer_id: 1,
    source_tt: 0,      # TT=0
    dest_tt: 3,       # TT=3
    priority: 1,
    description: "TT 0→3"
  },

  # 2 NPI NAI
  %{
    gt_prefix: "1",
    peer_id: 2,
    source_npi: 1,    # NPI=1ISDN
    source_nai: 4,    # NAI=4
    dest_nai: 3,      # NAI=3
    priority: 1,
    description: "→ NAI"
  },

  # 3 SSN
  %{
    gt_prefix: "33",
    source_ssn: 8,    # SMSC
    source_tt: 0,     # TT=0
    dest_ssn: 6,      # SSN HLR
    dest_tt: 2,       # TT=2
    dest_npi: 1,      # NPI=1ISDN
    dest_nai: 4,      # NAI=4
    peer_id: 3,
    priority: 1,
    description: "SMS"
  },

  # 4 TT NPI
  %{
    gt_prefix: "49",
    source_tt: nil,   # TT
  }
]

```

```
    source_npi: 6,      # 00 NPI=60000
    dest_npi: 1,       # 000 NPI=10ISDN
    peer_id: 4,
    priority: 1,
    description: "0000000000"
  }
]
```

00 TT/NPI/NAI 0

0000 (TT)0

- 0 = 00
- 1 = 00
- 2 = 00
- 3 = 0000

0000000 (NPI)0

- 0 = 00
- 1 = ISDN/000E.1640
- 3 = 000X.1210
- 4 = 000F.690
- 6 = 00000E.2120

0000000 (NAI)0

- 0 = 00
- 1 = 0000
- 2 = 00000000
- 3 = 000000
- 4 = 0000

0000000

00000000

- GT: "447712345678"

- SSN: 8
- TT: 0
- NPI: 1
- NAI: 4

□□□□□□□□

```
# □□ A□□□□ TT
%{gt_prefix: "447", peer_id: 1, priority: 1}

# □□ B□□□ TT
%{gt_prefix: "447", source_tt: 0, peer_id: 2, priority: 1}

# □□ C□□□□ TT + NPI
%{gt_prefix: "447", source_tt: 0, source_npi: 1, peer_id: 3,
priority: 1}
```

□□□□□ C □□□□□□□□□□ GT + TT + NPI□

□□□□□□□□□□□ C □ dest_tt □ dest_npi □ dest_nai □□□□□□

GT 規則

GT	SSN	TT	NPI	NAI	規則	結果
447712345678	6	-	-	-	"447" → GT 3	447712345678
441234567890	6	-	-	-	"44" → GT 1	441234567890
12125551234	6	-	-	-	"1" → GT 2	12125551234
555881234567	8	-	-	-	"555" → SSN 8 → GT 4	GT + SSN SSN 6
555881234567	6	-	-	-	"555" → SSN X → GT X	GT SSN
441234567890	6	0	-	-	"44" → TT=0 → GT 1	GT + TT TT 3
12125551234	8	0	1	4	"1" → TT=0 NPI=1 NAI=4	GT+TT+NPI+NAI

TT/NPI/NAI 規則

1. TT

- SSN 8桁
- SSN 6桁
- SSN 8桁 TT=0 3桁 → SSN 8桁 TT=1

2. NPI/NAI

- SSN 8桁

- 國際號碼 NPI=6 PSTN NPI=1

3. 國際號碼

- 國際號碼 NAI
- 國際號碼 NAI=4 國際號碼 NAI=3

4. 國際號碼

- 國際號碼
- 國際號碼 TT=0 國際號碼 A TT=2 國際號碼 B

5. 國際號碼

- 國際號碼
- STP 國際號碼

國際號碼

國際號碼

國際號碼

國際號碼

國際號碼 enabled 國際號碼

```
config :omniss7,  
  m3ua_routes: [  
    # 國際號碼  
    %{\br/>      dest_pc: 100,  
      peer_id: 1,  
      priority: 1,  
      network_indicator: :international,  
      enabled: true # 國際號碼
```

USSD 開発

← 目次

OmniSS7 **USSD** 開発 SS7/MAP USSD HTTP/JSON 開発
HTTP USSD 開発

目次

1. 概要
2. 環境
3. USSD 概要
4. 開発
5. HTTP 開発
6. USSD (API)
7. 開発
8. 開発
9. 開発
10. 開発
11. 開発

目次

USSD 開発 USSD 開発

- 開発 (SS7) — 開発 *100# 開発 MAP `processUnstructuredSS-Request` 開発 59 開発 HTTP 開発 SS7 開発
- 開発 (HTTP) — 開発 REST API 開発 USSD 開発 SS7 開発 MAP `unstructuredSS-Request` 開発 60 開発

開発 開発 — 開発

□□□□

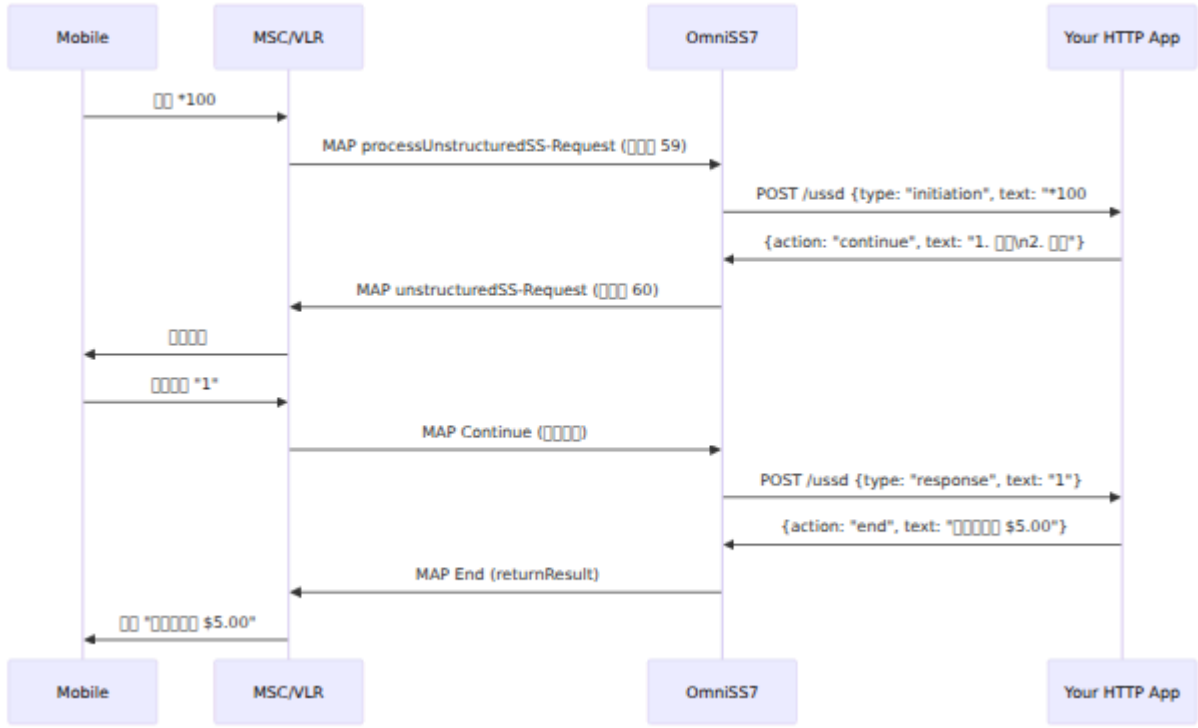
□□	□
□□	□□□□ HTTP POST
□□	GSM 7 □□□□□□ (DCS 0x0F) □□ 3GPP TS 23.038
□□□□□□	182 □□□□□□□□
□□□□	□□□□□□□□□□ UUID
□□	□□□□ SS7 □□□
□□	□□□□□□□□□□ URL

3GPP □□

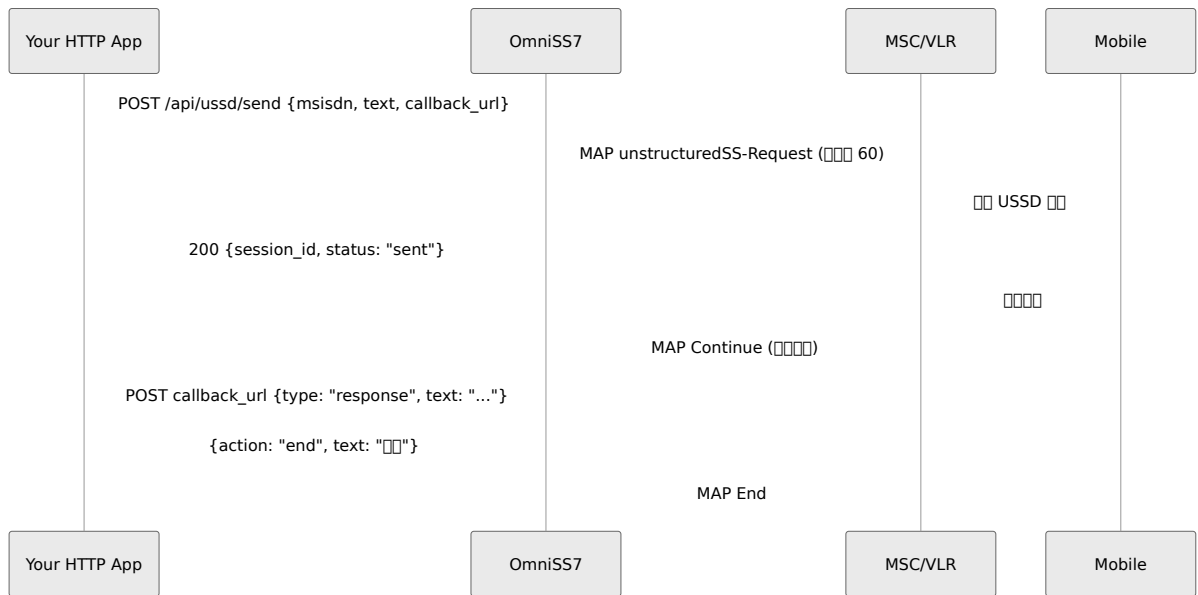
□□	□□□
3GPP TS 23.090	USSD □□□□ — □□□□□
3GPP TS 24.090	USSD □□□□ — □□□□
3GPP TS 29.002	MAP □□ — USSD-Arg, USSD-Res, □□□ 59/60/61
3GPP TS 23.038	GSM 7 □□□□□□□□□□□□□□

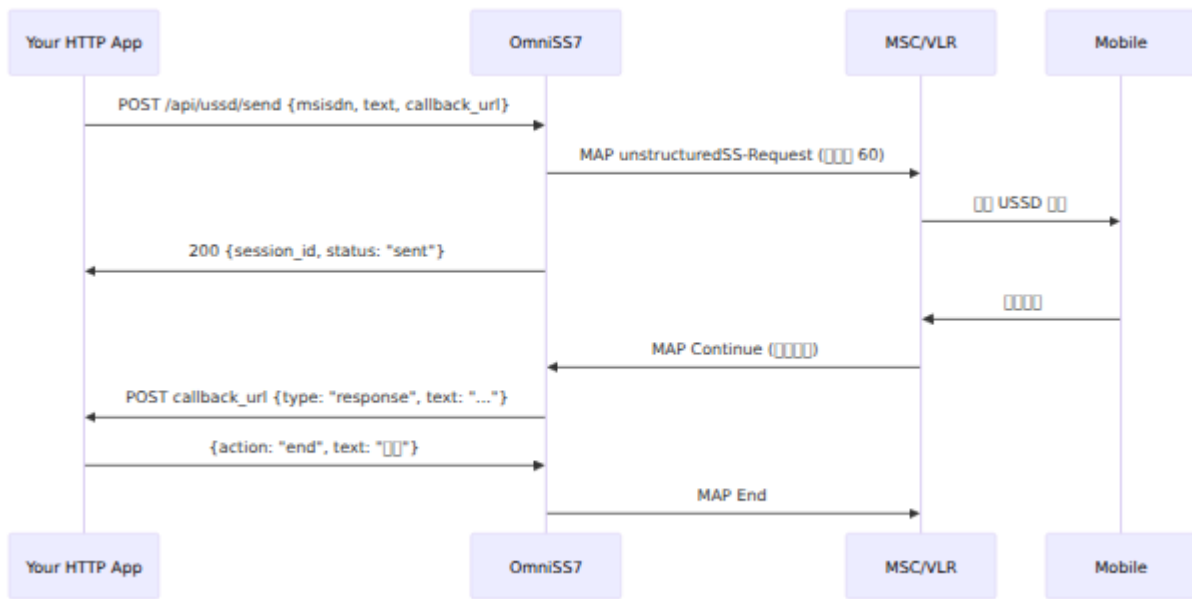
□□

□□□□□□ (□□)



□□□□□□ (□□□□)





USSD

USSD MAP

```
config :omniss7,
  map_client_enabled: true,
  ussd_gateway_enabled: true
```

M3UA MAP M3UA



USSD 配置

```
config :omniss7,  
  ussd_gateway_enabled: true,  
  ussd_gateway: %{  
    # 配置 - 路由  
    routes: [  
      %{pattern: "*100", url: "http://balance-app:9000/ussd"},  
      %{pattern: "*200", url: "http://topup-app:9000/ussd"},  
      %{pattern: "*", url: "http://default-app:9000/ussd"}  
    ],  
  
    # 会话  
    session_timeout_ms: 180_000, # 会话超时3000  
    turn_timeout_ms: 30_000, # 会话超时3000  
  
    # HTTP 配置  
    http_timeout_ms: 5_000, # 会话超时 HTTP POST 5000  
  
    # 文本  
    max_text_length: 182 # GSM 7 位编码  
  }  
}
```

配置

配置名	タイプ	デフォルト値	説明
<code>ussd_gateway_enabled</code>	ブール値	<code>false</code>	USSD を有効にするかどうか
<code>ussd_gateway.routes</code>	配列	<code>[]</code>	ルート定義の配列。各要素は、 <code>pattern</code> 、 <code>url</code> 、 <code>URL</code> を含むオブジェクト
<code>ussd_gateway.session_timeout_ms</code>	整数	<code>180_000</code>	セッションタイムアウト時間 (ms)
<code>ussd_gateway.turn_timeout_ms</code>	整数	<code>30_000</code>	ターンタイムアウト時間 (ms)
<code>ussd_gateway.http_timeout_ms</code>	整数	<code>5_000</code>	HTTP タイムアウト時間 (ms)
<code>ussd_gateway.max_text_length</code>	整数	<code>182</code>	USSD テキストの最大長

例

`routes` [ルート定義の配列]

項目	形式	長さ	説明
pattern	正規表現	1	任意の文字列を指定する。 "*" は任意の文字列を指定する。
url	URL	1	任意の文字列を指定する。 POST の HTTP の URL。

パターン

任意の文字列を指定する。 "*" は任意の文字列を指定する。 *100#

- "*100" は任意の文字列を 4 文字指定する。
- "*10" は任意の文字列を 3 文字指定する。
- "*" は任意の文字列を 1 文字指定する。

任意の文字列を指定する。 MAP は任意の文字列を指定する。

HTTP パターン

任意の文字列を指定する。 HTTP POST の任意の JSON を指定する。

任意の文字列を指定する。

Content-Type: application/json

任意の文字列を指定する。

```
{
  "session_id": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
  "msisdn": "+254712345678",
  "type": "initiation",
  "text": "*100#",
  "turn": 1
}
```

任意の文字列を指定する。

```
{
  "session_id": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
  "msisdn": "+254712345678",
  "type": "response",
  "text": "1",
  "turn": 2
}
```

□□□□

□□	□□	□□
session_id	□□□	□□□□□ UUID□□□□ USSD □□□□□□□□□□□□□□□□
msisdn	□□□	□□□□ MSISDN□□□□□ MAP □□□□□□□□□□□□□□□□
type	□□□	□□□□□□□ "initiation"□□□□□□□□□□ "response"□
text	□□□	□□□□□□□□□□□□□□□□ *100#□□□□□□□□□□□□□□□□ 1□□
turn	□□	□ 1 □□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□

□□□□□□□□ JSON □□□□□□□□ action □ text□

□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
{
  "action": "continue",
  "text": "1. □□\n2. □□\n3. □□"
}
```

□□□□□□□□□□□□□□□□□□□□□□□□□□□□

```
{
  "action": "end",
  "text": "☑️ $5.00"
}
```

☑️

Field	Type	Required	Description
action	String	Optional	"continue" to continue the session, "end" to end the session
text	String	Optional	Message text, max_text_length 182, can contain \n

☑️ USSD (☑️ API)

☑️ USSD ☑️

☑️

POST /api/ussd/send

☑️

```
{
  "msisdn": "+254712345678",
  "text": "☑️ 1 ☑️",
  "callback_url": "http://billing-app:9000/ussd"
}
```

Request

Field	Type	Required	Description
<code>msisdn</code>	String	Yes	MSISDN (Mobile Station International Subscriber Directory Number)
<code>text</code>	String	Yes	USSD message content (GSM 7-bit encoding)
<code>callback_url</code>	String	No	URL to call back after the message is sent

Response

200 OK:

```
{
  "session_id": "xyz-789-abc-123",
  "status": "sent"
}
```

Errors

HTTP Status	Error Message	Description
400	<code>{"error": "invalid request", "required": ["msisdn", "text", "callback_url"]}</code>	Missing required fields
400	<code>{"error": "gsm7_encode_failed", ...}</code>	USSD message content (GSM 7-bit encoding) failed
500	<code>{"error": "send_failed", ...}</code>	M3UA (Mobile Switching Center) error
503	<code>{"error": "USSD gateway not enabled"}</code>	<code>ussd_gateway_enabled</code> is <code>false</code>

cURL

```
curl -X POST http://localhost:8080/api/uszd/send \  
-H "Content-Type: application/json" \  
-d '{  
  "msisdn": "+254712345678",  
  "text": "XXXXXXXXXXXXXXXX 1 XXXXX",  
  "callback_url": "http://billing-app:9000/uszd"  
}'
```

XXXXXXXXXX

USSD XXXXXXXXXXXXXXXX session_id XXXXX

XXXXX



MAP XXX 59

Begin

XXXX "continue"

WaitingForReply

XXXX "end" XXXXX/XX

XXXX "continue"
XXXX (MAP Continue)

XXXX XXXX XXXXX

Processing

XXXX "end"
XXXXXX/XX



MAP End

MAP End 消息的响应消息体中，包含 "end" 消息体。

MAP Continue

MAP Continue 消息的响应消息体中，包含 "continue" 消息体。

1. 在 `UssdGateway.Registry` 中注册 **GenServer**
2. 在 `60` 秒内发送 `unstructuredSS-Request` 消息，MAP Continue 消息
3. 在 `turn_timeout_ms` 中设置超时时间
4. 在 `turn_timeout_ms` 中设置超时时间
5. 在 `"end"` 消息体中设置超时时间

MAP Error

消息	超时	说明
MAP Error	30 秒	在 30 秒内发送 MAP Error 消息
MAP Error	3 秒	在 3 秒内发送 MAP Error 消息
HTTP 消息	5 秒	在 5 秒内发送 MAP Error 消息

MAP Error

MAP Error 消息的响应消息体中，包含 MAP Error 消息体。

項目	説明	MAP 状態
HTTP ステータス 5xx	エラー発生 MAP End	34 (systemFailure)
JSON 解析エラー	エラー発生 MAP End	34 (systemFailure)
USSD 送信 max_text_length	送信完了	N/A (不明)
エラー発生	エラー発生 MAP End	34 (systemFailure)
エラー発生	エラー発生 MAP End	34 (systemFailure)
GenServer 停止	停止発生	N/A (不明)
USSD 送信エラー	送信エラー	21 (facilityNotSupported)

監視

USSD 監視は `/metrics` エンドポイントで 8080 ポートで Prometheus が提供されています。

USSD 監視

メトリック: `ussd_requests_total`

単位: 回

ラベル: 送信方向 USSD 番号

オプション:

- `direction` — `"inbound"` (受信) / `"outbound"` (送信)

📌: `ussd_active_sessions`

📌: 📌📌

📌: 📌📌📌📌 USSD 📌📌📌

📌: `map_request_duration_milliseconds`

📌: 📌📌

📌: USSD 📌📌📌📌📌📌📌📌📌

📌:

- `operation` — `"ussd_send"` 📌📌📌📌📌

📌 Prometheus 📌

```
# 📌📌 USSD 📌📌  
rate(ussd_requests_total[5m])  
  
# 📌📌  
ussd_active_sessions  
  
# 📌 USSD 📌 (p95)  
histogram_quantile(0.95,  
rate(map_request_duration_milliseconds_bucket{operation="ussd_send"}  
[5m]))
```



Python (Flask)

```
from flask import Flask, request, jsonify

app = Flask(__name__)
sessions = {}

@app.route('/ussd', methods=['POST'])
def ussd():
    data = request.json
    session_id = data['session_id']
    text = data['text']
    turn = data['turn']

    if data['type'] == 'initiation':
        sessions[session_id] = {'state': 'main_menu'}
        return jsonify({
            'action': 'continue',
            'text': '1. 2. 3. '
        })

    state = sessions.get(session_id, {}).get('state')

    if state == 'main_menu':
        if text == '1':
            del sessions[session_id]
            return jsonify({
                'action': 'end',
                'text': '$5.00'
            })
        elif text == '2':
            sessions[session_id]['state'] = 'buy_airtime'
            return jsonify({
                'action': 'continue',
                'text': ''
            })
        else:
            del sessions[session_id]
            return jsonify({
                'action': 'end',
```

```
        'text': '██████████0000'
    })

elif state == 'buy_airtime':
    del sessions[session_id]
    return jsonify({
        'action': 'end',
        'text': f'██████ ${text} ██████████'
    })

return jsonify({'action': 'end', 'text': '████████'})

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=9000)
```

Node.js (Express)

```
const express = require('express');
const app = express();
app.use(express.json());

const sessions = new Map();

app.post('/ussd', (req, res) => {
  const { session_id, text, type } = req.body;

  if (type === 'initiation') {
    sessions.set(session_id, { state: 'main_menu' });
    return res.json({
      action: 'continue',
      text: '□□□□\n1. □□□□\n2. □□□□'
    });
  }

  const session = sessions.get(session_id);
  if (!session) {
    return res.json({ action: 'end', text: '□□□□□□' });
  }

  if (session.state === 'main_menu' && text === '1') {
    sessions.delete(session_id);
    return res.json({ action: 'end', text: '□□□□□ $5.00' });
  }

  sessions.delete(session_id);
  return res.json({ action: 'end', text: '□□□' });
});

app.listen(9000, () => console.log('USSD □□□□□ 9000 □□□'));
```

□□□□

USSD □□❓❓□ "□□□□□"

□□: □□□□□□□□□□□□□□□□

□□□□:

- `ussd_gateway_enabled` □ `false`
- □□□□□□□□□□□□
- M3UA □□□□□

□□□□:

1. □□□□□ `ussd_gateway_enabled: true`
2. □□□□□□□□□□□□□□□□ * □□□□□
3. □□ Web UI □□ M3UA □□□□□□□□□□□□

□□□□□□□□

□□: □□□□□□ USSD □□□□□□□□□□□□ HTTP POST□

□□□□:

- □□ URL □ OmniSS7 □□□□□
- □□□□□ OmniSS7 □□□ HTTP
- □□□□□□□

□□□□:

1. □□□□□ `curl -v http://your-app:9000/ussd` □ OmniSS7 □□
2. □□□□ HTTP □□□□□□□
3. □□□□□□□□□□□□□□□□

□□□□□□□□

□□: □□□□□ "systemFailure" □□□□□□□□□□□□

原因:

- `turn_timeout_ms` 配置值过小
- `http_timeout_ms` 配置值过小
- OmniSS7 配置值过小

解决:

1. 将 `turn_timeout_ms` 配置为 30 秒
2. 将 `http_timeout_ms` 配置为 30 秒
3. 将 OmniSS7 配置为 30 秒

GSM 7 问题

现象: 返回 `gsm7_encode_failed` 错误码 `/api/usd/send` 返回 400 错误

原因:

- 配置 GSM 7 编码表时使用了非 CJK 字符

解决:

- 将 USSD 内容转换为 GSM 7 编码表支持的 ASCII 字符
- 参考 [3GPP TS 23.038 6.2.1](#) 章节

相关链接

- [API 文档](#) — REST API 接口 `/api/usd/send`
- [MAP 文档](#) — USSD 消息 M3UA 消息
- [配置文档](#) — 配置项
- [监控文档](#) — Web UI 集成 Prometheus 监控

Web UI

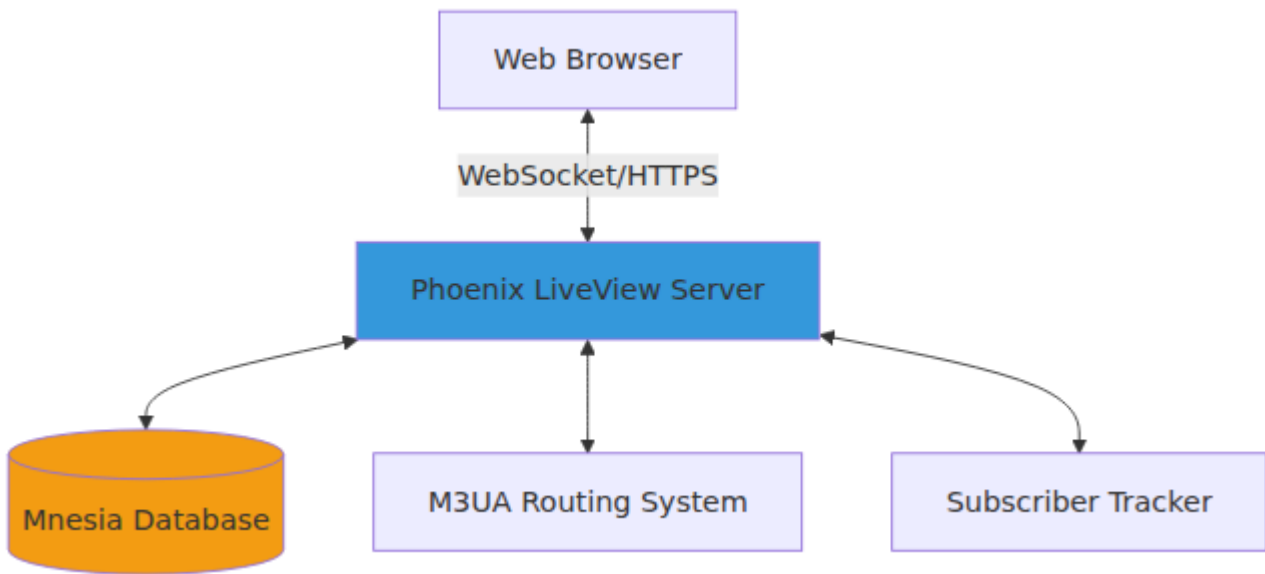
←

OmniSS7 **Web UI** Phoenix LiveView

- 1.
2. **Web UI**
- 3.
- 4.
- 5.
- 6.

OmniSS7 Web UI **Phoenix LiveView**
STP HLR SMS

Web UI



Configuration

- `ssl:` HTTPS
- `ssl:` 443 `config/runtime.exs` `ssl`
- `ssl IP:` 0.0.0.0
- `ssl:` `priv/cert/`

`ssl URL:` `https://[server-ip]:443`

Web UI

Configuration

1. **SSL** `ssl:` `priv/cert/` `ssl` `SSL`
 - `omnitouch.crt` - `ssl`
 - `omnitouch.pem` - `ssl`
2. `ssl`: `mix` `ssl`
3. `ssl:` `ssl` 443 `ssl` `SSL`

□□□□□□□□

□□	STP □□	HLR □□	SMSc □□	□□
SS7 □□	□	□	□	□□□□□ SCCP □□□□
SS7 □□□	□	□	□	□□ MAP □□□□
M3UA	□	□	□	M3UA □□□□
□□	□	□	□	M3UA □□□□□
□□□□	□	□	□	□□□□□□□□
HLR □□	□	□	□	HLR API □□□□□□□□
□□□□	□	□	□	□□□□□□□□□□HLR□
SMSc □□	□	□	□	SMSc API □□□□□□□□
SMSc □□	□	□	□	□□□□□□□□SMSc□
□□□□	□	□	□	□□□□□□□□
□□	□	□	□	□□□□□□

□□□□□

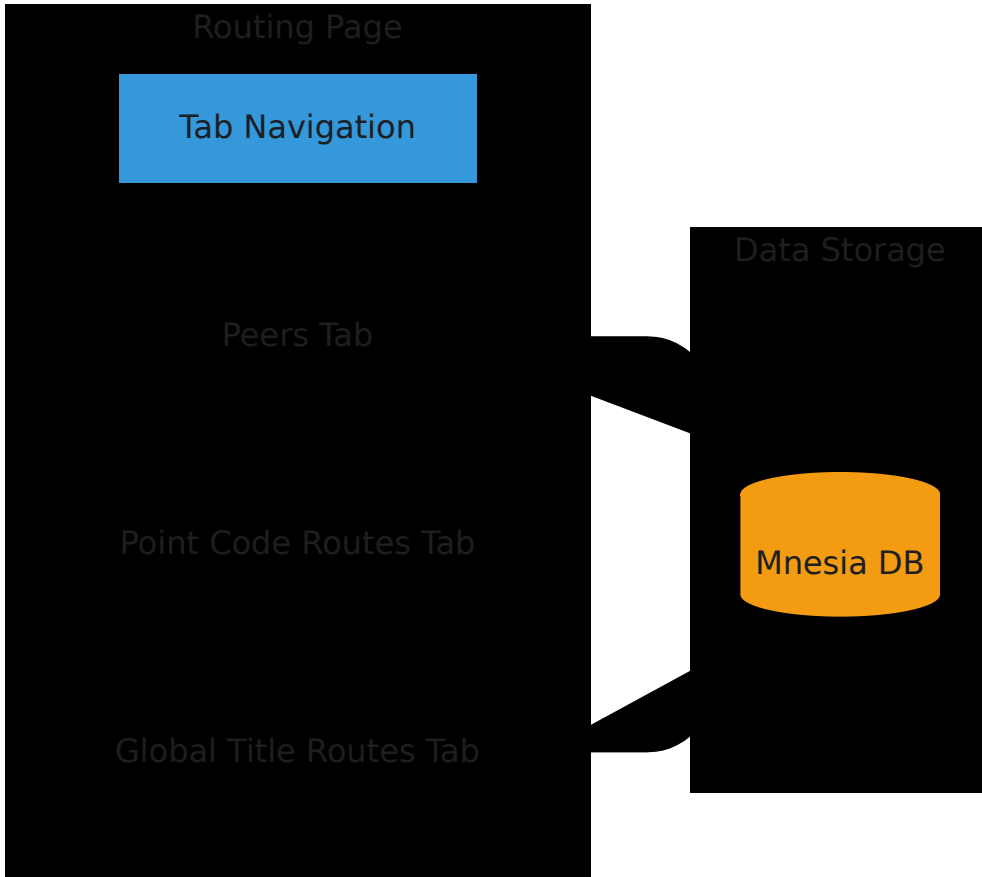
□□: /routing

□□: STP, SMSc

□□□□: □ 5 □

□□□□□□□□□□□□□□□□□□□□ M3UA □□□□

□□□□



Peers □□□

□□ M3UA □□□□□□ STP□HLR□MSC□SMSC□□

□□□□

項目	説明	値
ID	識別子	1
名前	名前	"STP_West"
タイプ	タイプ	client, server, stp
ポート	ポート SS7	100
アドレス	IP アドレス	10.0.0.10:2905
ステータス	ステータス	active, aspup, down
コメント	コメント	-

手順

1. Peers を確認

2. 設定を確認

- ID: 1
- 名前: "STP_West"
- タイプ: client, server, stp
- ポート: SS7 100
- IP アドレス: 10.0.0.10
- ステータス: 0
- IP アドレス: 10.0.0.10
- ポート: 2905
- ステータス: M3UA ID
- 名前: international, national

3. "Add Peer"

コマンド: Mnesia

確認

1. 点击按钮 "Edit" 后
2. 显示输入框
3. 点击 "Update Peer"

注意: 输入框 ID 为 `peer_id`

删除

1. 点击按钮 "Delete" 后
2. 显示确认对话框

数据表

状态	操作	描述
<code>active</code>	已启用	正常运行的节点
<code>aspup</code>	正在启动	ASP 正在启动
<code>down</code>	已禁用	节点已下线

数据表

数据表

数据表

項目	設定内容	値
PC	zone.area.id	1.2.3 (100)
	PC の	/14 (), /8 ()
ID		1
		"STP_West"
	1 =	1
		international
	/	-

1. "Point Code Routes"

2.

- zone.area.id 1.2.3 0-16383
- : /14
- ID:
- : 1 =
- : international national

3. "Add Route"

:

- 3-8-3** : zone.area.id 1.2.3
- : 0-16383 1100

14 0-16383

IP	Number of PCs	IP Range
/14	1 (PC)	1000
/13	2 PCs	1000-1001
/8	64 PCs	1000-1063
/0	16,384 PCs	0-65535

IP:

- PC 1000 /14 → PC 1000
- PC 1000 /8 → PC 1000-1063 (64 PCs)
- PC 0 /0 → 0-65535

IP

IP

IP

IP SCCP IP

IP: IP

```
config :omniss7,
  enable_gt_routing: true
```

IP

Field	Description	Value
GT	Global Title (GT) =	"1234", ""
SSN	SSN	6 (HLR), any
ID	ID	1
		"HLR_West (1)"
SSN	SSN	6, preserve
		1
		"US numbers"
		-

Configuration

1. "Global Title Routes"
2.
 - o **GT**: "1234"
 - o **SSN**: - SSN
 - o **ID**: 1
 - o **SSN**: - SSN
 - o : 1 =
 - o : "
3. "Add Route"

GT: GT

SSN

SSN

SSN	名前
6	HLR (データベース)
7	VLR (データベース)
8	MSC (データベース)
9	EIR (データベース)
10	AUC (データベース)
142	RANAP
145	gsmSCF (データベース)
146	SGSN

SSN 一覧

- **1 SSN:** データベース SSN
- **2 SSN:** データベース SSN
 - 1 = データベース SSN
 - 2 = データベース SSN

例: 1 SSN=6 HLR データベース SSN=7 VLR

インストール

データベース **Mnesia** をインストール

インストール

1. **Web UI** 例: `http://localhost:8080` Mnesia
2. データベース: データベース Mnesia
3. **Runtime.exs** 例: `config/runtime.exs` データベース Mnesia

Table 1

Field	Label	Value
IMSI	IMSI	"50557123456789"
VLR	VLR GT	"555123155"
MSC	MSC GT	"555123155"
Timestamp	UpdateLocation	"2025-10-25 14:23:45 UTC"
Duration		"2h 15m 34s"

Table 2

Table 2: Summary of VLR and MSC

- **VLRs**: VLR
- **MSCs**: MSC

Table 3

Table 3: Summary of VLR and MSC

Table 3: Summary of VLR and MSC

Table 3: Summary of VLR and MSC

Table 4

Table 4: Summary of VLR and MSC

Table 1

Field	Description	Value
MSISDN	MSISDN	"15551234567"
IMSI	IMSI	"001010123456789"
HLR GT	HLR GT (alertServiceCenter)	"15551111111"
MT-FSM	MT-FSM	5
MO-FSM	MO-FSM	2
Status	Active or Failed	● Active
Timestamp	Timestamp	"2025-10-30 14:23:45 UTC"
Duration	Duration	"15m 34s"

Table 2

- **Active** (●): alertServiceCenter
- **Failed** (○): SRI-for-SM

Table 3

Table 3 Headers

- Field: Description
- Field: Description
- Field: Description
- Field **HLRs**: HLR

□□□□

□□□□: □□□□□□□□□□

□□□□□□: □□□□□□□□□□

□□: □□□□□□□□□□□□□□

□□:

- □□□□□□□□□□□□
- □□□□□□□□□□
- □□□□ HLR □□□□□□

□□□□□

□□□□□□□□□□□□

- □□□□□: □ SRI-for-SM □□□□□ MT-FSM □□□
- □□□□□: □□□□□□ MO-FSM □□□

□□□□

□□□ **2** □ □□□□□□□□□□□□□□□□□

□□□□

□□□□□

□□□ Web UI □□□□□□□□□□/□□□□□□□□□□□□□□

1. □□□□□□□□□□ (Ctrl+F / Cmd+F)
2. □□□□□□□□□□□□ GT □□

□□□□

□□□□□□□□□□

1. □□ **1**: □□ REST API □□□□□□
2. □□ **2**: □□ config/runtime.exs □□□□□□
3. □□ **3**: □□ Web UI □□□□□□□□

□□/□□

□□: Web UI □□□□□□□□□□□□□□□□

- □□□ Mnesia □□□□□□
- □ config/runtime.exs □□□

□□□□□□

1. **Mnesia**: □□ Mnesia.{node_name}/ □□
2. □□: □□□□ config/runtime.exs

□□□□□□□

□□□□□□□□□□□□

□□	□□□□	□□
□□□□	5 □	□□□□□□□□
□□□□	2 □	□□□□□□□□
M3UA □□	□□□□□□	□□□□□□

WebSocket □□: □□□□□□□ Phoenix LiveView WebSocket □□□□□□□□□□

□□□□: □□ WebSocket □□□□□□□□□□□□□□□□

SSL

SSL

1. **HTTPS** : `priv/cert/omnitouch.crt` `.pem`
2. **443**: HTTPS
3. : `iex -S mix`
4. : SSL

SSL

1. **Mnesia** : `mnesia_storage_type: :disc_copies`
2. **Mnesia** : Mnesia
3. : Mnesia

SSL

1. **WebSocket** : WebSocket
 2. :
 3. : (F5)
-

SSL

- **STP** -
 - **HLR** -
 - **API** - REST API
 - -
-

SSL

OmniSS7 Web UI

□ □□□ - □□□□□□□□

□ □□□ - Mnesia □□□□□□□□

□ □□□ **UI** - □□□□□□□□STP/HLR/SMSc□□□□

□ □□□ - □□□□□□□□□□

□ □□□ - □□□□□□□□

□□□□□□□□□□ **API** □□

