

Nokia AirScale 部署

RAN Monitor 部署

前提条件

- 网络就绪
 - 数据库就绪
 - WebLM 就绪
 - 网络就绪
 - 网络就绪
 - 网络就绪
 - 网络就绪
-

部署

部署 RAN Monitor 需要部署 Nokia AirScale 部署包。部署 RAN Monitor 需要部署 Nokia Web Element Manager (WebLM) 部署包。

部署 RAN Monitor (PMCADM) 需要部署 Nokia AirScale 部署包。

部署选项

部署 RAN Monitor 需要部署 **Omnitouch** 部署包。

- 部署 RAN Monitor
- 部署 RAN Monitor
- 部署 RAN Monitor
- 部署 RAN Monitor

部署 **Omnitouch** 需要部署 **Omnitouch** 部署包。

Table of Contents

Table of Contents

- Introduction - Introduction
- Overview - Overview
- **RAN Monitor** - RAN Monitor IP
- Configuration - Nokia AirScale

Table of Contents

Item	Value	Default
RAN Monitor IP	RAN Monitor IP	10.179.2.139
RAN Monitor Port	Port (Port: 9076)	9076
Timeout	Timeout	60 s

WebLM

1. Web Element Manager

1. Introduction
2. Configuration

```
http://<base-station-ip>/
```


Table 2: Configuration Parameters

Table Structure 1: Configuration parameters for the RAN Monitor.

Table

Parameter	Value	Description
TLS <code>tls.enabled</code>	enabled	RSA (used for TLS)
<code>tls.hostname</code>	RAN Monitor IP 10.179.2.139	RAN Monitor IP (10.179.2.139)
<code>tls.port</code>	RAN Monitor <code>tls.port</code>	9076 (port)
<code>tls.timeout</code>	60s	60s (timeout)
<code>tls.verify</code>	false	false (verify)

Table

項目	項目	項目
SDL 項目	項目	1
SDL 項目	項目	12345678
項目	項目 0 項目	false (項目)

項目 3 項目

項目

1. 項目 - 項目 IP 項目
2. 項目
 - 項目 **Create Plan** 項目
 - 項目
 - 項目 ID
3. 項目
 - 項目 **Validate Plan** 項目
 - 項目 ID
 - 項目
 - 項目
4. 項目
 - 項目 **Activate Plan** 項目
 - 項目 ID
 - 項目
 - 項目

項目 XML 項目

項目 XML 項目 PMCADM 項目

```
<managedObject class="com.nokia.srbts.mnl:PMCADM" distName="MRBTS-132/MNL-1/MNLENT-1/PMCADM-1" version="MNL25R1_2420_110" operation="create">
  <p name="act3gppXmlEnrichment">false</p>
  <p name="reportingIntervalPm">5min</p>
  <p name="sdlMaxUploadFileNumber">1</p>
  <p name="sdlPrimaryDestIp">10.179.2.139</p>
  <list name="rTPmCollEntity">
    <item>
      <p name="certTypeForTlsAuth">RSA</p>
      <p name="rTPmCollEntityHost">10.179.2.139</p>
      <p name="rTPmCollEntityPortNum">9076</p>
      <p name="rTPmCollInterval">60s</p>
      <p name="tlsEnabled">>false</p>
    </item>
  </list>
</managedObject>
```

XML 属性

- `rTPmCollEntityHost` - RAN Monitor IP
- `rTPmCollEntityPortNum` - 9076 (webhook)
- `rTPmCollInterval` - (60s)
- `tlsEnabled` - false
- `sdlPrimaryDestIp` - RAN Monitor IP

10.179.2.139 RAN Monitor IP ID MRBTS-132

PMCADM-1

PMCADM-1

PMCADM (PMCADM) 属性

属性

- 属性

- 000000000000
- 000000000000
- 000000000000

000000000000

00000000000000000000

certTypeForTlsAuth - TLS 00000000

- 000 00 (RSA, DSA, ECDSA)
- 0000 TLS 00000000
- 000 RSA
- 000 00 tlsEnabled = true 000

rTpmCollEntityHost - 00000000

- 000 IP 00 (IPv4 0 IPv6)
- 000 00000000 IP 00
- 000 0
- 000 10.179.2.139
- 000 0000000000000000

rTpmCollEntityPortNum - 00000000

- 000 00 (1-65535)
- 000 00000000 TCP 00
- 000 9076
- 000 000 RAN Monitor 0000

rTpmCollInterval - 0000

- 000 00 (0)
- 000 000000000000
- 000 15s, 30s, 60s, 300s, 900s, 1800s
- 000 60s
- 000 00000 60s000000000 15s

tlsEnabled - TLS

- true/false
- true
- false
- true

sdlMaxUploadFileNumber - SDL

- 1
- 1
- 1
- 1

sdlNonce - SDL

- 8
- SDL
- 12345678
- 12345678

suppressZeroValueCount -

- true/false
- true
- false
- true/false



RAN Monitor

RAN Monitor Web UI

1. RAN Monitor Web UI: `http://<ran-monitor-ip>:4000/`

2. 状态 **eNodeB** 为 连接
3. 状态为 **Connected**
4. 状态为 **Connected** (连接)
5. 状态 **Session** 为 **Active**

状态

- 状态 **Connected** (连接)
- 状态 **Active**
- 状态 为 **IP** 地址
- 状态 为 **Active**

状态为 **Pending**

状态为 **Pending**

□□□□

- □□ ID □□□□□□
- RAN Monitor □□□□□□
- □□□□

□□□□□□□□

□□□□ RAN Monitor□

□□□□

- PMCADM 0000 IP 000000
- 0000000
- 00000000 8080
- RAN Monitor 000000

00000000

00 **InfluxDB** 000

1. 0 RAN Monitor Web UI 000000 **InfluxDB** 00
2. 00000000000000
3. 00 0000 0000000000
4. 00 0000 0000000000

000000

- 000000 00000000
- 000 000000
- 000 0000000000000000 0

□□□□□□

1. □□□ □□□□□□ □□
2. □□□□□□
3. □□ □□□□□□□ □ □□ □□

□□□□

□□□□□□ **RAN Monitor** □

□□□ □□□□□□ eNodeB □□□□□

□□□□□

1. □□□□□□□

```
ping <base-station-ip>
```

2. RAN Monitor

- config/runtime.exs
- IP
-

3. RAN Monitor

-
-
-

4.

- WebLM PMCADM-1
- RAN Monitor IP
- 9076

"Pending"

"Pending"

1.

- RAN Monitor ID
-

2.

- runtime.exs
-

3.

- 30-60
-

問題

問題 "enetunreach" 問題

問題

1. 問題

- RAN Monitor 問題
- 問題
- VLAN/subnet 問題

2. 問題

- 9076 問題 (問題)
- 8080 問題 (SOAP API 問題)
- ACL 問題
- RAN Monitor 問題 iptables 問題

3. RAN Monitor 問題

```
# SOAP API 問題 webhook 問題  
netstat -tlnp | grep -E '8080|9076'
```

InfluxDB 問題

問題 問題

問題

1. 問題

- PMCADM-1 rTpmCollInterval 問題
- 問題
- InfluxDB 問題

2. InfluxDB 問題

- InfluxDB 問題

- 检查 "Connected" 状态
- 检查日志

3. 检查 RAN Monitor 配置

- 检查 InfluxDB 配置
- 检查数据库连接
- 检查 API 配置

TLS/证书

检查 TLS 配置

配置

1. 检查证书

- 检查证书文件
- 检查 RAN Monitor 是否信任 CA 证书

2. 检查 TLS 配置

- 检查 `tlsEnabled = false`
- 检查配置项
- 检查 TLS 配置

3. 检查日志

- 检查日志文件
 - 检查日志内容
 - 检查日志格式
-

□□□□

□□□□

- □□□□ - □□□ RAN Monitor □□□□
- □□□□□□□□ - RAN Monitor □□□□
- **Nokia Counter Reference** - □□□□□□□□
- **Grafana** □□ - □□□□□□□□□□□□□□
- **API** □□ - □□□□□ REST API □□
- □□□□□□□□ - □□□□□□□□□□

□□□□

- **config/runtime.exs** - RAN Monitor □□□□

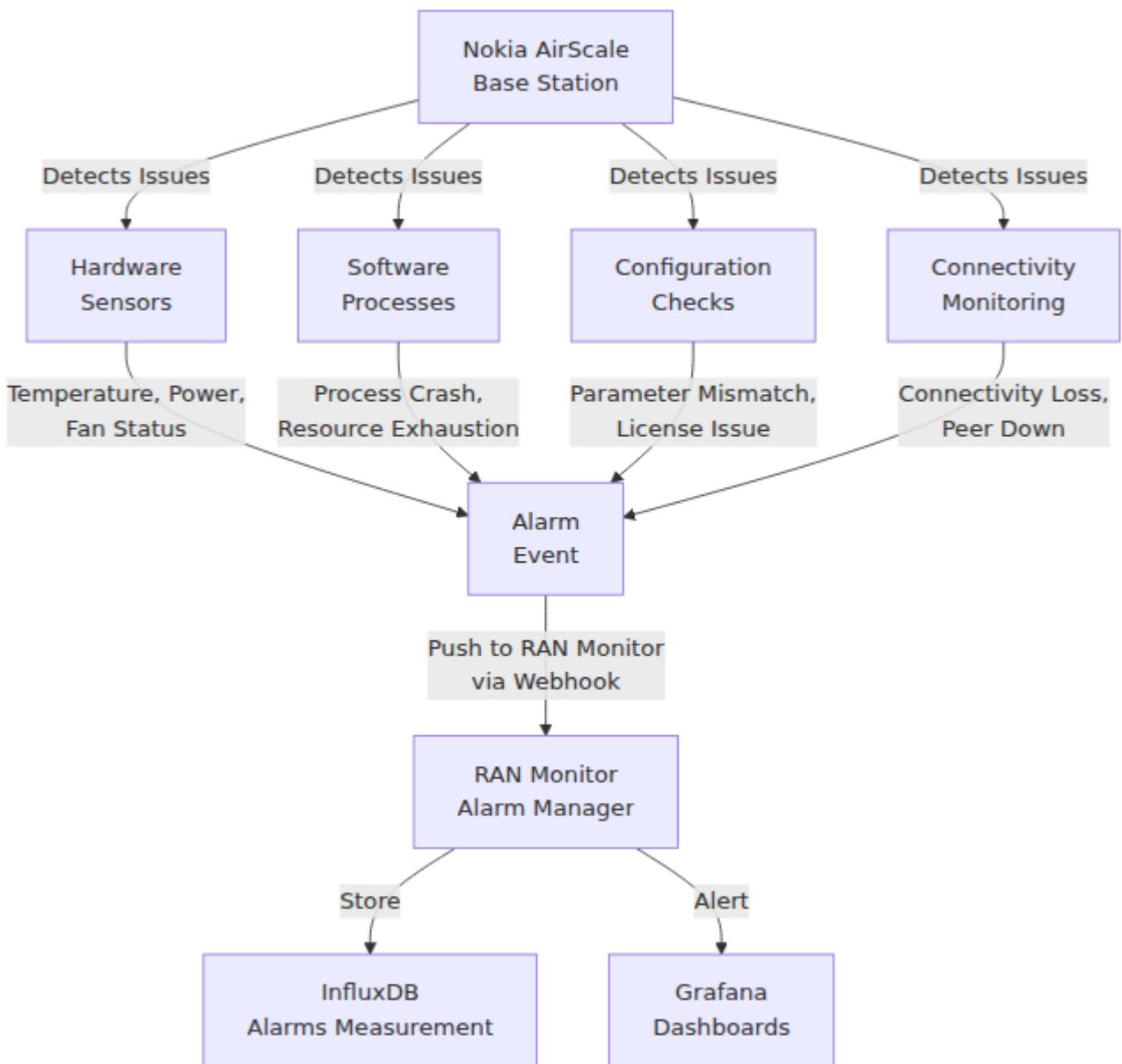
□□

□□□□□□□□□□□□

1. □□ RAN Monitor □□□□
2. □□□□□□□□□□ Nokia □□□□
3. □□□□□□□□□□□□
4. □□□□□□□□□□

4G

□□□□



□□□□□□

□□□□□□

項目	値	説明
ID	a1b2c3d4-e5f6-...	識別子
名前	XXXXXXXXXX	名前
住所	"XXXXXXXX"	住所
年齢	"S1 XXXXX"	年齢
住所	/BSC-1/BTS-23/Cell-A	住所DN
日時	2025-12-10 14:23:45	日時
性別	男 / 女	性別

□□□□□□

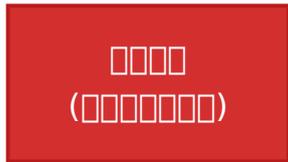
□□□□



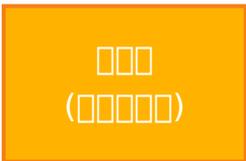
□□□□



□□□□



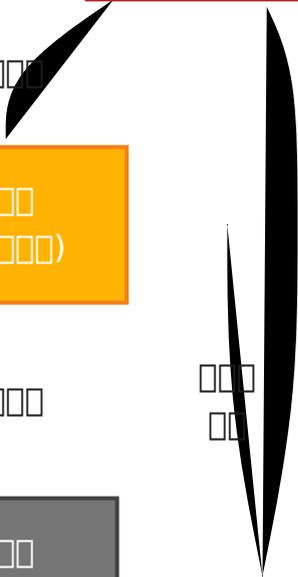
□□□□

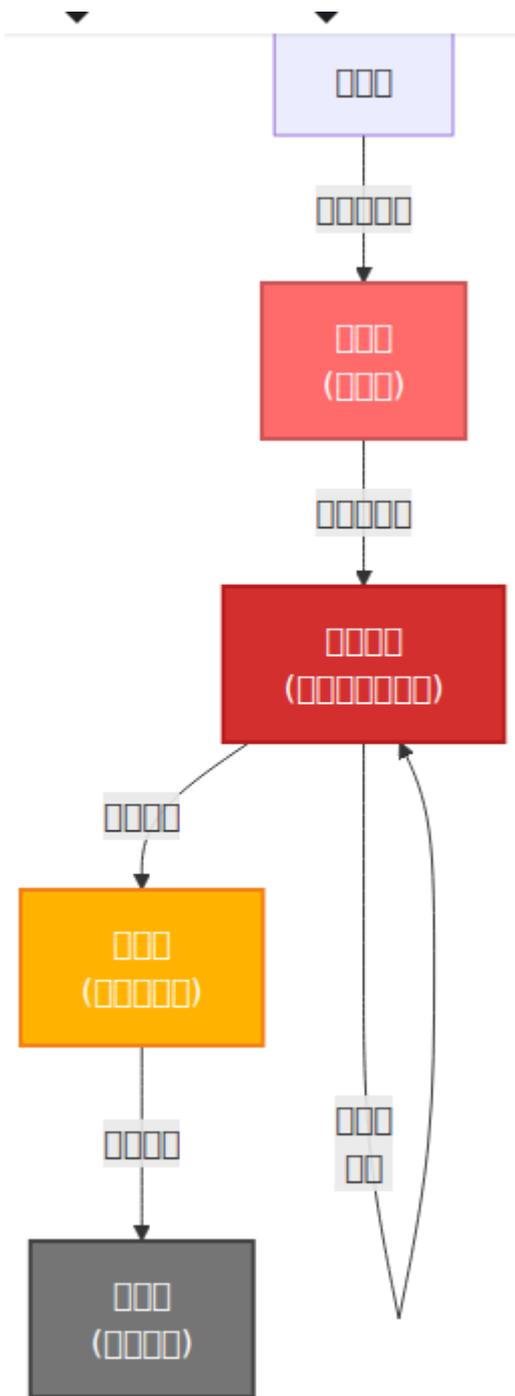


□□□□



□□□
□□





000

- 0000000000000000
- 00000000000000
- 00000000S1 000
- 00000000
- 00000000

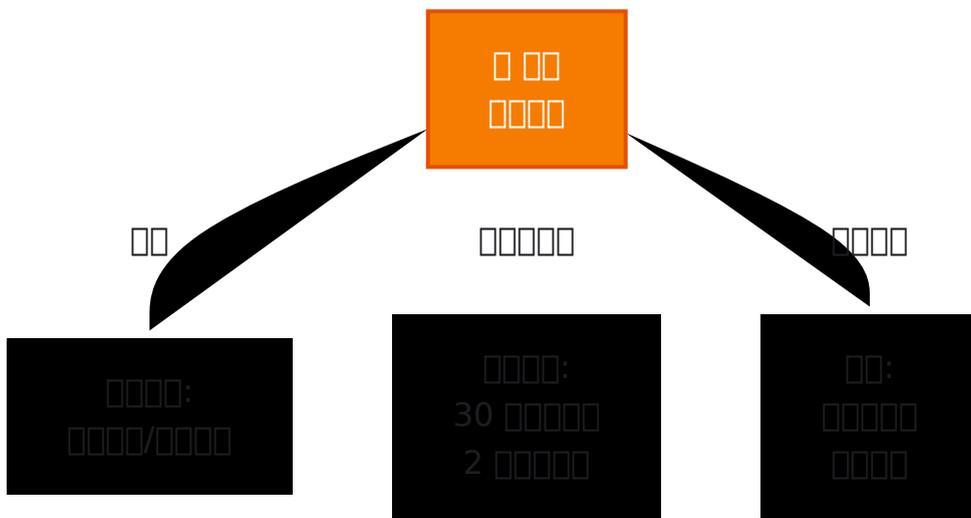
000

- 网络性能指标
- 网络可用性
- 网络延迟
- 网络 SLA 保障

网络 SLA < 15 分钟

网络性能

网络可用性



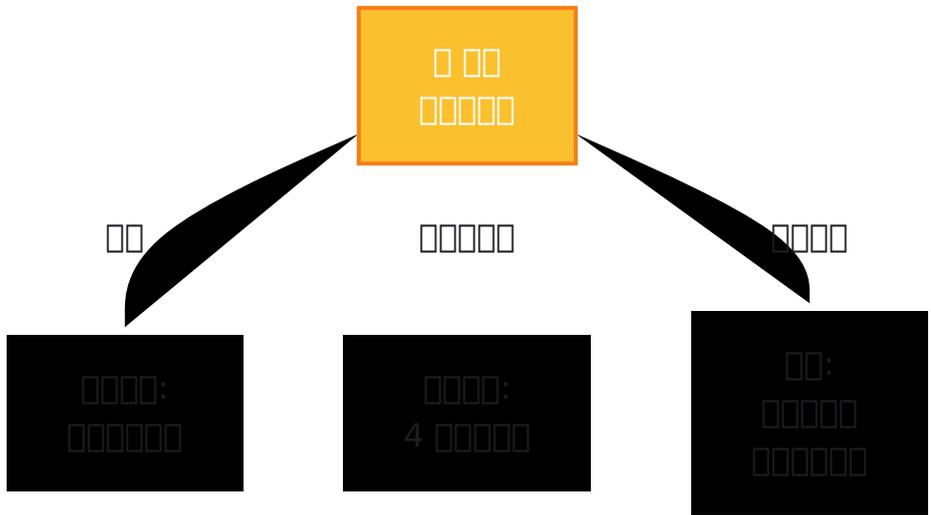
网络性能

- 网络可用性 < 95% 网络 15 分钟
- 网络延迟 < 95%
- DL/UL 网络吞吐量 > 95% 网络性能
- RLC 丢包 > 5%
- 网络性能指标
- 网络性能 E1/T1 网络性能

网络性能

- 网络 NOC 网络 + 网络性能
- 网络性能指标
- 网络 30 网络性能指标
- 网络性能指标/网络性能指标

SLA < 30



- 95-98%
-
-
-
-
- < 5

-
-
-
-

SLA

□□□□□

□□□ □□□□□□□□

□□□

- □□□□□ > 98% □□□□□
- □□/□□□□□□□□□□□□
- □□□□□□ 60-70%
- □□□□□□□□□□□□
- □□□□□□□□□□□

□□□

- □□□□□□□□
- □□□□□
- □□□□□□□

□□□

□□□ □□□□□□□□□□□□

□□□

- □□□□□□□
 - □□□□□□□□□□MTTR□
 - □□ SLA □□□□
 - □□□□□□
-

□□□□

□□□□

□□□□□□

□□□□

- S1 □□□□ → MME/SGW □□□□
- □□□□□□ → IP □□□□
- USIM □□□□□□ → HSS □□□□
- NTP □□□□ → □□□□□□□□

□□□□□□□□□□□□□□

□□□

1. □□□□□□□□□□□□
2. □□□□□□□□□□□□
3. □□□□□□□□□□
4. □□□□□□□□□□

□□□□□□□

□□□□□□□□

□□□□

- □□□□ → □□□□□□
- □□□□ → UPS/PSU □□
- □□□□ → □□□□□□
- □□□□□□ → □□□□□□
- □□□□ → □□□□□□

□□□□□□□□□□□□□□

□□□

1. □□□□□□□□□□□□
2. □□□□□□
3. □□□□□□□□
4. □□□□□ CPU □□□

□□□□□□

□□□□□□

□□□□□

- □□□□ → □□□□ OOM
- □ CPU □□□ → □□□□
- □□□□ → □□□□□□
- □□□□□ → □□□□

□□□□□□□□□□

□□□

1. □□□□□□
2. □□□□□□□□□□
3. □□ CPU/□□/□□□□
4. □□□□□□□

□□□□□□

□□□□□□□

□□□□□

- □□□□□ → □□□□□/□□
- DL □□□□□ → □□□□
- □□□□□□ → □□□□□□
- □□□□□ → RF □□□□□□□

□□□□□□□□□□

□□□

1. □□□□□□□□
2. □□□□□□□□
3. □□ RF □□□□
4. □□□□□□

□□□□

□□□□□□

□□□□□

- □□□□□ → □□□□□
- □□□□□ → □□□□□
- □□□□□□□ → □□□□□
- □□□□□□□ → □□□□□□□□□

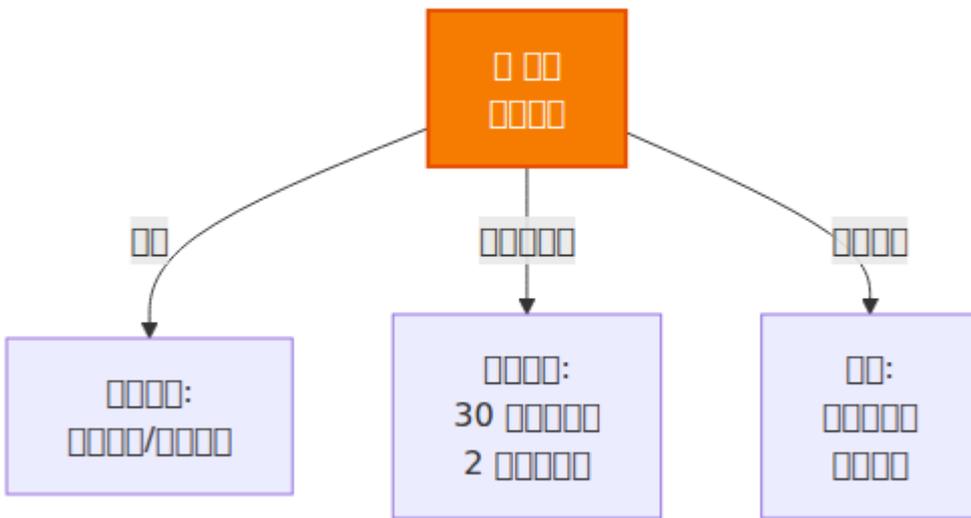
□□□□□□□□□□

□□□

1. □□□□□□
2. □□□□□□□□□
3. □□□□□□□□□□□□
4. □□□□□□□□□

□□□□□□□□

□□□□□□□



□□□□□□□□□□□□

□□□□□□□□□□□□□□

○○○○○○○

- ○○ ID ○○○○○○
- ○○○○○○○○
- ○○○○○○ DN○○○/○○/○○○
- ○○○○○○○○○○
- ○○○○○○○○○○
- ○○○○○○○○○○

○○○

- RAN Monitor Web UI → ○○○○
- Grafana → ○○○○○
- InfluxDB → ○○○○○○○○

○○○○○○○○○○○○

○○○○○○○○○○○○○○○○○○○○

○○○○○

- ○○○ AirScale ○○○○○○
- ○○○○○○○○○○○○○○
- RAN Monitor ○○○○○○○○
- ○○○○○○○○○○○○○○

○○○○○○○○○○○○

○○○○○○○○○○○○○○○○○○○○

```
○○○“DL ○○○○○”○○  
├ ○○ DL ○○○○○○ > 95%  
├ ○○○○○○○○○○○○○○  
├ ○○○○○○○○○○○○○○  
├ ○○○○○○○○○○○○○○  
└ ○○○○○○○○○○○○○○
```

□□□

- Grafana → □□□□□□□□
- Web UI → □□□□□□□□ → □□❓❓□
- InfluxDB □□□□□□□□□□

□□□□□□□□□□□□□□

□□□□□□□□□□□□□□

□□□□

- ├ □□□□□□□□ 4 □□□
- ├ □□□□□□□□ 24 □□□
- ├ □□□□□□□□□□ 7 □□
- ├ □□□□□□□□ 7 □□
- ├ □□□□□□□□ 7 □□
- └ □□□□□□□□□□□□

□□□

- RAN Monitor → □□□□□
- □□□□□□□
- □□□□□□□□□□□□□□
- □□□□□□□□

□□□□□□□□□□□□□□

□□□□□□□□□□□□□□

□□□□□□□□“□□□□□□”□□

□□□□□□



└ □□□□□□□□

└ □ → □□□□□□□□□□□□

└ □ → □□

└ □□□□□□□□□□□□

└ □□□□ → □□□□/□□□□

└ □□□□ → □□

└ □□□□□□□□□□

└ □ → □□□□□□□□□□□□

└ □ → □□

└ □□□□□□□□□□□□

└ □ → □□□□□□□□□□□□

└ → □□□□□□□□□□□□

└ □ → □□□□□□□□□□

└ → □□□□□□□□ DSP □□

└ □□□□□□□□□□

→ □□□□

→ □□□□□□

→ □□□□

□□□□□□□□□□

□□□□□□□□□□□□

□□□□□□

項目	単位	説明
項目	項目/項目	項目項目項目項目項目
項目	項目項目	項目項目
項目	項目/項目	項目項目項目項目
項目	項目/項目	項目項目項目
項目	項目項目	項目 BGP 項目項目項目

項目項目項目項目項目

項目項目項目項目項目

- 項目項目
- RAN Monitor 項目項目項目
- 項目項目項目項目
- 項目/項目項目
- 項目項目項目項目
- 項目項目項目項目項目
- 項目項目 > 30 項目項目

項目項目項目項目項目

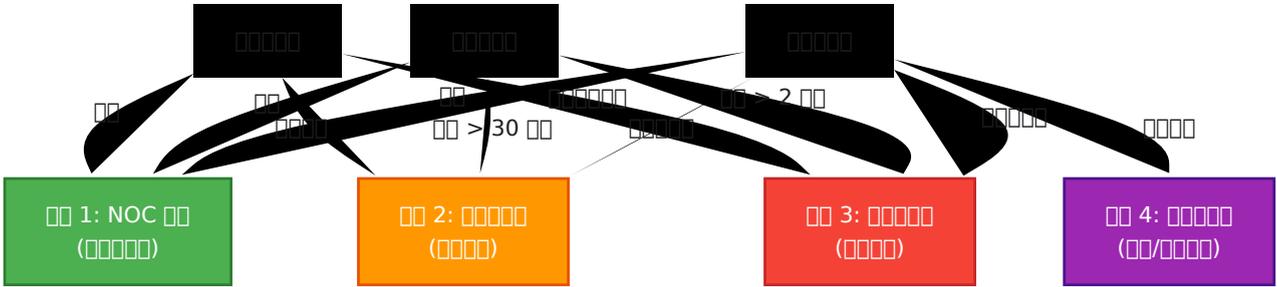
項目項目項目項目項目

項目

- 項目項目項目項目
- 項目項目
- 項目項目項目 SLA 項目
- 項目項目項目
- 項目項目項目

□□□□

□□□□



□□□□□□

□□□□□□□□□□□□□□□□ 2□

- □□□□□ 15 □□□□□□
- □□□□□ 30 □□□□□□
- □□□□□ NOC □□□□□□□□
- □□□□□□□/□□□□□
- □□□□□ > 1 □□□□

□□□□□□□□□□□□□□□□ 3□

- □□ 2 □ 1 □□□□□□□□
- □□□□□□□ > 30 □□
- □□□□□□□
- □□□□□□□□
- □□□□□□□□

□□□□□□[?][?][?]□□□□□ 4□□□□□□□□□□□□

- □□□□□□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□□□
- □□□□□/□□□□□□
- □□□□□□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□□□

0000

0000 2 0000

00000 - [000] - [00] - [00]

00000[2025-12-10 14:30 UTC]

00000[15 00]

000[SITE_A_BS1]

000[000000]

000

- 00 A1 00000
- 000000000000
- 00 ping 00

00000000

- 000000
- 000000000
- 00000000
- 000000

000

- 00000000%
- 000000
- 00000000

000000

- 0000000000
- 000000000000

00000000

- 0000
- 0000000000
- 000000000000

000000 [000000]

SLA

SLA

SLA

SLA

```
└─ 14:23:45 ← SLA
└─ 14:23:47 2 ← SLA
└─ 14:23:50 5 ← SLA
└─ 14:24:15 30 ← SLA
└─ 14:24:15 → 14:28:00 3.75 SLA
└─ 14:28:00 → 14:28:35 35 SLA + SLA
└─ 14:28:36 ← SLA
  └─ SLA 5 51
```

SLA

```
└─ SLA 2
└─ SLA 30
└─ SLA 5 51
└─ SLA SLA < 15 SLA
```

SLA

SLA

SLA

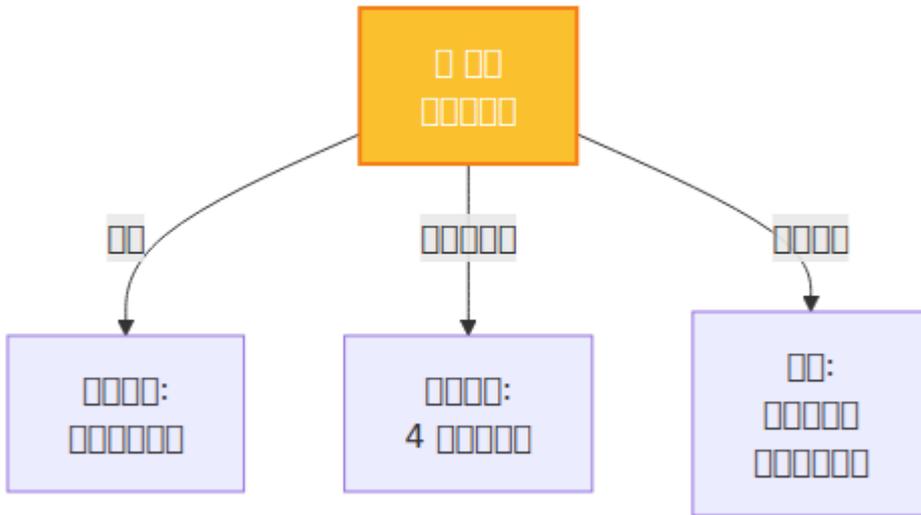
- SLA
- SLA
- SLA
- SLA
- MTTR SLA

SLA

- Grafana → SLA
- SLA

- 0000/0000 MTTR

0000000



00000

00000

1. 000000

- 00000000000000000000
- 00000000000000000000
- 000000
- 00000000

2. 0000

- 0000000000
- 000000000000
- 00000000000000
- 000000000000

3. 00000

- □□□□□□
- □□□□□□
- □□□□□□□□
- □□□□□□

4. □□□□

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

□□□□□□

□□□□□□□□□□□□□□□□

□□□

□□□[□□□□□]
□□□[□]
MTTR□[5-15 □□]
SLA □□□[30 □□□□□]

-
- □□□“□□□□□□”
 - □□□□□□□□
 - □□□□□□□□ 0%

- < 5 □□□□
1. □□□□□□ ping□
 2. □□□□□□□□□□
 3. □□□□□□□□□□□□□□□□

-
- 1□□□□□□□□□□
- ping □□□ping 192.168.1.100
 - □□□□□□□ → □□□□□□□

- 2□□□□□□
- □□□□□□□□□□
 - □□□□□□□ LED
 - □□□□□□□

- 3□□□□□□
- □□□□□□□□□□
 - □□ 60 □□□□
 - □□□□□□□□

- 4□□□□□□□□
- □□□□□□ 2
 - □□□□□□□
 - □□□□□

-
- □□ > 15 □□ → □□□ [□□□□□□]
 - □□ > 30 □□ → □□□ [□□□□□□]
 - □□□□□□□ → □□□□□□□□

-
- □□□□□□□□□□

- □□□□□□□□
- □□□□□□□□

API 監査ツール

このツールは RAN 監視用の REST API

を簡単に監視するための API を RAN 監視

目次

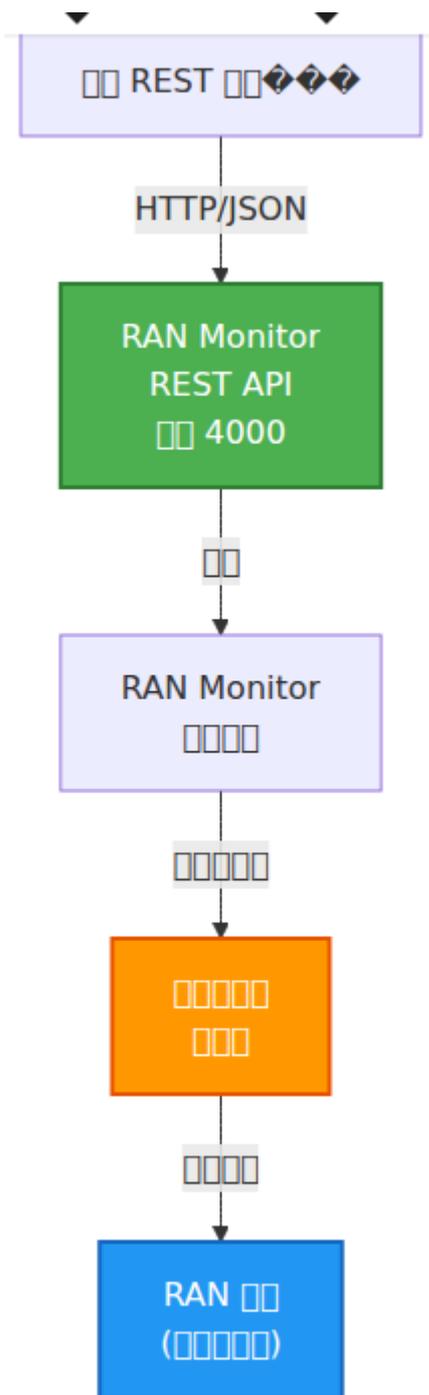
1. 概要
 2. API 一覧
 3. 監視対象
 4. 監視方法
 5. 監視結果
 6. 監視ログ
 7. 監視レポート
 8. 監視アラート
 9. API 仕様
-

概要

RAN Monitor 監視用の REST API を RAN 監視用の API を監視するための API を

- 監視対象 - 監視対象
- 監視方法 - 監視方法
- 監視結果 - 監視結果
- 監視ログ - 監視ログ
- 監視レポート - 監視レポート

API



API

- **RESTful** API - uses HTTP methods (GET, POST, PUT, DELETE)
- **JSON** API - uses JSON for data exchange
- **RESTful** API - uses RESTful API
- **RESTful** API - uses RESTful API

- `API - RAN Monitor`
- `API - RAN Monitor`

API

API

RAN Monitor RAN Monitor

API

1. API
- 2.
3. `???`
- 4.

API

RAN Monitor API

- `API`
 - `Authorization: Bearer <api-key>`
 - OAuth2
 - /VPN
- `API`
 -
 -
- `API`
 - API
 -

□□□□

□□□□

□□□□□□

```
GET /api/v1/devices
```

□□□

```
{
  "devices": [
    {
      "id": "nokia_bs1",
      "name": "SITE_A_BS1",
      "vendor": "Nokia",
      "address": "192.168.1.100",
      "port": 8080,
      "status": "registered",
      "registered_at": "2025-12-10T14:30:00Z",
      "session_active": true,
      "software_version": "BSC-2250.5.0",
      "license_required": false
    }
  ]
}
```

□□□□□□

```
GET /api/v1/devices/:id
```

□□□

```
{
  "device": {
    "id": "nokia_bs1",
    "name": "SITE_A_BS1",
    "vendor": "Nokia",
    "address": "192.168.1.100",
    "registration_status": "registered",
    "registration_key": "base64_encoded_key",
    "session_id": "nonuniquesession",
    "session_expiry": "2025-12-11T14:30:00Z",
    "device_info": {
      "type": "AirScale",
      "software_release": "5.0.0",
      "hardware_version": "2.0",
      "agent_unique_id": "airscale-001"
    }
  }
}
```

□□□□

```
PUT /api/v1/devices/:id/register
Content-Type: application/json
```

```
{
  "address": "192.168.1.100:8080",
  "web_username": "admin",
  "web_password": "password",
  "webhook_url": "http://manager.example.com:9076/webhook",
  "private_key_path": "/etc/certs/private.key",
  "public_key_path": "/etc/certs/public.key"
}
```

□□□

```
{
  "result": "Success",
  "registration_key": "base64_encoded_nonce",
  "device_id": "nokia_bs1",
  "message": "[]"
}
```

[]

- 200 - []
- 400 - []
- 409 - []
- 500 - []

[]

```
DELETE /api/v1/devices/:id
```

[]

```
{
  "result": "Success",
  "message": "[]",
  "device_id": "nokia_bs1"
}
```

[]

[]

```
PUT /api/v1/devices/:id/sessions
Content-Type: application/json
```

```
{
  "session_id": "session_unique_identifrier"
}
```

□□□

```
{
  "result": "Success",
  "session_id": "session_unique_identifrier",
  "session_timeout": 86400,
  "expires_at": "2025-12-11T14:30:00Z"
}
```

□□□□□□□□

- □□□□□□24 □□
- □□□□□□□□□□
- □ 20 □□□□□□□□

□□□□□□□□

```
GET /api/v1/devices/:id/sessions
```

□□□

```
{
  "session": {
    "active": true,
    "session_id": "session_unique_identifrier",
    "expires_at": "2025-12-11T14:30:00Z",
    "time_remaining_seconds": 82400,
    "last_activity": "2025-12-10T14:30:00Z"
  }
}
```

□□□□□

```
POST /api/v1/devices/:id/sessions/keep-alive
```

□□□

```
{  
  "result": "Success",  
  "new_expiry": "2025-12-11T14:30:00Z"  
}
```

□□□□

□□□□

□□□□□□□□

```
PUT /api/v1/devices/:id/config/upload  
Content-Type: application/json
```

```
{  
  "filter": {  
    "uploadType": "configuration",  
    "objects": [  
      {  
        "sdn": "/BSC-1/BTS-23/*",  
        "depth": 100  
      }  
    ],  
    "objectClass": ""  
  }  
}
```

□□□

```
{
  "result": "Success",
  "configuration": {
    "timestamp": "2025-12-10T14:30:00Z",
    "device_id": "nokia_bs1",
    "parameters": {
      "/BSC-1/BTS-23": {
        "BtsBasics": {
          "BtsName": "CELL_A",
          "BtsType": "MACRO",
          "EnvironmentalSpecifications": {
            "TemperatureRange": "Industrial"
          }
        },
        "CarrierAggregation": {
          "CarrierAggregationCapability": true,
          "MaxUECarriers": 5
        }
      }
    }
  }
}
```

□□□□□□

```
PUT /api/v1/devices/:id/config/set
Content-Type: application/json

{
  "parameter_path": "/BSC-1/BTS-23/BtsBasics/BtsName",
  "value": "NEW_CELL_NAME",
  "value_type": "string"
}
```

□□□

```

{
  "result": "Success",
  "parameter": "/BSC-1/BTS-23/BtsBasics/BtsName",
  "old_value": "CELL_A",
  "new_value": "NEW_CELL_NAME",
  "applied_at": "2025-12-10T14:30:45Z"
}

```

□□□□□□

□□	□□	□□	□□
BtsName	□□□	"SITE_A_Cell_1"	□□/□□□□□□
MaxUEsServed	□□	256	□□□□□□□□
CellTXPower	□□	40 (dBm)	□□□□□□□□
EnableCarrierAgg	□□	true	□□□□□□□□
HandoverHysteresis	□□	3 (dB)	□□  □□

□□□□□□

```
GET /api/v1/devices/:id/config/history?limit=10&days=7
```

□□□

```
{
  "history": [
    {
      "timestamp": "2025-12-10T14:30:45Z",
      "change_type": "parameter_modified",
      "parameter": "/BSC-1/BTS-23/BtsBasics/BtsName",
      "old_value": "CELL_A",
      "new_value": "NEW_CELL_NAME",
      "reason": "□□□□□□"
    }
  ]
}
```

□□□□

□□□□□□

□□□□□□□□□□

PUT /api/v1/devices/:id/metrics/upload
Content-Type: application/json

```
{
  "filter": {
    "uploadType": "measurement",
    "objects": [
      {
        "sdn": "*",
        "depth": 100
      }
    ]
  }
}
```

□□□

```
{
  "result": "Success",
  "metrics": {
    "timestamp": "2025-12-10T14:30:00Z",
    "measurement_interval": 300,
    "counters": [
      {
        "id": "M1C1",
        "name": "□□□□□□□□",
        "value": 125.4,
        "unit": "Mbps",
        "cell_dn": "/BSC-1/BTS-23/Cell-1"
      },
      {
        "id": "M1C2",
        "name": "□□□□□□□□",
        "value": 89.2,
        "unit": "Mbps",
        "cell_dn": "/BSC-1/BTS-23/Cell-1"
      }
    ]
  }
}
```

□□□□□□

```
PUT /api/v1/devices/:id/alarms/upload
Content-Type: application/json
```

```
{
  "filter": {
    "uploadType": "active_faults"
  }
}
```

□□□

```
{
  "result": "Success",
  "alarms": [
    {
      "alarm_id": "a1b2c3d4",
      "severity": "Critical",
      "probable_cause": "□□□□□",
      "specific_problem": "□□□□□",
      "affected_dn": "/BSC-1/BTS-23/Cell-1",
      "event_time": "2025-12-10T14:15:30Z",
      "description": "□□ 1 □□□□□□□□□□"
    }
  ]
}
```

□□□□□□

```
PUT /api/v1/devices/:id/topology/upload
Content-Type: application/json
```

```
{
  "filter": {
    "uploadType": "topology",
    "objects": [
      {
        "sdn": "*",
        "depth": 100
      }
    ]
  }
}
```

□□□

```
{
  "result": "Success",
  "topology": {
    "device_dn": "/BSC-1",
    "managed_elements": [
      {
        "name": "BTS-23",
        "type": "BTS",
        "dn": "/BSC-1/BTS-23",
        "cells": [
          {
            "name": "Cell-1",
            "type": "EUTRANCell",
            "physical_cell_id": 100,
            "frequency": 2110
          }
        ]
      }
    ]
  }
}
```

□□□□

Ping □□

```
PUT /api/v1/devices/:id/ping
```

□□□

```
{
  "result": "Success",
  "device_id": "nokia_bs1",
  "latency_ms": 45,
  "status": "OK"
}
```

□□□□□□

GET /api/v1/health/status

□□□

```
{
  "status": "healthy",
  "devices": {
    "total": 50,
    "registered": 48,
    "active_sessions": 45,
    "unreachable": 2
  },
  "database": {
    "mysql": "connected",
    "influxdb": "connected"
  },
  "timestamp": "2025-12-10T14:30:00Z"
}
```

□□□□

□□□□□□

Nokia eNodeB □□□□□□□□

```

/SystemFunctions
├─ /BSC-1 (□□□□□)
│  └─ /BTS-23 (□□□□□□)
│     └─ BtsBasics (□□□□□□□□)
│        └─ /Cell-1
│           └─ CellCommonData
│              └─ CellAdvanced
│                 └─ CarrierAggregation
│                    └─ /Cell-2
│                       └─ ...
└─ /Connectivity
   └─ S1Interface
      └─ X2Interface
         └─ NetworkConfiguration

```

□□□□□□

□□/□□□□

```

{
  "parameter_path": "/BSC-1/BTS-23/Cell-1/CellCommonData/AdminState",
  "value": "UNLOCKED",
  "value_type": "enum"
}

```

□□□□□ LOCKED, UNLOCKED, SHUTTING_DOWN

□□□□□□

```

{
  "parameter_path": "/BSC-1/BTS-23/Cell-1/CellAdvanced/CellTXPower",
  "value": "35",
  "value_type": "integer"
}

```

□□□□0-46 dBm□□□□□□□□

□□□□□□

```
{  
  "parameter_path": "/BSC-1/BTS-23/Cell-  
1/CellAdvanced/HandoverHysteresis",  
  "value": "3",  
  "value_type": "integer"  
}
```

□□□dB□□□□□□0-8 dB

□□□□□□□□

```
{  
  "parameter_path": "/BSC-1/BTS-23/MaxUEsServed",  
  "value": "256",  
  "value_type": "integer"  
}
```

□□□□□□□□

□□□□

□□□ **1**□□□□

□□□□□

1. □□□□
POST /api/devices

2. □□□□
PUT
/api/devices/:id/register

3. □□□□
PUT
/api/devices/:id/sessions

4. □□□□
PUT
/api/devices/:id/config/upload

5. □□□□
PUT
/api/devices/:id/metrics/upload

□□□□□
□□□

□□□

```
# 1. 创建设备
curl -X POST http://localhost:4000/api/v1/devices \
  -H "Content-Type: application/json" \
  -d '{
    "id": "site_a_bs1",
    "name": "SITE_A_BS1",
    "vendor": "Nokia",
    "address": "192.168.1.100:8080",
    "credentials": {
      "username": "admin",
      "password": "password"
    }
  }'
```

```
# 2. 注册设备
curl -X PUT
http://localhost:4000/api/v1/devices/site_a_bs1/register \
  -H "Content-Type: application/json" \
  -d '{
    "webhook_url": "http://manager.example.com:9076/webhook"
  }'
```

```
# 3. 创建会话
curl -X PUT
http://localhost:4000/api/v1/devices/site_a_bs1/sessions \
  -H "Content-Type: application/json" \
  -d '{"session_id": "session_001"}'
```

```
# 4. 上传配置
curl -X PUT
http://localhost:4000/api/v1/devices/site_a_bs1/config/upload \
  -H "Content-Type: application/json" \
  -d '{
    "filter": {
      "uploadType": "configuration",
      "objects": [{"sdn": "*", "depth": 100}]
    }
  }'
```

000 2000000

```
# 1. 00000
curl -X PUT
http://localhost:4000/api/v1/devices/site_a_bs1/config/upload \
-H "Content-Type: application/json" \
-d '{
  "filter": {
    "uploadType": "configuration",
    "objects": [{"sdn": "/BSC-1/BTS-23/Cell-1", "depth": 10}]
  }
}' | jq '.configuration.parameters["/BSC-1/BTS-23/Cell-1"]'

# 2. 00000
curl -X PUT
http://localhost:4000/api/v1/devices/site_a_bs1/config/set \
-H "Content-Type: application/json" \
-d '{
  "parameter_path": "/BSC-1/BTS-23/Cell-
1/CellAdvanced/CellTXPower",
  "value": "38",
  "value_type": "integer"
}'

# 3. 00000
curl -X PUT
http://localhost:4000/api/v1/devices/site_a_bs1/config/upload \
-H "Content-Type: application/json" \
-d '{
  "filter": {
    "uploadType": "configuration",
    "objects": [{"sdn": "/BSC-1/BTS-23/Cell-1/CellAdvanced",
"depth": 5}]
  }
}' | jq '.configuration.parameters["/BSC-1/BTS-23/Cell-
1/CellAdvanced/CellTXPower"]'
```

000 3000000

```
# 0000000000000000
#!/bin/bash

DEVICE="site_a_bs1"
INTERVAL=300 # 5 00

while true; do
    # 0000
    METRICS=$(curl -s -X PUT
http://localhost:4000/api/v1/devices/$DEVICE/metrics/upload \
-H "Content-Type: application/json" \
-d '{
    "filter": {
        "uploadType": "measurement",
        "objects": [{"sdn": "*", "depth": 100}]
    }
}')

    # 000000
    DL=$(echo $METRICS | jq '.metrics.counters[] |
select(.id=="M1C1") | .value')
    CELLS=$(echo $METRICS | jq '.metrics.counters | length')

    echo "$(date): DL=$DL Mbps, Cells=$CELLS"

    # 0000
    ALARMS=$(curl -s -X PUT
http://localhost:4000/api/v1/devices/$DEVICE/alarms/upload \
-H "Content-Type: application/json" \
-d '{
    "filter": {
        "uploadType": "active_faults"
    }
}' | jq '.alarms | length')

    if [ "$ALARMS" -gt 0 ]; then
        echo "000$ALARMS 00000"
    fi
fi
```

```
sleep $INTERVAL
done
```

□□□□

HTTP □□□

□□	□□	□□
200	□□	□□□□□
201	□□	□□□□□
400	□□□□	□□□ JSON □□□
401	□□□	□□/□□□ API □□
404	□□□	□□□□□
409	□□	□□□□□
500	□□□□□	□□□□□□□□
503	□□□	□□□□

□□□□□□

```
{
  "error": {
    "code": "DEVICE_NOT_FOUND",
    "message": "□□ 'site_a_bs1' □□□",
    "details": {
      "device_id": "site_a_bs1",
      "timestamp": "2025-12-10T14:30:00Z"
    }
  }
}
```

□□□□

□□□□□□

```
{
  "error": {
    "code": "NOT_REGISTERED",
    "message": "□□□□□□□□□□",
    "solution": "□□□□ PUT /api/devices/:id/register"
  }
}
```

□□□□□

```
{
  "error": {
    "code": "SESSION_EXPIRED",
    "message": "□□□□□□",
    "solution": "□□ PUT /api/devices/:id/sessions □□□□□□"
  }
}
```

□□□□□□□□

```
{
  "error": {
    "code": "INVALID_PARAMETER",
    "message": "□□□□□□□□",
    "details": {
      "parameter": "/BSC-1/BTS-23/Cell-
1/CellAdvanced/CellTXPower",
      "value": "99",
      "valid_range": "0-46 dBm"
    }
  }
}
```

API □□

Python □□□□□

```
import requests
import json

class RanMonitorClient:
    def __init__(self, base_url="http://localhost:4000/api/v1"):
        self.base_url = base_url
        self.session = requests.Session()

    def register_device(self, device_id, address, username,
password):
        """□□□□□□"""
        url = f"{self.base_url}/devices/{device_id}/register"
        payload = {
            "address": address,
            "web_username": username,
            "web_password": password,
            "webhook_url": "http://manager:9076/webhook"
        }
        response = self.session.put(url, json=payload)
        return response.json()

    def get_config(self, device_id, sdn="*", depth=100):
        """□□□□□□□□"""
        url = f"{self.base_url}/devices/{device_id}/config/upload"
        payload = {
            "filter": {
                "uploadType": "configuration",
                "objects": [{"sdn": sdn, "depth": depth}]
            }
        }
        response = self.session.put(url, json=payload)
        return response.json()

    def set_config(self, device_id, parameter_path, value,
value_type="string"):
        """□□□□□□□□"""
        url = f"{self.base_url}/devices/{device_id}/config/set"
        payload = {
```

```

        "parameter_path": parameter_path,
        "value": value,
        "value_type": value_type
    }
    response = self.session.put(url, json=payload)
    return response.json()

def get_metrics(self, device_id):
    """Get metrics"""
    url = f"
{self.base_url}/devices/{device_id}/metrics/upload"
    payload = {
        "filter": {
            "uploadType": "measurement",
            "objects": [{"sdn": "*", "depth": 100}]
        }
    }
    response = self.session.put(url, json=payload)
    return response.json()

# Create client
client = RanMonitorClient()

# Register device
result = client.register_device(
    device_id="site_a_bs1",
    address="192.168.1.100:8080",
    username="admin",
    password="password"
)
print(f"Registered device: {result}")

# Get configuration
config = client.get_config("site_a_bs1")
print(f"Configuration: {json.dumps(config, indent=2)}")

# Set configuration
update = client.set_config(
    "site_a_bs1",
    "/BSC-1/BTS-23/Cell-1/CellAdvanced/CellTXPower",
    "38",
    "integer"
)

```

```
)  
print(f"□□□{update}")
```

cURL □□

□□□□□

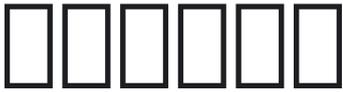
```
curl -X PUT  
http://localhost:4000/api/v1/devices/site_a_bs1/register \  
-H "Content-Type: application/json" \  
-d '{  
  "address": "192.168.1.100:8080",  
  "web_username": "admin",  
  "web_password": "password",  
  "webhook_url": "http://manager:9076/webhook"  
}'
```

□□□□□□□□

```
curl -X GET http://localhost:4000/api/v1/devices/site_a_bs1
```

□□□□□

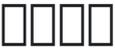
```
curl -X PUT  
http://localhost:4000/api/v1/devices/site_a_bs1/config/upload \  
-H "Content-Type: application/json" \  
-d '{  
  "filter": {  
    "uploadType": "configuration",  
    "objects": [{"sdn": "/BSC-1/*", "depth": 50}]  
  }  
}' | jq '.'
```



AirScale インストールと設定



このドキュメントは AirScale のインストールと設定、InfluxDB のインストールと XML のインストールと設定について説明しています。



- インストール - インストール手順
 - インストール - インストールオプション
 - インストール - インストール環境
 - インストール - インストール容量 100 MB ~ 690 MB
 - **Web** インターフェース - インストール
 - インストール - インストールオプション
 - **InfluxDB** インターフェース - インストール
 - インストール - インストールオプション
-



このドキュメントは AirScale のインストールと設定

- インストール 1 手順
- インストール オプション
- インストール インストール環境 
- インストール インストール容量

目次

目次 RAN Monitor 目次

```
priv/airscale_configs/
├── <airscale-name>/
│   ├── current.xml # 目次
│   ├── ONS-Lab-Airscale_config_20251230_143522.xml # 目次 12 月
30 目次 2:35
│   ├── ONS-Lab-Airscale_config_20251229_120000.xml # 目次
│   ├── ONS-Lab-Airscale_config_20251228_093045.xml # 目次
│   └── ... # 目次
100 MB 目次
```

目次 <AirScaleName>_config_YYYYMMDD_HHMMSS.xml

目次 AirScale 目次

目次

- 目次 目次 current.xml
- 目次 目次
- 目次 目次 100 MB 目次
- 目次 目次 Web UI 目次 current.xml 目次
- 目次 目次
- 目次 目次

目次

目次

目次 目次 目次

- 目次 目次 100 MB 目次
- 目次 目次

- 000000 000000000000
- 0000000 00000 `current.xml` 00000000
- 00000 000000 145KB 000 ~690 0000000000000000

00000

000000000000

1. 000000 - 00000 `<AirScale>_config_YYYYMMDD_HHMMSS.xml`
2. 00000 - `current.xml` 00000000
3. 00000 - 000000000000000000
4. 000000 - 000000 > 100 MB 0000000000000000
5. 00000 - 000000000000

□□□□

```
□□□□□□ 95 MB□655 □□□□□□ 145KB□  
├─ ONS-Lab-Airscale_config_20240101_100000.xml <- □□□145KB□  
├─ ONS-Lab-Airscale_config_20240102_100000.xml (145KB)  
├─ ... (□□ 653 □□□)  
└─ ONS-Lab-Airscale_config_20251230_100000.xml <- □□□145KB□
```

□ 2025-12-31 10:00:00 □□□□□□□145KB□

□□□

1. □□□ONS-Lab-Airscale_config_20251231_100000.xml (145KB)
2. □□□□□□95 MB + 145KB = 95.14 MB□□□ 100 MB □□□□
3. □□□□□
4. □□□656 □□□□□□ 95.14 MB

□□□□□□□□□□□□□□□□□□□□□□□□ 500KB□

□□□

1. □□□ONS-Lab-Airscale_config_20251231_150000.xml (500KB)
2. □□□□□□95.14 MB + 500KB = 95.64 MB□□□□□□□□
3. □□□□□
4. □□□657 □□□□□□ 95.64 MB

□□□□□□□□□□□□□□

□□□□□□99.8 MB□685 □□□□

□□□□200KB

1. □□□□□
2. □□□□□□100 MB□□□□□□□□
3. □□□□□□□□□□□□ < 100 MB
4. □□□□□□“□□ 3 □□□□□□ 435 KB”
5. □□□682 □□□□□□ 99.6 MB

URL `https://<ran-monitor-ip>:9443/nokia/config-archive`

////////////////////////////////////

□□□□

□□□□□□□□□□□□□□

1. □□□□

- □□□□ - □□□□□□□□ AirScale □□
- □□□□ - □□□□□□□□□□□□
- □□ - □□□□□□□□□□□□
- □□□□□□ - □□□□□□□□□□□□

2. □□□□

□□□□□□□□□□□□□□

項目	内容
時刻	UTC
設定ファイル	config_20251230_143522.xml
容量	KB / MB
バージョン	"2" / "3"
備考	

時刻は UTC を使用します。

設定ファイルは current.xml を使用します。

3. 備考

備考欄

- 項目 - 設定値
- 項目 - 設定値
- 項目 - 設定値

備考欄

備考欄

時刻は UTC を使用します。

備考

- 項目 - 設定値
- 項目 - 設定値
- 項目 - 設定値

4. 如何 如何 如何

5. 如何如何如何 `<AirScaleName>_config_YYYYMMDD_HHMMSS.xml` 如何如何如何如何

如何

- 如何如何
- 如何如何如何如何
- 如何如何如何如何如何
- 如何如何如何如何如何

如何如何

如何 如何如何如何如何如何如何如何

如何

1. 如何如何如何如何
2. 如何
3. 如何如何如何如何
4. 如何 如何 如何
5. 如何如何如何如何如何
6. 如何如何如何如何

如何如何

- 如何 `current.xml` 如何如何如何如何
- 如何如何如何如何 `???`
- 如何如何如何如何如何如何如何

如何如何

如何 如何如何如何如何如何如何如何

如何如何

1. 如何如何如何如何如何
2. 如何 XML 如何如何如何 `xmldiff` `Beyond Compare` `WinMerge`

3. 配置文件的差异

配置文件的差异

```
# 下载配置文件
wget https://<server>:9443/download/config/ONS-Lab-Airscale/ONS-Lab-Airscale_config_20251230_143522.xml
wget https://<server>:9443/download/config/ONS-Lab-Airscale/ONS-Lab-Airscale_config_20251229_120000.xml

# 使用 diff 命令比较
diff ONS-Lab-Airscale_config_20251229_120000.xml ONS-Lab-Airscale_config_20251230_143522.xml

# 使用 xmldiff 命令比较
xmldiff ONS-Lab-Airscale_config_20251229_120000.xml ONS-Lab-Airscale_config_20251230_143522.xml
```

配置文件的差异

配置文件的差异

配置文件的差异

配置文件的差异

1. 配置文件的差异
2. 配置文件的差异
3. 配置文件的差异
4. 配置文件的差异
5. 配置文件的差异
6. 配置文件的差异

配置文件的差异

ファイル

- ONS-Lab-Airscale_config_20251230_143522.xml (143KB) - ファイル
- ONS-Lab-Airscale_config_20251228_091045.xml (142KB) - 2 ファイル
- ONS-Lab-Airscale_config_20251225_180000.xml (142KB) - 5 ファイル

ディレクトリ

- ディレクトリ 142KB ディレクトリ 143KB 12 12 30
- ディレクトリ 12 12 28 ディレクトリ 12 12 30
- ディレクトリ

操作

操作 操作

操作

1. 操作
2. 操作
3. 操作 Web UI
4. 操作 → 操作 ID
5. 操作 → 操作
6. 操作
7. 操作
8. 操作

操作

- ✓ 操作
- ✓ 操作
- ✓ 操作
- ✓ 操作
- ✓ 操作

操作

操作 “操作”

□□□□

1. □□□□□□□□
 2. □□□□□□
 3. □□□□□□□□□□
 4. □□□□□□□
 5. □□□□□□
 6. □□□□□□□□
-

□□□□

□□□□□□

□□□□□□□□□□□□□□

□□□□□

1. □□□□/□□□□
2. □ XML □□□□□□□□
3. □□□□□□□
4. □□□□□□□□□□

□□□

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

□□□

```

<!-- 10000000 -->

<!-- 100000000 -->
<parameter>
  <name>cellId</name>
  <value>1</value>
</parameter>

<!-- 20000 -->
<parameter><name>cellId</name><value>1</value></parameter>

```

概要

10000000 ~145 KB 100000000 AirScale 1000

仕様

項目	仕様	100000000 145KB	項目
10	100 MB	~690	1 GB
50	100 MB	~690	5 GB
100	100 MB	~690	10 GB
500	100 MB	~690	50 GB
1000	100 MB	~690	100 GB

備考

- 100000000 100 MB
- 100000000 ~690 145KB
- 100000000 100000000 current.xml 145KB
- 100000000 100 MB
- 100000000 100000000

□□□□

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

□□□□

□□□□

- □□□□□□ **100 MB** □□
- □□□□□□□□□□□□
- □□□□ `current.xml` □□□□□□
- □□□□□□□□□□□□
- □□□□□□□□□□□□

□□□□□□

□□□□□ ConfigStorage □□□□□□

```
# □ lib/ran_monitor/nokia/airscale/config_storage.ex □
# □□□□□□□□□□
@max_storage_bytes 100 * 1024 * 1024 # □□ 100 MB

# □□□□□□□□□□
@max_storage_bytes 50 * 1024 * 1024 # 50 MB□□□ ~345 □□□□
@max_storage_bytes 200 * 1024 * 1024 # 200 MB□□□ ~1380 □□□□

# □□□□□□□□□□
def cleanup_old_versions(airscale_name, max_size_bytes \
@max_storage_bytes)
```

□□□□□□□□□□

```
mix compile
# □□ RAN Monitor □□□□□□
```

□□□□

□□□□ 1 □□□3,600,000 □□□

□□□□□□□□

□□ `lib/ran_monitor/nokia/airscale/manager.ex` □

```
defp schedule_get_airscale_config do
  # □ 1 □□□□□□□3,600,000 ms□
  Process.send_after(self(), :get_airscale_config, 3_600_000)
end
```

□□□□□

- 30 □□□ `1_800_000`
- 1 □□□ `3_600_000` □□□□
- 2 □□□ `7_200_000`
- 4 □□□ `14_400_000`
- 24 □□□ `86_400_000`

□□□□□□□□□□ RAN Monitor □

□□□□

□□□□□□□□

□□□

- □□□□□□□□“0 □□□”
- □□□□□□□□

□□□□□

1. □□□□□

- 000000
- 000000“000”00
- 0000000000000000

2. 000000

- 0000000000
- 00000000“00”
- 00000000

3. 0000000

- 0000000000000000
- 00000000

```
Kernel.send(Process.whereis(RanMonitor.Nokia.Airscale.Manager),
: get_ airscale_ config)
```

4. 0000000

- 00 `priv/airscale_configs/` 00000000
- 00 RAN Monitor 00000000
- 00000000000000000000

0000 404 00

000

- 00000000“00000000”
- 000000 404 00

000000

1. 000000000

- 000000000000000000000000
- 00 `priv/airscale_configs/` 0000000000

2. 000000

- 配置文件的目录结构
- 配置文件的命名规范

3. 配置

- 配置 Web 服务的配置
- 配置文件的权限

配置

```
# 配置文件的目录结构
ls -la priv/airscale_configs/

# 配置文件的命名规范
ls -la priv/airscale_configs/<device-name>/

# 配置文件的权限
chmod 755 priv/airscale_configs/
chmod 644 priv/airscale_configs/*/*.xml
```

配置

配置

- 配置 `current.xml` 文件
- 配置文件的权限为 1

配置

1. 配置

- 配置文件的目录结构
- 配置文件的命名规范“配置文件的命名规范”

2. 配置

- 配置文件的目录结构
- 配置文件的命名规范
- 配置文件的权限

3. 環境構築

- 環境構築
- 環境構築

環境

```
# 環境構築
grep "process_configuration" <log-file>

# 環境構築
# IEx 環境構築
Kernel.send(Process.whereis(RanMonitor.Nokia.Airscale.Manager),
: get_airscale_config)
```

環境構築

環境構築

環境構築

環境構築

```
#!/bin/bash
# backup-configs.sh - backup configs

BACKUP_DIR="/backup/ran-monitor/configs"
CONFIG_DIR="priv/airscale_configs"
DATE=$(date +%Y%m%d)

# create backup directory
mkdir -p "$BACKUP_DIR/$DATE"

# sync files
rsync -av "$CONFIG_DIR/" "$BACKUP_DIR/$DATE/"

# cleanup old backups
# keep 30 days
find "$BACKUP_DIR" -type d -mtime +30 -exec rm -rf {} +

echo "Backup completed: $BACKUP_DIR/$DATE"
```

cron

```
0 2 * * * /path/to/backup-configs.sh
```

FAQ

How to backup files

How to

1. Create backup directory
2. Sync files
3. Cleanup old backups

How to

```
# config_changes.log

2025-12-30 14:35:22 - ONS-Lab-Airscale
#####John Smith
##### 1 #####
#####txPower 40dBm 43dBm
#####
#####2025-12-30 14:40:00
#####

2025-12-28 09:10:45 - ONS-Lab-Airscale
#####Jane Doe
#####
##### 101112
#####
#####2025-12-28 09:15:00
#####
```

####

#####

#####

```
# #####
du -sh priv/airscale_configs/

# #####
du -sh priv/airscale_configs/*/

# #####
find priv/airscale_configs/ -name "*.xml" | \
  sed 's|/[^/]*\.xml||' | uniq -c
```

####

- #####500MB#####
- #####
- ##### 10% #####

□□□□□□□□

□□□□□□□□□□□□□□□□

□□□□□□

□□□□□□

- □□□□□□□□ `current.xml`
- □□□□□□□□“□□”□□□□□□□□□□

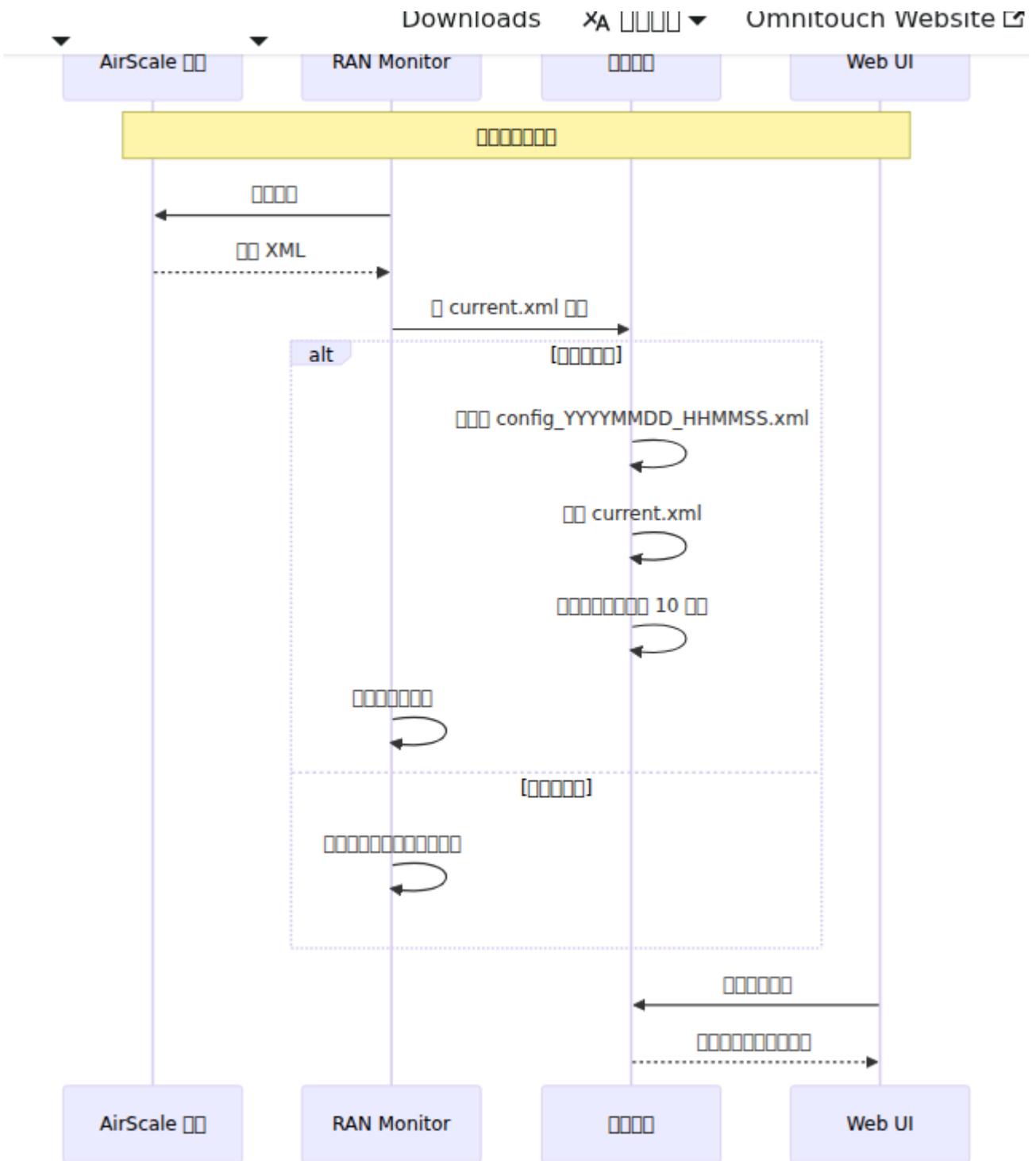
□□□□□□□□

- □□□□□□□□
- □□ → □□ → □□□□□□
- □□□□□□□□□□□□□□

□□□□□

- □□□□□□□□
- □□□□□□□□
- □□□□ → □□□□□□□□

□□□



API □□

□□□□□□□□□□ Web UI □□□□□□□□□□ HTTP □□□□□□□□


```
# 列出配置版本
RanMonitor.Nokia.Airscale.ConfigStorage.list_config_versions("ONS-
Lab-Airscale")

# 获取当前配置
{:ok, xml} =
RanMonitor.Nokia.Airscale.ConfigStorage.get_current_config("ONS-
Lab-Airscale")

# 统计版本数量
RanMonitor.Nokia.Airscale.ConfigStorage.count_versions("ONS-Lab-
Airscale")
# 返回655

# 获取最大存储字节数
RanMonitor.Nokia.Airscale.ConfigStorage.max_storage_bytes()
# 返回104857600即100 MB

# 获取存储使用情况
RanMonitor.Nokia.Airscale.ConfigStorage.get_storage_usage("ONS-
Lab-Airscale")
# 返回99614055字节

# 获取存储统计信息
RanMonitor.Nokia.Airscale.ConfigStorage.get_storage_stats("ONS-
Lab-Airscale")
# 返回%{version_count: 655, total_size_bytes: 99614055, ...}

# 清理旧版本配置 50 MB 以下
RanMonitor.Nokia.Airscale.ConfigStorage.cleanup_old_versions("ONS-
Lab-Airscale", 50 * 1024 * 1024)
# 返回{:ok, 345, 50000000} - 删除345个配置，释放50MB

# 清理旧版本配置 100 MB 以下
RanMonitor.Nokia.Airscale.ConfigStorage.cleanup_old_versions("ONS-
Lab-Airscale")
# 返回{:ok, 0, 0} - 没有删除任何配置
```



- **Web UI** □□ - □□□□□□□□
- **AirScale** □□□□ - □□□□
- □□□□□□ - □□□□□□
- □□□□□□□□ - □□□□

部署

前提

RAN Monitor には、InfluxDB が必要です。

インストール

インストール

- URL: <https://localhost:9443>
- インストール
- eNodeB インストール

インストール

- eNodeB
- “”
-
-

インストール

-
- eNodeB

□□

□□□□□□□□□□ eNodeB □□□□□□□□□□

□ □□

□□□□□□

- □□□□□□: 720 □□□30 □□
- □□□: □ `config/config.exs` □□□
- □□: □□□□□□□□□□□□□□ eNodeB

□□ **eNodeB** □□□

- □□□□: □□□ eNodeB □□□□□□□□
- □□□□□□: □□□□ `airscales` □□
- □□: □□□□

□□□□

- □□: □□□□□□□□
- □□□□□□: `RanMonitor.Data.RetentionCleanupWorker`
- □□ **eNodeB**: □□□□□□□□□□
- □□: □□□□□□□□□□□□□□□□□□

□□□□□□

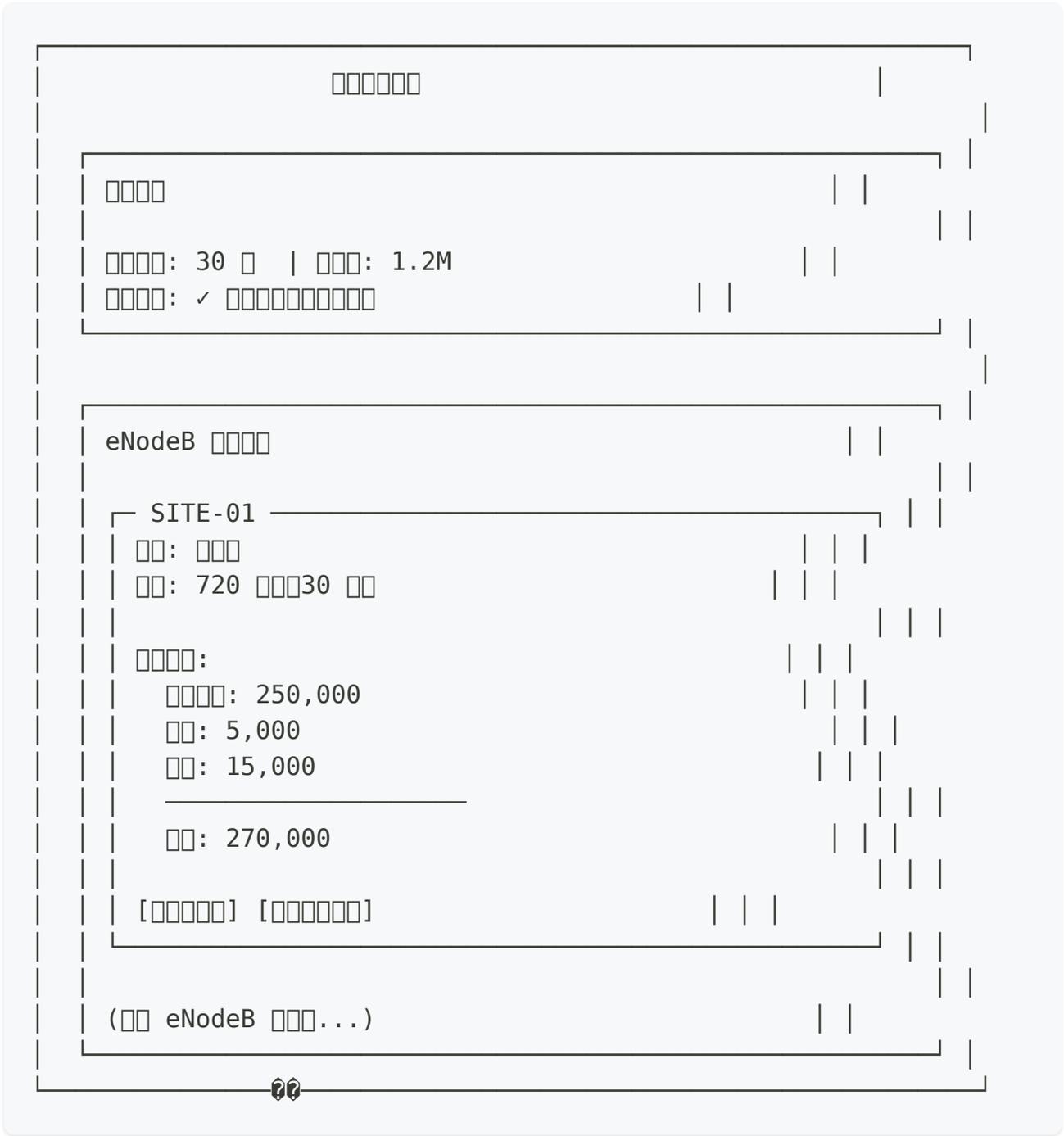
- □□□□□□: □□□□□□□□□□□□□□□□□□□
 - □□□□□□
 - □□
 - □□
 - □□□□: □□□□ eNodeB □□□□□□
 - □□: □□□□□□□□□□
-

□ □□□□

□□□□□

□□□□□□□□□□□□□□□□ *eNodeB* □□□□□□□□□□

□□□□□



000000

- 00 (✓): eNodeB 000000000000
- 00 (X): eNodeB 00000000
- 00: 00000000 eNodeB 0000

□□□□

□□	□□	□□
□□□□□	□□□□□	□□□□□□□□□□□□
□□□□□□□	□□□□□	□□□ eNodeB □□□□□ (△ □□□□□□)
□□	□□□□□	□□□□□□□□□□□□

⚙️ □□

□□□□□□□

□□ `config/config.exs`:

```
config :ran_monitor,  
  ecto_repos: [RanMonitor.Repo],  
  generators: [context_app: :ran_monitor],  
  data_retention_hours: 720 # 30 □□□□□□□□
```

□□□□□□

□□	□□	□	□□□□
1 □□	1	0.04	□□□□□
1 □	24	1	□□□□
7 □	168	7	□□□□
14 □	336	14	□□□□
30 □	720	30	□□□□□□□□
90 □	2160	90	□□□□
180 □	4320	180	□□□□
1 □	8760	365	□□□□

□□□□

□□□□□□□□

```
export DATA_RETENTION_HOURS=1440 # 60 □  
mix phx.server
```

□ □□□□

□□□□□□

1. □□□□
 - ├ □□□□ → InfluxDB
 - ├ □□□□ → InfluxDB
 - └ □□ → InfluxDB
2. □□□□□□□□
 - ├ RetentionCleanupWorker □□
 - ├ □□□ eNodeB□
 - | └ □□□□□□□□ eNodeB □□□□
 - | └ □□□□□□□□
 - | └ □□□□□□□□□□□□
 - └ □□□□
3. □□□□□□□□
 - ├ □□□ UI □□□□□□
 - ├ □□□□□□□□
 - ├ □□□□□□□□
 - └ □□□□□□□□□□
4. □□
 - └ □□ UI □□□□□□□□

□□□□

□□ **eNodeB** □□□□□□

```
effective_retention = case airscales.retention_hours do
  nil -> Config.data_retention_hours() # □□□□□720h□
  hours -> hours # □□□□ eNodeB □□□□□
end
```

□□□

- □□□□□720 □□□30 □□

- eNodeB "SITE-01" 168 7
- eNodeB "SITE-02" nil → 720

1

$current_time - (retention_hours * 3600)$

30

| 2025-12-11 10:00:00

| 720 30

| 2025-11-11 10:00:00

| $current_time < target_time$

2

3

4

[RetentionCleanupWorker] 5

[RetentionCleanupWorker] SITE-01 720h

[RetentionCleanupWorker] SITE-01 15,000

[RetentionCleanupWorker] 5 75,000

5

6

1. 5
2. eNodeB
3. "SITE-01"

7

- 5

- InfluxDB
 -
-

```
alias RanMonitor.Data.RetentionPolicy
alias RanMonitor.Database.Nokia

# eNodeB
airscale = Nokia.get_airscale!(1)
hours = RetentionPolicy.get_retention_hours(airscale)
# => 720

# eNodeB
counts = RetentionPolicy.get_record_counts("SITE-01")
# => %{"PerformanceMetrics" => 250000, "Configuration" => 5000,
"Alarms" => 15000}

#
total = RetentionPolicy.get_total_record_count("SITE-01")
# => 270000

#
{:ok, deleted_count} = RetentionPolicy.delete_old_records("SITE-01", 720)
# => {:ok, 50000} (50k)

# eNodeB
{:ok, deleted_count} = RetentionPolicy.clear_all_records("SITE-01")
# => {:ok, 270000} (270k)
```

□□□□□□

□□ lib/ran_monitor/data/retention_cleanup_worker.ex:

```
# □ 1 □□□3600000ms□□□□ 30 □□□1800000ms□  
@cleanup_interval_ms 1800000 # 30 □□
```

□□□□□□

```
mix compile
```

□□□□□□

□□□□□□□□

```
SELECT name, retention_hours FROM airscales;
```

□□□□□□□□

```
UPDATE airscales  
SET retention_hours = 168  
WHERE name = 'SITE-01';
```

□ □□□□

□□□□□□

□□□ < 7 □□

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

7-30

-
-
-

> 30

-
-
-

	7-14	
	30-60	
	90	
	180-365	

- 1000 \approx 1-5 KB
- 100 \approx 1-5 GB
- \times =

##

```
# 查看 InfluxDB 数据目录
influx bucket list

# 查看磁盘使用情况
df -h /path/to/influxdb/data
```

目录结构

数据目录

- 存储 InfluxDB 数据
- 存储 InfluxDB 元数据
- 存储 InfluxDB 配置

配置

- 配置文件: `influxdb.conf`
- 环境变量: `INFLUXDB_*`
- 系统配置: `/etc/influxdb/`

安装

1. 安装 **InfluxDB** 二进制文件
 2. 配置 InfluxDB
 3. 启动 InfluxDB 服务
 4. 验证 InfluxDB 安装
 5. 配置 InfluxDB 与 Grafana 集成
-

1. 背景

1.1 背景

背景

- 背景
- 背景

背景

1. 背景 `ps aux | grep mix`
2. 背景 `RetentionCleanupWorker` 背景
 - 背景 `[RetentionCleanupWorker]` 背景
3. 背景 InfluxDB 背景
 - 背景 InfluxDB 背景 `https://localhost:9443/nokia/influx`
4. 背景 背景
 - 背景 `config/config.exs` 背景 `data_retention_hours`

1.2 背景

背景

- 背景“背景”背景
- 背景

背景

1. 背景 InfluxDB 背景
 - 背景
2. 背景 背景
 - 背景“背景”背景
3. 背景 背景
 - 背景 `[RetentionPolicy]` 背景
4. 背景 eNodeB 背景
 - 背景 eNodeB 背景

□□□□□□□□□□

□□□

- □□□□□□
- □□□□□□

□□□□□

1. □□□□□□□□□□
2. □□ 5-10 □□□□□□
3. □□□□□□□□
 - □□ @cleanup_interval_ms □□□ 1 □□□
4. □□□□□□ eNodeB □□□□□

□□□□□□□□□

□□□

- □□□□□ InfluxDB UI □□□
- “□□□□□□”□□□□□□

□□□□□

1. □□“□□”□□□□□
2. □□ InfluxDB □□□
 - □□□□□□□□□□□□□□□□
3. □□□□□□□□
 - InfluxDB □□□□□□□□□□
4. □□ eNodeB □□□□□□□□□□
 - □□□□□□□□□□

□ □□□□□

-   - □□□□□□□□

- **Web UI** `ui` - `ui`
 - `ui` - `ui`
 - `ui` - `ui`
 - **Grafana** `grafana` - `grafana`
-

📄 📄📄📄

- `ui`: `https://localhost:9443/nokia/retention`
 - **eNodeB** `enb`: `https://localhost:9443/nokia/status`
 - **InfluxDB** `influx`: `https://localhost:9443/nokia/influx`
 - `logs`: `https://localhost:9443/nokia/logs`
-

📄 📄📄📄

`ui`

`ui` `ui`

📄📄📄📄📄 **eNodeB** 📄📄📄📄📄

`ui` `ui` eNodeB `ui`

`ui`

`ui` `ui` `@cleanup_interval_ms`

`ui`

`ui` `ui`

□ □□□□

□□□□□

□□	□□
<code>lib/ran_monitor/database/nokia/airscale.ex</code>	□□ <code>retention_hours</code> □□
<code>lib/ran_monitor/config/config.ex</code>	□□ <code>data_retention_hours()</code> □□□
<code>config/config.exs</code>	□□□□□□□□□□□□□□
<code>lib/ran_monitor/application.ex</code>	□□□□□□□□□□□□□□

□□□□□

□□
<code>lib/ran_monitor/data/retention_policy.ex</code>
<code>lib/ran_monitor/data/retention_cleanup_worker.ex</code>
<code>lib/ran_monitor/web/live/retention_policy_live.ex</code>
<code>priv/repo/migrations/20251211065257_add_retention_hours_to_airscales.</code>

□□□□

RetentionPolicy □□□

- `get_retention_hours(airscale)` - □□□□□□
- `get_record_counts(airscale_name)` - □□□□□□

- `get_total_record_count(airscale_name)` - 00
- `delete_old_records(name, hours)` - 000000
- `clear_all_records(name)` - 0000

RetentionCleanupWorker GenServer

- `start_link(opts)` - 00000000
 - `init(:ok)` - 00000000
 - `handle_info(:cleanup, state)` - 00000000
-

0 00

00000000

1. 000000

```
mix ecto.migrate
```

2. 00000000

```
mix phx.server
```

3. 000000

- 000 `https://localhost:9443/nokia/retention`
- 00000000000000

□□□□

1. □□□□□□

- □□□□□□□□720 □□□
- □□□□ eNodeB □□□□□

2. □□□□□□□□□□

- □ `config/config.exs` □□□□□□□□
- □□□ UI □□□□ eNodeB

3. □□□□□

- □□□□□□ `[RetentionCleanupWorker]` □□
 - □□□□□□□□□□□□
-

□ □□

□□□□□

1. □□□□ □□ [RetentionPolicy] □ [RetentionCleanupWorker] □□
2. □□□□□□ □□□□□□□□□□□□□□□□
3. □□□□□□□ □□□□□□□□□□
4. □□□□□ □□□□□□□□□□□□□□

□□□□

□□□

- UI □□□□□□□□□□
- □□□□ eNodeB □□
- □□□□□□
- □□□□□□□□□□
- □□□□

□ □□□□

□□□□

- **InfluxDB v2.x:** □□□□□□□□□□□□□□
- □□□□: □□□□□□□□
- □□: □□□□□□□□
- □□□□: □□□□□□□□□□

□□□

- InfluxDB □□
- Elixir GenServer □□
- Phoenix LiveView

📊 RAN Monitor

RAN Monitor 📊📡📶📱📡📶📱

📊📡📶📱📡📶📱 RAN Monitor

📋

1. 📊
 2. 📡📶
 3. 📱📡📶
 4. 📊
 5. 📡📶
-

📋

📊📡📶📱 RAN Monitor 📊📡📶📱📡📶📱

📋📋📋

📊📡📶📱

- ✓ 📊📡📶📱MySQL📊InfluxDB📊
- ✓ 📊📡📶📱 RAN Monitor
- ✓ 📊 RAN Monitor 📊📊
- ✓ 📊📡📶📱 Nokia AirScale 📊
- ✓ 📊📡📶📱 InfluxDB
- ✓ 📊 Web UI 📊

📊📡📶📱 📊📡📶📱 30-60 📊

□□□□

□□ RAN Monitor □□□□□□□□□□□□

□□□□□□

MySQL □□□□□□

- □□ MySQL 5.7+ □ MariaDB 10.3+
- □□□□ RAN Monitor □□□□□□□□
- □□ CREATE □ SELECT □ INSERT □ UPDATE □ DELETE □
- □□□□ RAN Monitor □□□□□□□□
- □□□□□□□□□□□□□□

InfluxDB □□□□□□

- □□ InfluxDB 1.8+ □ 2.0+
- □□□□ RAN Monitor □□□□□□□□
- Bucket/□□□□□□□□□□□□□□□□
- API □□□□□□□□□□□□ InfluxDB 2.x□
- □□□□□□□□□□□□□□□□□□

RAN Monitor □□□

- □□□□ Linux □ Ubuntu 20.04+ □ CentOS 8+ □□□□
- RAM □□□ 4GB □□□ 8GB
- CPU □□□ 2 □□□□□ 4+ □□
- □□□□□□□□□□□□ 20GB
- □□□□□□□ MySQL □ InfluxDB □□□

□□□□

□□□□

- RAN Monitor → Nokia AirScale □□□□□ 8080□
- Nokia □□ → RAN Monitor □□□ 9076 □□ Webhook□

- RAN Monitor → MySQL 3306
- RAN Monitor → InfluxDB 8086
- → RAN Monitor Web UI 9443

Ports

- 8080 InfluxDB
- 9076 Webhook
- 9443 HTTPS Web UI
- MySQL InfluxDB

Nokia

Ports

- IP -
- - 8080
- - WebLM
- - ping
- -

Ports

- - PEM
- - DER
- Nokia OpenSSL

Grafana

- Grafana 8.0+
 - InfluxDB
 -
-

000000

1 00000000

1.1 00 MySQL 000

0 RAN Monitor 000000

```
CREATE DATABASE ran_monitor CHARACTER SET utf8mb4 COLLATE  
utf8mb4_unicode_ci;
```

0000000000000000

```
CREATE USER 'ran_monitor_user'@'%' IDENTIFIED BY  
'secure_password';  
GRANT CREATE, SELECT, INSERT, UPDATE, DELETE ON ran_monitor.* TO  
'ran_monitor_user'@'%;  
FLUSH PRIVILEGES;
```

0000 RAN Monitor 00000000

```
mysql -h <mysql-host> -u ran_monitor_user -p ran_monitor
```

1.2 00 InfluxDB

00 InfluxDB 1.x 00000000

```
influx -execute 'CREATE DATABASE "nokia-monitor"'
```

00 InfluxDB 2.x 000000 bucket

```
influx bucket create -n nokia-monitor -o your-org
```

000000000000 API 000 InfluxDB 2.x 00

```
influx auth create --org your-org --write-buckets
```

□□□□□□□□□□

1.3 □□□□□□

□□□□□□□□□□□□

```
# □□□□□□□□□□  
ping 10.7.15.66  
  
# □□□□□□□□□□  
telnet 10.7.15.66 8080
```

□□ MySQL □ InfluxDB □□□□□□

```
# □□ MySQL □□□  
telnet <mysql-host> 3306  
  
# □□ InfluxDB □□□  
curl http://<influxdb-host>:8086/ping
```

□ 2 □□□□ RAN Monitor

□□□□□□ `config/runtime.exs` □□□□□□

2.1 □□□□□□

□□ `config/runtime.exs` □□□ MySQL □□□

```
config :ran_monitor, RanMonitor.Repo,  
  username: "ran_monitor_user",  
  password: "secure_password",  
  hostname: "mysql-host",  
  database: "ran_monitor",  
  stacktrace: true,  
  show_sensitive_data_on_connection_error: true,  
  pool_size: 10
```

2.2 InfluxDB ☐☐

☐☐ InfluxDB ☐☐☐

```
config :ran_monitor, RanMonitor.InfluxDbConnection,  
  auth: [  
    username: "monitor",  
    password: "influx_password" # ☐ InfluxDB 2.x ☐ API ☐☐  
  ],  
  database: "nokia-monitor",  
  host: "influxdb-host"
```

2.3 Web ☐☐☐☐

☐☐ Web ☐☐☐

```

# SOAP/API
config :ran_monitor, RanMonitor.Web.Endpoint,
  http: [ip: {0, 0, 0, 0}, port: 8080],
  check_origin: false,
  secret_key_base: "generate_with_mix_phx_gen_secret",
  server: true

# Web UI HTTPS
config :control_panel, ControlPanelWeb.Endpoint,
  url: [host: "0.0.0.0", port: 9443, scheme: "https"],
  https: [
    ip: {0, 0, 0, 0},
    port: 9443,
    keyfile: "priv/cert/omnitouch.pem",
    certfile: "priv/cert/omnitouch.crt"
  ]

# Webhook
config :ran_monitor, RanMonitor.Web.Nokia.Airscale.Endpoint,
  url: [host: "0.0.0.0"],
  http: [ip: {0, 0, 0, 0}, port: 9076],
  server: true

```

2.4 Nokia

```

config :ran_monitor,
  general: %{
    mcc: "001", # 00000000
    mnc: "001"  # 00000000
  },
  nokia: %{
    ne3s: %{
      webhook_url: "http://<ran-monitor-ip>:9076/webhook",
      private_key: Path.join(Application.app_dir(:ran_monitor,
"priv"), "external/nokia/ne.key.pem"),
      public_key: Path.join(Application.app_dir(:ran_monitor,
"priv"), "external/nokia/ne.cert.der"),
      reregister_interval: 30
    },
    airscales: [
      %{
        address: "10.7.15.66",
        name: "Site-A-BS1",
        port: "8080",
        web_username: "admin",
        web_password: "password"
      }
    ]
  }
}

```

2.5 SSL 証明書

HTTPS Web UI SSL 証明

```

# 証明書/鍵
openssl req -newkey rsa:2048 -nodes -keyout
priv/cert/omnitouch.pem \
  -x509 -days 365 -out priv/cert/omnitouch.crt

```

証明書 CA 証明書

証明書 証明書

3 部署

RAN Monitor

3.1 環境構築

環境構築

```
mix ecto.migrate
```

環境構築

3.2 RAN Monitor

環境構築

```
mix phx.server
```

環境構築

```
MIX_ENV=prod mix release  
_build/prod/rel/ran_monitor/bin/ran_monitor start
```

3.3 確認

環境構築

```
[info] Running RanMonitor.Web.Endpoint with cowboy  
[info] Running ControlPanelWeb.Endpoint with cowboy  
[info] Running RanMonitor.Web.Nokia.Airscale.Endpoint with cowboy  
[info] Starting RAN Monitor Manager  
[info] Connecting to InfluxDB...  
[info] InfluxDB connection established  
[info] Attempting registration with device: Site-A-BS1  
[info] Successfully registered with Site-A-BS1
```

確認

- Web 管理
 - 監視
 - InfluxDB 管理
 - 設定
-

00

4 監視

監視

4.1 Web UI

監視

`https://<ran-monitor-ip>:9443`

監視 RAN Monitor 監視

4.2 監視

Web UI

1. 監視
2. 監視
3. 監視“監視”監視
4. 監視“監視”
5. 監視

監視/監視

- 監視
- 監視
- 監視
- 監視

4.3 InfluxDB

Web UI

1. InfluxDB
- 2.
- 3.
4. “” “” “”

InfluxDB

```
# InfluxDB 1.x
influx -database 'nokia-monitor' -execute 'SELECT COUNT(*) FROM PerformanceMetrics'

# InfluxDB 2.x
influx query 'from(bucket:"nokia-monitor")
  |> range(start: -1h)
  |> filter(fn: (r) => r._measurement == "PerformanceMetrics")
  |> count()'
```

4.4

Web UI

- 1.
2. “”
- 3.

mix phx.server

4.5

Web UI

- 1.
2.
 - o

- 配置
 - 部署
 - 测试
-

部署

部署 RAN Monitor 部署

部署

1. 部署

- `config/runtime.exs` 配置
- 部署
- 部署 `部署`

2. 部署 Grafana 部署

- 部署 Grafana
- 部署 InfluxDB 部署
- 部署
- 部署 Grafana 部署

3. 部署

- 部署
- 部署
- 部署 `部署`

4. 部署

- 部署 Web UI 部署
- 部署 Grafana 部署
- 部署
- 部署 `部署`

□□□□

□□□□

- □□ Web UI □□ □□□□□□
- □□ □□□□□□ □□□□□□
- □□ □□□□□□ □□□□□□

□□□□

- □□□□□□□□ Web UI
- □□□□□□□□□□□□□□□□□□
- □□□□□□□□□□

□□□□

- □ Grafana □□□□□□□□
- □□□□□□□□□□
- □□□□□□□□Slack□□□□□□□□PagerDuty□

□□□□

- □ CA □□□□□□□□□□□□
- □□□□□□□□□□
- □□ `config/runtime.exe` □□□□□□
- □□□□□□□□

□□□□

□□□□

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

□□□□

- □□□□□□□□
- □□ 24 □□□□□□
- □□□□□□
- □□□□□□□□

□□□□

- □□ Web UI □□□□□□□□
 - □□□□□□□□
 - □□□□ Grafana □□□□□□
 - □□□□□□
-

□□□□

□□□□□□

- □□□□□□ - □□□□□□□□
- Web UI □□ - □□□□□□
- □□□□□□ - □□□□□□

□□

- □□□□ - □□□□□□□□
- □□□□□□□□ - □□□□□□
- AirScale □□ - □□□□

□□□□□□

□□□□□□

- □□□□□□□□ping□
- □□□□□□□□

- 8080 8086
- 8086

InfluxDB

- InfluxDB
-
- API
- `curl http://<influxdb-host>:8086/ping`

Web UI

- HTTPS 9443
- SSL
- Web
-

- -
- **Web UI** -
- -
- -
- **AirScale** -
- **Grafana** -
- -
- -
- -

Grafana □□□□□□□□

□□□□ **RAN** □□□□□□□□□□

Grafana □□□□□□□□ *KPI* □□□□□□□□

□□

1. □□
 2. Grafana □ InfluxDB □□
 3. □□□□□
 4. □□□□□□□
 5. □□□□
 6. □□□□□□□
 7. □□□□□
 8. □□□□
-

□□

Grafana □□□□□□□□□□ RAN □□□□□□□□□□□□□□□□□□□□

□□□□□

□ Grafana □□□ InfluxDB □□□□□

1. □□□□□

- Grafana → □□ → □□□

2. □□□□□□

- □□“□□□□□□”
- □□“InfluxDB”

3. □□□□□

□□	□	□□
□□	RAN Monitor	Grafana □□□□□□
URL	<code>http://influxdb:8086</code>	□□□□□ Grafana □□
□□	□□□□□□□□	Grafana □□□□□□□□
□□	omnitouch	□□ InfluxDB □□
□□	□API □□□	□□ API □□□□
□□□□□□	ran_metrics	RAN Monitor □□□□□□
□□□□□□	10s	□□□□□□

4. □□□□□

- □□“□□”□□
- □□□“□□□□□□□□□□”

□□□ InfluxDB □□□□□□□□□□□□□□□□□□□□ RAN Monitor □□□□□□□□ InfluxDB □□□□□□□□□□ □□□□□□□□ □ AirScale □□□ □□□□□□□□□□□□

Flux

Grafana Flux InfluxDB

```
from(bucket:"ran_metrics")
  |> range(start: -7d, stop: now())
  |> filter(fn: (r) => r._measurement == "PerformanceMetrics")
  |> filter(fn: (r) => r.device == "SITE_A_BS1")
  |> group(by: ["_field"])
  |> aggregateWindow(every: 1h, fn: mean)
```

- `from()` -
 - `range()` -
 - `filter()` -
 - `group()` -
 - `aggregateWindow()` -
-

□□□□□□□□

□□□□□□□□

□□□□
(CEO/□□)

□□

□□□□□□
(NOC □□)

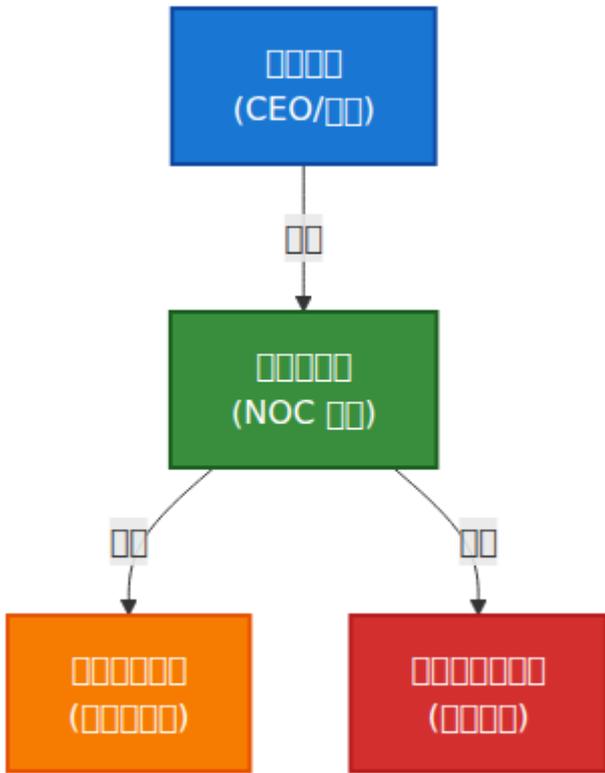
□□

□□

□□□□□□
(□□□□□)

□□□□□□
(□□□□)

組織図



業務内容

〇〇〇〇〇〇〇〇〇〇〇〇〇〇〇〇〇〇

〇〇〇〇〇〇〇〇〇〇〇〇〇〇

〇〇〇〇〇〇			
〇〇〇〇	〇〇〇〇	〇〇〇〇〇〇〇〇	
48/50 (96%)	3 〇〇〇〇〇	98.5%	
〇〇〇〇〇 [2000] 〇〇〇			

備考

- 〇〇〇〇〇〇〇〇 < 10 〇〇〇〇
- 〇〇/〇〇〇〇〇〇〇〇〇〇〇〇〇〇

- 0000000000000000

0000000000000000

0000000000000000

```

| 0000007 00 |
| [0000000000/0000] |
| 000250 Gbps000000 2 00 |
| 00080 Gbps000000 3 00 |

```

000

- 000000
- 000000
- 000000
- 0000

🔗🔗0000000000000000

00000000

```

| 0000000000000000 |
| 00 | 00 | 00 | 0000 |
| 0 | SITE_A_BS1 | 0000 | 45 00 |
| 0 | SITE_B_BS2 | 00 | 2 00 |

```

000

- 000000
- 000000
- 0000000000000000

Queries

Queries for Nokia performance metrics in InfluxDB  

Cell Availability

Queries for the last 24 hours

```
from(bucket:"ran_metrics")
  |> range(start: -24h)
  |> filter(fn: (r) => r._measurement == "PerformanceMetrics")
  |> filter(fn: (r) => r._field == "CellAvailability")
  |> group(by: ["device"])
  |> aggregateWindow(every: 1h, fn: mean)
  |> yield(name: "cell_availability")
```

Queries

- SLA queries
- Performance metrics
- > 99.5% queries

Queries for the last 7 days

```
from(bucket:"ran_metrics")
  |> range(start: -7d)
  |> filter(fn: (r) => r._measurement == "PerformanceMetrics")
  |> filter(fn: (r) => r._field =~ /Throughput.*/)
  |> group(by: ["device", "_field"])
  |> aggregateWindow(every: 10m, fn: mean)
  |> yield(name: "traffic_trend")
```

Queries

- Queries

- 100%
- 100%

100%

```
from(bucket:"ran_metrics")
  |> range(start: -1h)
  |> filter(fn: (r) => r._measurement == "PerformanceMetrics")
  |> filter(fn: (r) => r._field == "DLResourceUtilization")
  |> filter(fn: (r) => r.device == "SITE_A_BS1")
  |> aggregateWindow(every: 10s, fn: last)
  |> yield(name: "dl_resource")
```

100%

- 100%
- 80% 95% 100%
- 100%

100%

100% **24** 100%

```
from(bucket:"ran_metrics")
  |> range(start: -24h)
  |> filter(fn: (r) => r._measurement == "Alarms")
  |> filter(fn: (r) => r.status == "active")
  |> group(by: ["severity"])
  |> count()
  |> yield(name: "alarm_count")
```

100%

- 100%
- 100%
- 100%

100%

```
from(bucket:"ran_metrics")
  |> range(start: -7d)
  |> filter(fn: (r) => r._measurement == "Alarms")
  |> group(by: ["severity"])
  |> aggregateWindow(every: 1h, fn: count)
  |> yield(name: "alarm_rate")
```

□□□

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

□□□□□□□□

```
from(bucket:"ran_metrics")
  |> range(start: -7d)
  |> filter(fn: (r) => r._measurement == "Alarms")
  |> group(by: ["alarm_description"])
  |> count()
  |> sort(columns: ["_value"], desc: true)
  |> limit(n: 10)
  |> yield(name: "top_alarms")
```

□□□

- □□□□□□
- □□□□□□□□□□
- □□□□□□□□□□

□□□□□

□□□□□□□□□□□□□□

```

from(bucket:"ran_metrics")
  |> range(start: -30d)
  |> filter(fn: (r) => r._measurement == "PerformanceMetrics")
  |> filter(fn: (r) => r._field == "CellAvailability")
  |> filter(fn: (r) => r.device == "SITE_A_BS1")
  |> aggregateWindow(every: 1h, fn: mean)
  |> statefulWindow(every: 1h, period: 24h)
  |> map(fn: (r) => ({r with _value: float(v: r._value)}))
  |> reduce(fn: (r, acc) => ({
    x: acc.x + [float(v: r._time)],
    y: acc.y + [r._value]
  }),
  initial: {x: [], y: []})
  |> yield(name: "availability_forecast")

```

□□□

- □□ SLA □□□□□□□□
- □□□□□□
- □□□□

□□□□□□□□□□□□

```

from(bucket:"ran_metrics")
  |> range(start: -7d)
  |> filter(fn: (r) => r._measurement == "PerformanceMetrics")
  |> filter(fn: (r) => r._field =~ /HandoverSuccess|Traffic/)
  |> group(by: ["device", "_field"])
  |> aggregateWindow(every: 1h, fn: mean)
  |> pivot(rowKey: ["_time"], columnKey: ["_field"], valueColumn:
  "_value")
  |> map(fn: (r) => ({r with correlation: float(v:
r.HandoverSuccess) * float(v: r.Traffic)}))
  |> yield(name: "ho_traffic_correlation")

```

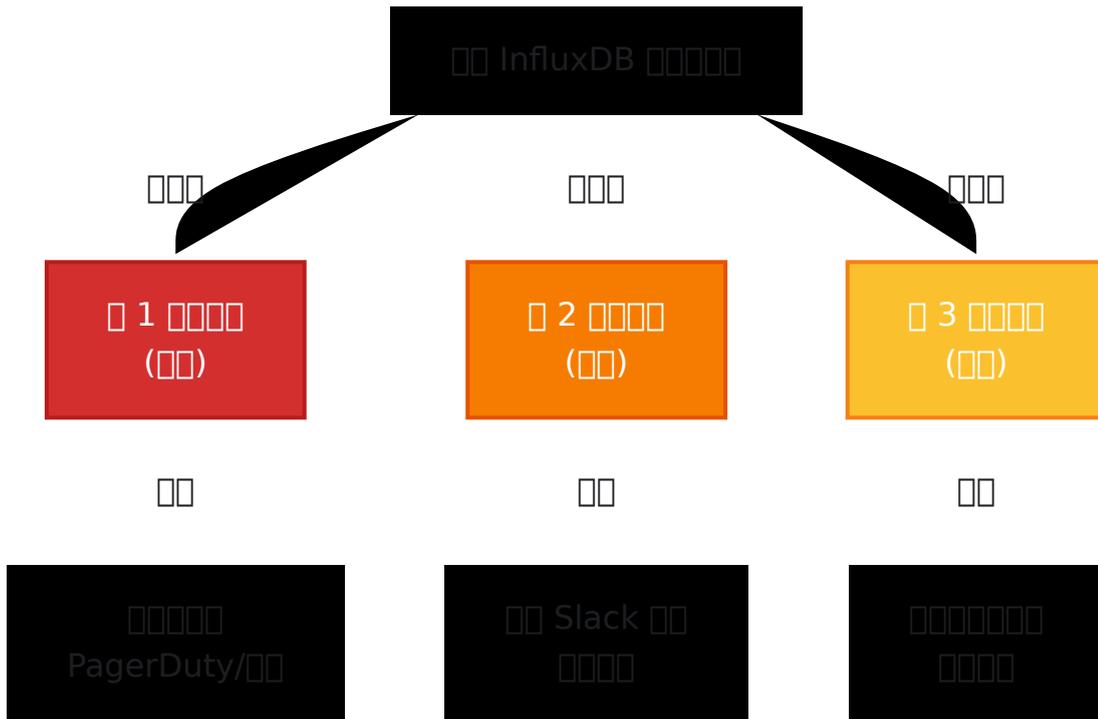
□□□

- □□□□□□□□□□□□
- □□□□□□□□

- 監視システム

監視システム

監視システム



Grafana 監視システム

1 監視システム

1. 監視システム
2. 監視システム
3. 監視 → 監視
4. 監視 → 監視 → 監視システム

2 監視システム

1 監視システム

```
CellAvailability < 95%
  15
  1
  15
```

•

- 95% SLA 99.5%
- 15
-

2

```
count(active_alarms) > 10
  5
  2
  5
```

•

- 10
- 5
-

3

```
DLResourceUtilization > 90%
  30
  5
  30
```

•

-
- 30
- 5

- Slack + Grafana
- Grafana URL / Slack
- **SLA** 確保

概要

Slack 連携

1. Grafana Slack 連携
2. Grafana URL
3. Grafana 設定 → Slack
4. Slack "Slack" 連携
5. Grafana URL
6. Slack

Slack 連携内容

Slack チャンネル - SITE_A_BS1_Cell1
 毎日 04:45 開始
 毎日 20:00 終了
 毎日 2:15 開始

[Slack] [Slack] [Slack]

PagerDuty 連携

1. Grafana PagerDuty 連携
2. Grafana 設定 → PagerDuty
3. Grafana "PagerDuty" 連携
4. Grafana
5. Grafana
 - Grafana → PagerDuty
 - Grafana → Slack
 - Grafana → Slack

詳細

1. Grafana SMTP
2. →
3. “”
- 4.
5. CSV

1

5

4

- /
-
- %
- Gbps

2.

- 7
- 7
-

3.

-
- 7

4.

- CPU %

3.

- / 24
- 24
-

4.

-
-

5.

- IP/
-
-

6.

- /
-
-

4G 4G

3

1

1.

- o
- o +
- o

2.

- o SINR
- o RSRP
- o

3.

- RLC փոխանակում
- RRC փոխանակում
- փոխանակում

4. փոխանակում

- փոխանակում
- փոխանակում

5. փոխանակում

- փոխանակում
- փոխանակում
- փոխանակում

Պատ 4 փոխանակում

Պատ փոխանակում

փոխանակում 5

Պատ փոխանակում

Պատ

1. փոխանակում

- փոխանակում
- փոխանակում/փոխանակում

2. փոխանակում

- փոխանակում
- փոխանակում

3. փոխանակում

- փոխանակում
- փոխանակում
- փոխանակում

4. 測試項目

- 測試 4 個測試項目
- 測試
- 測試

5. 測試項目

- 測試 30 個測試項目
- 測試
- 測試

6. 測試項目

- 測試
- 測試
- 測試/測試

測試 5 Nokia AirScale 測試

測試 RF 測試

測試 30 個

測試 Nokia 測試 KPI

測試 Nokia 測試 AirScale 測試 Nokia 測試

測試 1 測試

測試 PRB 測試

```

// PRB
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8011C37")
  |> filter(fn: (r) => r._field == "counterValue")
  |> group()
  |> mean()
  |> map(fn: (r) => ({ r with _value: r._value / 10.0})) // 
[]
  |> rename(columns: {_value: "PRB"})

// PRB
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8011C24")
  |> filter(fn: (r) => r._field == "counterValue")
  |> group()
  |> mean()
  |> map(fn: (r) => ({ r with _value: r._value / 10.0})) // 
[]
  |> rename(columns: {_value: "PRB"})

```

PRB PRB

- PRB0-70%
- PRB70-85%
- PRB85-100%

PRB 2PRB

PRB PDCP PRB

```
// 00000
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8012C26")
  |> filter(fn: (r) => r._field == "counterValue")
  |> map(fn: (r) => ({ r with _value: r._value / 1000.0})) // 000
Mbps
  |> rename(columns: {_value: "000 Mbps"})
```

```
// 00000
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8012C23")
  |> filter(fn: (r) => r._field == "counterValue")
  |> map(fn: (r) => ({ r with _value: r._value / 1000.0})) // 000
Mbps
  |> rename(columns: {_value: "000 Mbps"})
```

00000000

- **M8012C26** - PDCP 000000000000/00
- **M8012C23** - PDCP 000000000000/00

00 3000 **UE** 00000000

0000000000

```
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8018C1")
  |> filter(fn: (r) => r._field == "counterValue")
  |> rename(columns: {_value: "000 UE"})
```

00000000

- **M8018C1** - eNB UE

4

```
import "strings"

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8020C3" or
                    r["metricCounter"] == "M8020C6" or
                    r["metricCounter"] == "M8020C4")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> pivot(rowKey:["_time"], columnKey: ["metricCounter"],
valueColumn: "_value")
  |> map(fn: (r) => ({
    _time: r._time,
    "": 100.0 * r.M8020C3 / (r.M8020C6 - r.M8020C4)
  })))
```

- **M8020C3** -
- **M8020C6** -
- **M8020C4** -

< 99%

5 **PRB**


```

import "strings"

// PDCP 70000000
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8012C26")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> rename(columns: {"_value": "PDCP 70000000"})

// PDCP 70000000
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8012C23")
  |> filter(fn: (r) => r._field == "counterValue")
  |> group()
  |> rename(columns: {"_value": "PDCP 70000000"})

```

70RSSI

```

import "strings"

// PUCCH RSSI
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8005C0")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> rename(columns: {"_value": "PUCCH RSSI"})

// PUCCH RSSI
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8005C2")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> rename(columns: {"_value": "PUCCH RSSI"})

// PUCCH RSSI
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8005C1")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> rename(columns: {"_value": "PUCCH RSSI"})

```

□□□□□□

- **M8005C0** - PUCCH □□ RSSI□dBm□
- **M8005C1** - PUCCH □□ RSSI□dBm□

- **M8005C2** - PUCCH RSSI dBm

8

PDCP SDU

```
import "strings"

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8001C2")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> rename(columns: {"_value": ""})
```

- **M8001C2** - PDCP SDU DTCH

9 RRC

```

import "strings"

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8013C5" or
                    r["metricCounter"] == "M8013C17" or
                    r["metricCounter"] == "M8013C18" or
                    r["metricCounter"] == "M8013C19" or
                    r["metricCounter"] == "M8013C34" or
                    r["metricCounter"] == "M8013C31" or
                    r["metricCounter"] == "M8013C21" or
                    r["metricCounter"] == "M8013C93" or
                    r["metricCounter"] == "M8013C91")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> pivot(rowKey:["_time"], columnKey: ["metricCounter"],
valueColumn: "_value")
  |> map(fn: (r) => ({
    _time: r._time,
    "RSSI": 100.0 * r.M8013C5 / (r.M8013C17 + r.M8013C18 +
r.M8013C19 + r.M8013C34 + r.M8013C31 + r.M8013C21 + r.M8013C93 +
r.M8013C91)
  })))

```

📊

- **M8013C5** - 📶
- **M8013C17-M8013C93** - 📶

📶 📶 < 95%

📶 **10** 📶 **VSWR** 📶

📶

```

import "strings"

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M40001C0")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${RadioKey}"))
  |> map(fn: (r) => ({
    r with
    "DN": strings.split(v: r["DN"], t: "/")[5],
    "VSWR": r._value / 10.0
  }))
  |> group()
  |> pivot(rowKey: ["_time"], columnKey: ["DN"], valueColumn:
"VSWR")

```

□□□□□□□

- **M40001C0** - □□□□□□□ VSWR□0.1 □□□

□□□□□ VSWR > 2.0

□□ **11**□□□□□□□□□□

□□□□□□□

```

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M40002C2")
  |> filter(fn: (r) => r._field == "counterValue")
  |> group()
  |> map(fn: (r) => ({ r with _value: r._value / 100000.0 }))
  |> rename(columns: {"_value": "□□"})

```

□□□□□□□

- **M40002C2** - 100000

Grafana

- **{Airscale}** -
- **{CellKey}** -
- **{RadioKey}** - VSWR

1. **PRB** - DL UL PRB > 85% 5
2. - < 99% 10
3. - RRC < 95% 5
4. **VSWR** - VSWR > 2.0 15
5. - > 20%

1. - PRB
2. - RSSI
3. - VSWR
4. - SLA

Nokia

-

RSSI []/[]/[] RRC [] VSWR [] RMOD []

[] - []

[] RRC [] VSWR RMOD []

[] - [] **VSWR** []

- 0000000000 RAN Monitor 0000
- 0000000000000000

00000000

000

- 00000000 > 5 0
- 000000000

000

1. 00000000 > 80
2. 00000000/00000000
3. InfluxDB 00000
4. 00000

00000

- 00000000
- 00000000024 00 vs. 1 00
- 0 InfluxDB 00000000
- 00 InfluxDB CPU/00
- 00000000

0000000

000

- 000000000
- 000000
- 000000

000

1. 数据源配置
2. 数据源连接
3. 数据源测试
4. 数据源验证

数据源

- 数据源配置 → 数据源
- 数据源连接 → 数据源
- 数据源测试 URL/端口
- 使用 Grafana 数据源
- 数据源验证

数据源配置

数据源

- 数据源配置
- 数据源连接
- 数据源测试

数据源

1. 数据源配置
2. 数据源连接
3. 数据源/数据源
4. 数据源验证

数据源

- 数据源配置 InfluxDB 数据源
- 使用 aggregateWindow 数据源/数据源
- 使用 InfluxDB 数据源
- 数据源验证

□□□□□□

□□□

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

□□□□□

- □□ □□□□□□ □□□□□□□□
- □□□□□□□□□□□□□□□□□□□□
- □□□□□□□□□ eNodeB □□□□□□

□□□□□

□□□

- InfluxDB □□□□□
- □□□□□□ eNodeB □□
- □□□□□□□□□□

□□□□□

- □□ □□□□□□□ □□□□□□□□
- □□ AirScale □□ □□□□
- □□ eNodeB □□□□□

□□□□□

- **Nokia** □□□□□ - □□□□□□□□□□□□
- □□□□□□ - □□□□□□□□□□□□
- □□□□□□□□ - □□□□□□□□
- **AirScale** □□ - □□□□□□□□

項目	値	説明
IP アドレス	10.7.15.67	監視サーバーの IP アドレス
ポート	8080	監視サーバーのポート番号 (Nokia AirScale 8080)
監視対象	Site-B-Tower-1	監視対象のサイト名
ユーザー名	admin	監視サーバーのユーザー名
パスワード	password123	監視サーバーのパスワード

監視対象のリスト

- 監視対象のリスト
- 監視対象のリスト
- 監視対象のリスト
- 監視対象のリスト: NYC-SiteA-BS1, LAX-Tower-Main, CHI-Indoor-DAS

ステップ 3: 監視対象の追加

監視対象の追加手順

1. Web UI: `https://<ran-monitor-ip>:9443`
2. 監視対象の追加: **eNodeBs**
3. 監視対象の IP アドレスと ID
4. 監視対象の ID

監視対象の追加完了 RAN 監視

ステップ 4: 監視対象の追加

監視対象の追加: `config/runtime.exe` 監視対象: `airscales`

```

config :ran_monitor,
  nokia: %{
    ne3s: %{
      # ... ne3s ...
    },
    airscales: [
      # 
      %{
        address: "10.7.15.66",
        name: "Site-A-BS1",
        port: "8080",
        web_username: "admin",
        web_password: "password1"
      },

      # 
      %{
        address: "10.7.15.67",           # IP 
        name: "Site-B-Tower-1",         # 
        port: "8080",                   # 
        web_username: "admin",          # WebLM 
        web_password: "password123"    # WebLM 
      }
    ]
  }
}

```

Elxir -

5

```
elixir -c config/runtime.exs
```

-

- 000000000000 { 0000 [0000
- 0000?0000
- 0000000000

00 6000 RAN 00

0000000000000000

```
# 0000000000
# [ Ctrl+C 00000000
mix phx.server

# 0000000000
systemctl restart ran_monitor

# 0000000000
/path/to/ran_monitor/bin/ran_monitor restart
```

00 7000000000

0000000000000000

1. 0000000000

```
[info] Attempting registration with device: Site-B-Tower-1
[info] Successfully registered with Site-B-Tower-1
```

2. 00 Web UI

- 000 00 00
- 00000000000000
- 0000 "000"00000
- 000000 "000"

3. 0000000000

- 0000

- 設定する
- 設定 "設定" を確認する

4. InfluxDB のインストール

- InfluxDB のインストール
- 設定
- 確認

8. Grafana のインストール

インストール手順

1. 設定
2. 設定
3. 設定 "設定" を確認する
4. 設定

インストール手順 [Grafana](#)

9. Grafana のインストール

Grafana のインストール

1. 設定
2. 設定
3. 設定

インストール手順 [Grafana](#)

インストール

インストール手順

00 1 000000000000

00000000000000000000

00 A 000000

- 000000000000
- 00000000000000000000

00 B 000000

- 000000000000
- 00 InfluxDB 00
- 000 - 000000

00 2 0000000000000000

0000000000000000

00 config/runtime.exe 00 airscales 00000000000000000000

```
airscales: [  
  {%  
    address: "10.7.15.66",  
    name: "Site-A-BS1",  
    port: "8080",  
    web_username: "admin",  
    web_password: "password1"  
  },  
  
  # 0000 - 00000000  
  # {%  
  #   address: "10.7.15.67",  
  #   name: "Site-B-Tower-1",  
  #   port: "8080",  
  #   web_username: "admin",  
  #   web_password: "password123"  
  # }  
]
```


06 Grafana 0000

0 Grafana 00000000

1. 0000000000
2. 00000000000000
3. 0000000000000000

00000000

000000000000 RAN 000000

00 1000000000

0000000000

1. 0000000000000000
2. 0000000000
3. 0000000000000000

00 20000000

00 `config/runtime.exs` 00000000

```
airscales: [  
  %{  
    address: "10.7.15.66",  
    name: "Site-A-BS1",  
    port: "8080",  
    web_username: "admin",  
    web_password: "new_password_here" # 000000  
  }  
]
```

Step 3: RAN

Restart

```
systemctl restart ran_monitor
```

Step 4: Verification

Check

1. Verify

```
[info] Attempting registration with device: Site-A-BS1  
[info] Successfully registered with Site-A-BS1
```

2. Web UI

- Verify "Site-A-BS1"
- Verify
- "Site-A-BS1" status

Check

- Verify
- Verify
- Verify
- Verify

Step 5: Summary

Step RAN configuration

□□□□□□

RAN □□□□□□□□□□□□□□□□

□□□□	□□□□	□□□	□□□□□□
□□□□	10 □	□	□□□□□□□□□□□□□□□□/□□□□
□□	10 □	□	□□□□□□□□□□□□□□
□□	60 □	□	□□□□□□□□□□□□□□
□□□□	30 □	□	□□□□□□□□

□□□□□□

□□□□□□□□□□

- □□□□□□□□□□
- □□□□□□□□□□
- SLA □□□□□□
- □□□□□□□□

□□□□□□□□□□□□

- □□□□□□
- □□ InfluxDB □□□□
- □□□□□□□□□□
- □□□□□□□□

□□ **1**□□□□□□

□□□□□□□□□□□□□□□□ runtime.exe □□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□

- □□□□□ `lib/ran_monitor/nokia/airscale/manager.exe`

- `lib/ran_monitor/nokia/airscale/manager.ex`
- `lib/ran_monitor/nokia/airscale/manager.ex`

OmniTouch

2

- $\text{CPU} \times \text{I/O} \times \text{DB} = \text{Cost}$
- $\text{CPU} = \text{Cost} / \text{I/O} / \text{DB}$
- $\text{I/O} = \text{Cost} / \text{CPU} / \text{DB}$

- $\text{CPU} \times \text{I/O} = \text{Cost}$
- $10 \text{ CPU} = 8,640 \text{ I/O}$
- InfluxDB

- $\text{CPU} = \text{RAN CPU}$
- $\text{I/O} = \text{RAN I/O}$

3

1. $\text{CPU} \times \text{I/O} \times \text{DB} = \text{Cost}$
2. $\text{CPU} = \text{Cost} / \text{I/O} / \text{DB}$
 - RAN CPU
 - I/O
 - InfluxDB I/O
3. $\text{I/O} = \text{Cost} / \text{CPU} / \text{DB}$
 - InfluxDB

- 0000000000
4. 0000000000
- 0000000000
 - 000000000000
-

000000

0000 RAN 000000000000

000000

0000000000000000 Web UI 00 - 000000

000000

1. 00 0000000000
2. 00 0000
3. 00 0000 - 000000 ID
4. 00 0000 ID 0000
5. 00 00000000
6. 00 000000

000000

00000000

- 000000000000
- 0000000000
- 000000000000

00000000

- 000000000000000000
- 00000000000000
- 00000000

部署前準備

- 環境構築
- ネットワーク
- 権限

インストール

- 依存関係
- 実行 ID 確認
- 実行
- 確認

起動

- 起動
- 確認
- 再起動 15-30 分
- 確認

確認

確認 RAN 確認

確認

確認

1. Web UI 確認

```
https://<ran-monitor-ip>:9443
```

2. 確認

- 確認

- 設定する
- 設定する

3. 設定

- 設定する
- 設定する
- 設定する

4. 設定

- InfluxDB 設定
- 設定
- 設定

5. 設定

- 設定
- "設定" 設定
- 設定

設定する [Web UI 設定](#) - [Web UI 設定](#)

Grafana 設定

00000000 S1 00000 LNMME00000000000000 UE000 PRB 000000000000000000
00000000000000000000000000000000

00000000

000000000000

1. 000000

- 00 Grafana 000000000000
- 000000000000
- 0000000000

2. 000000

- InfluxDB 0000000000
- MySQL 000000
- 000000000000

3. 00000000

- 000000000000000000
- 00000000
- 0000000000

4. 00000000

- RAN 00000000 CPU 0000
- 00000000
- 00000000

5. 00000

- 0000000000000000
- 000000000000
- 0000000000

000000 30-45 00

□□□□

□□□□□□

□□□□□□□□ □□□□□□□□

□□□□□

□□□□□□□

- □□□□□□□□□
- □□□□□□□□□□□□

□□□□□□

- □□□□□□□□□□□□
- □□ `config/config.exe` □□□□□□□□

□□□□□□

- □□□□□□□ "□□□□□" □□
- □□□□□□□□□□

□□□□□

- □□□□ InfluxDB □□□□□□
- □□□□□□□□□□
- □□□□□□□□□□□□□□

□□□□

□□□□□□□

1. □□□□□□□□□□□□
2. □□ "□□□□□"
3. □ XML □□□□□□□□□□

4. InfluxDB로 데이터 저장

InfluxDB에 데이터 저장

```
# InfluxDB에 데이터 저장
influx -database 'nokia-monitor' -execute '
SELECT * FROM PerformanceMetrics
WHERE basebandName=''Site-A-BS1''
AND time > now() - 7d
' -format csv > export.csv
```

Grafana로 데이터 시각화

- Grafana 설치
- InfluxDB 데이터 소스 추가
- "PerformanceMetrics" - "Site-A-BS1" - "CSV"

백업 및 복구

백업

복구

1. 백업

```
# 백업 실행
cp config/runtime.exe backups/runtime.exe.$(date +%Y%m%d)

# 백업 파일 압축
tar -czf backups/config-$(date +%Y%m%d).tar.gz config/
```

2. 복구

- Grafana Web UI에서 데이터 소스 설정
- 백업된 파일을 복원
- Grafana에서 데이터 시각화 확인

3. MySQL

```
# MySQL
mysqldump -u ran_monitor_user -p ran_monitor >
backups/ran_monitor-$(date +%Y%m%d).sql
```

4. InfluxDB

```
# InfluxDB 1.x
influxd backup -portable -database nokia-monitor
/backups/influx-$(date +%Y%m%d)

# InfluxDB 2.x
influx backup /backups/influx-$(date +%Y%m%d)
```

5. SSL

```
cp priv/cert/* backups/certificates-$(date +%Y%m%d)/
```

□□□□

□□□

- MySQL
-

□□□

- InfluxDB
-

□□□□

-
-
-

□□□□

□□□□□□□□

1. □□ RAN □□

```
systemctl stop ran_monitor
```

2. □□□□□□

```
cp backups/runtime.exs.20251230 config/runtime.exs
```

3. □□□□

```
elixir -c config/runtime.exs
```

4. □□ RAN □□

```
systemctl start ran_monitor
```

5. □□□□□□□□

□□□□□□□□

1. □□ RAN □□

```
systemctl stop ran_monitor
```

2. □□ MySQL □□□

```
mysql -u ran_monitor_user -p ran_monitor < backups/ran_monitor-20251230.sql
```

3. □□ RAN □□

```
systemctl start ran_monitor
```

4. 配置数据库

5. 配置服务

6. 配置 InfluxDB 数据库

配置数据库

1. 配置 RAN 数据库

2. 配置数据库

3. 配置 MySQL 数据库

4. 配置 InfluxDB 数据库

5. 配置 SSL 证书

6. 配置 RAN 数据库

7. 配置数据库

8. 配置数据库

配置

配置

配置

1. 配置

- 配置数据库
- 配置数据库
- 配置数据库

2. 配置

- 配置数据库
- 配置数据库

- 0000000000

3. 0000

- 0000000000
- 00000000000000
- 0000000000

4. 0000

- 0000000000
- 0000000000
- 00000000000000

0000

1. 000000

- 0000000000
- 000000000000
- 0000000000

2. 0000

- 00000000
- 00000000
- 0000000000 SSL 00

3. 0000

- 0000000000
- 000000000000
- 00000000000000

0000

1. SSL 0000

- 00000000 SSL 00

- 0000000000

2. 0000

- 0000000000
- 00000000
- 00 API 00

3. 0000

- 00 RAN 00000000
- 0000000000
- 00000000

0000

00000000

1. 0000000000

- 0000000000
- 000 Grafana 0000000000

2. 0000000000

- 00000000
- 00000000
- 0000000000

3. 00000000

- 00000000
- 00000000
- 0000000000000000

4. 000000

- 0000000000
- 00000000

- 000000

5. 0000000

- 00000000
- 000000
- 0000000
- 00000000

6. 00000

- 0000000
- 000000000
- 000000000

0000

- 0000 - 000000
- **Web UI** 00 - 00000000
- 0000000 - 0000
- **AirScale** 00 - 0000
- **Grafana** 0000 - 000000
- 000000 - 000000
- 00000000 - 00000000
- 000000 - 0000
- 0000 - 0000000

□□□□□□ (M8xxx)

□□	□□	□□□□□	□□□□□
M8000	S1 □□	33	□□□□□□□□S1 □□/□□□□UE □□□
M8001	□□□□	336	PDCP □□□RACH□□□□□MCS □□
M8004	X2 □□	4	X2 □□□□eNB □□□□□
M8005	□□□□□	237	RSSI□SINR□□□□□□□□AMC
M8006	EPS □□	54	□□□□/□□/□□
M8007	□□□□□□□□	14	DRB □□□□□
M8008	RRC □□□□	14	□□□□□□□□□
M8009	□□□□	8	HO □□□□
M8010	CQI □□	27	□□□□□□□□□
M8011	□□□□□	55	PRB □□□□□□□
M8012	□□□	121	PDCP □□□□□□□□□
M8013	□□□□	21	RRC □□□□□□□/□□
M8014	eNB □□□□□	14	□□ X2 □□□□□
M8015	eNB □□□□	13	□□□□□□
M8016	CS □□	18	□□□□□□□□□
M8017	□□□ HO	10	□□□□□ RAT□3G/2G□
M8018	eNB □□	8	□□ UE □□□□□□□

項目	単位	測定時間	測定項目
M8019	NACC	4	電圧変動率
M8020	電圧	7	電圧変動率
M8021	電圧 HO	17	電圧変動率
M8022	X2 電圧	2	X2 電圧
M8023	電圧	36	PDCP SDU 電圧変動率

電圧変動率 (M5xxx)

項目	単位	測定時間	測定項目
M5112	IP 電圧	112	電圧変動率
M5113	電圧 RX	21	電圧変動率

電圧変動率 (M4xxxx)

項目	単位	測定時間	測定項目
M40001	電圧	電圧	VSWR電圧変動率RF 電圧
M40002	電圧	電圧	電圧

電圧変動率

電圧変動率

- M8013Cxx - 000000

000000

- M8020Cxx - 000000
- M8049Cxx - 0000

000000

- M8005Cxx - 00000000
- M8001Cxx - 0000
- M8013Cxx - 0000

0000000000

M8011 - 000000

000000000000PRB000000000000000000000000

000	00	00	00	0 0	00
M8011C24	00 PRB 0000 TTI 00	0000000000000000 PRB 00	0.1%	0 0	00 10 00 000
M8011C37	00 PRB 0000 TTI 00	0000000000000000 PRB 00	0.1%	0 0	00 10 00 000

00 PRB 0000

- 000000PRB 0 LTE 0000000000000000
- **TTI**0000000000 = LTE 00 1 00
- 000000 = 0000000000000000
- 100% 0000000000000000

频段	制式	速率	带宽	时隙	业务
M8012C23	PDCCP 帧格式 帧格式	帧格式 PDCCP 帧格式	kbit/s	帧 帧	帧 UE 帧 PDCCP SDU 帧
M8012C26	PDCCP 帧格式 帧格式	帧格式 PDCCP 帧格式	kbit/s	帧 帧	帧 UE 帧 PDCCP SDU 帧

帧 PDCCP 帧格式

- PDCCP 帧格式帧格式帧格式
- 帧格式帧格式帧格式kbit/s帧格式帧格式
- 帧格式帧格式帧格式帧格式
- 帧格式帧格式

帧格式

- 帧 1 帧格式帧格式帧格式
- 帧格式帧格式帧格式
- 帧 VoLTE 帧格式帧格式

帧格式

帧格式M8012C26帧格式

- < 10 Mbps - 帧格式/帧格式
- 10-50 Mbps - 帧格式
- 50-100 Mbps - 帧格式/帧格式
- > 100 Mbps - 帧格式帧格式帧格式帧格式

帧格式M8012C23帧格式

- < 5 Mbps - 帧格式
- 5-20 Mbps - 帧格式
- 20-40 Mbps - 帧格式
- > 40 Mbps - 帧格式

UE		
0-50		
50-100		
100-150		
> 150		

-
- AirScale 150-250 UE
- UE

M8020 -

M8020C3					~10
M8020C4					~10
M8020C6					~10

$$\% = 100.0 \times M8020C3 / (M8020C6 - M8020C4)$$

□□□□

- **M8020C3** - □□□□□□□□□□
- **M8020C6** - □□□□□□
- **M8020C4** - □□□□□□□□□□□□□□

□□□□□□

□□□	□□	□□
> 99.9%	□□	□□□□□□ SLA
99.0-99.9%	□□	□□□□
95.0-99.0%	□□	□□□□
< 95.0%	□	□□ - □□□□□□□□□□

Grafana □□□□□

```
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] == "M8020C3" or
    r["metricCounter"] == "M8020C6" or
    r["metricCounter"] == "M8020C4")
  |> pivot(rowKey:["_time"], columnKey: ["metricCounter"],
valueColumn: "_value")
  |> map(fn: (r) => ({
    _time: r._time,
    "□□□□□": 100.0 * r.M8020C3 / (r.M8020C6 - r.M8020C4)
  })))
```


$$\text{Percentage} \% = 100.0 \times \text{M8013C5} / (\text{M8013C17} + \text{M8013C18} + \text{M8013C19} + \text{M8013C34} + \text{M8013C31} + \text{M8013C21} + \text{M8013C93} + \text{M8013C91})$$

□□□□

- **M8013C5** - □□□□□□□ RRC □□□□□□
- □□□□□□□ - □□□□□□□□□□

□□□□

- **MO-S (M8013C17)** - □□□□□□□□□□□□□□
- **MT (M8013C18)** - □□□□□□□/□□□□
- **MO-D (M8013C19)** - □□□□□□□□□□□□
- **MO-V (M8013C91)** - □□□□□□□VoLTE □□□□
- **□□ (M8013C21)** - □□□□□911□112□

□□□□

□□□	□□	□□
> 99%	□□	□□□□
95-99%	□□	□□□□□□
90-95%	□□	□□□□
< 90%	□	□□□□ - □□□□

□□□□□□□

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

QoS

M8001 -

M8001C2	PDCP SDU	eNB		

- **PDCP SDU** =
- =
- **DTCH** =
- =

< 10		VoLTE
10-20		
20-50		
> 50		

-
-
-
-

S1 网络接口

M8000 - S1 网络接口

网络接口 eNodeB 与 MME 网络接口 S1 网络接口 S1 网络接口

接口名称	接口描述	接口地址	接口类型	接口协议
M8000C0	网络接口	网络接口	接口	接口
M8000C1	网络接口	网络接口	接口	接口
M8000C2	网络接口 - 网络接口	网络接口	接口	接口
M8000C3	网络接口 - 网络接口	网络接口	接口	接口
M8000C6	S1 网络接口	S1 网络接口	接口	接口
M8000C7	S1 网络接口	网络接口 S1 网络接口	接口	接口
M8000C11	S1 网络接口	网络接口 MME 网络接口	接口	接口
M8000C12	UE 网络接口 S1 网络接口	网络接口 UE 网络接口 S1 网络接口	接口	接口
M8000C29	网络接口 NAS 网络接口	网络接口 MME 网络接口 NAS 网络接口	接口	接口
M8000C30	网络接口 NAS 网络接口	网络接口 MME 网络接口 NAS 网络接口	接口	接口

网络接口 S1 网络接口

- **S1** 网络接口 eNodeB 与 EPC 网络接口
- 网络接口 网络接口 UE 网络接口
- **S1** 网络接口 eNodeB 与 MME 网络接口
- **NAS** 网络接口 网络接口

网络接口

$$S1 \text{ } = 100 \times M8000C7 / M8000C6$$

$$\text{ } = 100 \times M8000C1 / M8000C0$$

- S1 > 99%
- > 95%

EPS

M8006 - EPS

E-UTRAN E-RAB EPS

M8006C0	EPS			
M8006C1	EPS			
M8006C2-C5				

EPS

- = UE
- =
- = QoS VoLTE

-
-

- 000000000000

00000

M8009 - 000000

M8014 - eNB 0000000

M8015 - eNB 000000

M8021 - 0000000

00000000000—UE 000000000000000000000000

00000

00	00	000
00 eNB	00000000000	M8015
eNB 00	00000000000X20	M8014
000	00000000000	M8021
000	000000000LTE→3G/2G0	M8017

00000

Scenario	Configuration	Performance Metric
M8009	Scenario	Performance Metric
M8014	Scenario X2 Scenario HO	Performance Metric
M8015	Scenario HO	Performance Metric
M8021	Scenario HO	Performance Metric

Performance Metric

$$\text{Performance Metric} = 100 \times (\text{Scenario}) / (\text{Performance Metric})$$

Performance Metric

- Scenario eNB HO Performance Metric > 99%
- eNB Scenario HO Performance Metric > 98%
- Scenario HO Performance Metric > 95%

Performance Metric

- Performance Metric
- Performance Metric
- Performance Metric UE Performance Metric
- Performance Metric

Performance Metric

M8010 - CQI Performance Metric

Performance Metric UE CQI Performance Metric

Scenario CQI

- **CQI** = UE ↔ eNodeB 通信品質の指標
- 範囲 CQI 0からCQI 15まで
- 範囲 対応する MCS 値を決定
- 範囲 範囲 通信品質の指標

CQI 範囲

CQI 範囲	範囲	範囲	範囲
0-3	範囲	< 1 Mbps	QPSK
4-6	範囲	1-5 Mbps	QPSK
7-9	範囲	5-15 Mbps	16-QAM
10-12	範囲	15-40 Mbps	64-QAM
13-15	範囲	40-150 Mbps	64-QAM

M8010 範囲

- M8010C0 - M8010C15 範囲 0-15 範囲 CQI 範囲

範囲

$$\text{範囲 CQI} = \frac{\sum(\text{CQI_level} \times \text{M8010C}[\text{level}])}{\sum(\text{M8010C}[\text{level}])}$$

範囲 CQI 範囲

- 範囲 CQI 10-15 範囲 通信品質の指標
 - 範囲 CQI 7-9 範囲 通信品質の指標
 - 範囲 CQI 0-6 範囲 通信品質の指標
-

CS 測試

M8016 - CS 測試

測試目的：驗證 CSFB 功能，LTE UE 在 2G/3G 網絡

CSFB 測試

- 測試 VoLTE 呼叫過程
- 測試 UE 在 LTE → 2G/3G → LTE
- 測試 LTE 網絡

測試環境

測試項目	測試方法	測試結果
M8016C0	CSFB 測試	CSFB 測試
M8016C1	CSFB 測試	CSFB 測試

CSFB 測試

1. 測試 UE 在 LTE 網絡 2G/3G 網絡
2. 測試 UE 在 LTE 網絡 2G/3G 網絡

測試結果

- CSFB 測試 = 測試 / 測試 CSFB 測試
- 測試 > 98%

測試

M8023 - PDCP SDU 測試

測試目的：驗證 PDCP SDU 測試

□□□□

- □□□□□□□□□□
- □ QCI□□□□□□□□□□□□□□□□
- □□□□□□□□□□

□□□

- □□□□□□
- □□□□□□□□□□□□
- □□□□□□□□□□
- QoS □□□□□□

□ M8012□□□□□□□□□□

- M8012□ □□□□□□□□kbit/s□
- M8023□ □□□□□□□□□□

X2 □□□□□□

M8004 - X2 □□□□□□

M8022 - X2 □□□□□□

□□□□□□□□ eNodeB □□□ X2 □□□□□□ eNB □□□□□□□□□□□□□□

M8004 - X2 □□□□□□

- □□□□□□□□□□ eNodeB □□□□□□□□□□
- □□□□□ X2 □□□□□□□□□□

M8022 - X2 □□□□□□

- M8022C0□ X2 □□□□□□□□□□
- M8022C1□ X2 □□□□□□□□□□

X2

- eNodeB
-
-

X2

$$X2 = 100 \times \frac{M8022C1}{M8022C0}$$

> 95%

M8007 - DRB

DRB

- DRB
- DRB
-

M8008 - RRC

RRC

- PRB
- UE
-
-

M8019 - 0000000000NACC

00 NACC 00000000 LTE 0 GSM 00000000

000 00000000 UE 00 GSM 0000000000000000

00000000

M5112 - IP 00000000

M5113 - 0000 RX 00

00000000000000000000

M5112 - IP 000000112 00000000

- 00000000
- 00000000
- 00000000
- 000000

M5113 - 0000 RX021 00000000

- 000000000000
- 000000
- 000000CRC 00000000

0000

- 00000000
 - 0000000000
 - 0000000000
 - 000000000000
-

□□□□□□□□□□

M40001 - □□□□□□

□□□	□□	□□	□□	□□	□□
M40001C0	□□□□□□ VSWR	□□□□□	0.1	□□	□□ 10

□□ VSWR□

- **VSWR** = □□□□□
- □□□□□□□□
- □□□□□□□□□□□□□□/□□□□
- □□□□ = □□

VSWR □□□

VSWR	□□	□□
1.0-1.5	□□	□□□□
1.5-2.0	□□	□□□
2.0-3.0	□□	□□
> 3.0	□	□□/□□□□ - □□□□

□ VSWR □□□□□□

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

Grafana □□□□□

```

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] == "M40001C0")
  |> filter(fn: (r) => r._field == "counterValue")
  |> map(fn: (r) => ({ r with "VSWR": r._value / 10.0}))

```

M40002 - 〇〇〇〇

〇〇〇	〇〇	〇〇	〇〇	〇〇	〇〇
M40002C2	〇〇	〇〇〇〇	100000 〇〇	〇〇	〇〇 100000

〇〇〇〇〇

- 〇〇〇〇〇〇〇〇〇
- 〇 OPEX 〇〇〇〇〇〇〇〇〇〇
- 〇〇〇〇〇〇/〇〇〇〇〇〇〇〇〇〇〇

Grafana 〇〇〇〇〇

```

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] == "M40002C2")
  |> filter(fn: (r) => r._field == "counterValue")
  |> map(fn: (r) => ({ r with "Power": r._value / 100000.0}))

```

〇 Grafana 〇〇〇〇〇〇〇

〇〇〇〇〇〇〇〇〇

1. 〇〇〇〇〇〇〇〇

〇〇 M8011C24 〇 M8011C37 〇〇〇〇/〇〇 PRB 〇〇〇

```
// PRB
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] == "M8011C24")
  |> map(fn: (r) => ({ r with _value: r._value / 10.0}))
  |> rename(columns: {"_value": "PRB %"})

// PRB
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] == "M8011C37")
  |> map(fn: (r) => ({ r with _value: r._value / 10.0}))
  |> rename(columns: {"_value": "PRB %"})
```

2. PRB

PRB

```
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] == "M8012C23" or
r["metricCounter"] == "M8012C26")
  |> map(fn: (r) => ({ r with _value: r._value / 1000.0})) // PRB
Mbps
  |> pivot(rowKey: ["_time"], columnKey: ["metricCounter"],
valueColumn: "_value")
  |> map(fn: (r) => ({
    _time: r._time,
    "PRB Mbps": r.M8012C23,
    "PRB Mbps": r.M8012C26
  })))
```

3. PRB

PRB

```

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] =~ /M8020C(3|4|6)/)
  |> pivot(rowKey: ["_time"], columnKey: ["metricCounter"],
valueColumn: "_value")
  |> map(fn: (r) => ({
    _time: r._time,
    "RSSI %": 100.0 * r.M8020C3 / (r.M8020C6 - r.M8020C4)
  })))

```

4. 4G LTE

4G RRC 4G

```

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] =~
/M8013C(5|17|18|19|21|31|34|91|93)/)
  |> pivot(rowKey: ["_time"], columnKey: ["metricCounter"],
valueColumn: "_value")
  |> map(fn: (r) => ({
    _time: r._time,
    "4G RRC %": 100.0 * r.M8013C5 / (r.M8013C17 + r.M8013C18 +
r.M8013C19 + r.M8013C34 + r.M8013C31 + r.M8013C21 + r.M8013C93 +
r.M8013C91)
  })))

```

4G RRC 4G

4G RRC 4G

- 4G RRC - 4G RRC PRB
- 4G - 4G RRC
- 4G RRC - 4G RRC UE
- 4G - 4G RRC RSSI

4G RRC 4G

- 4G RRC

- 网络性能指标
- 网络质量指标

网络性能

- PRB 利用率 < 70% 70-85% > 85%
- 吞吐量 > 99% 95-99% < 95%
- 丢包率 > 99% 95-99% < 95%

网络质量

- 网络延迟
- 网络抖动
- 网络丢包

网络

- 网络 - RAN 网络
 - Grafana 网络 - Grafana 网络
 - 网络 - RAN 网络
 - AirScale 网络 - 网络
 - 网络 - InfluxDB 网络
-

□□□□□

□□□□□□□

□□□	□□	□□	□□
M8011C24	□□ PRB □□□□	□□□□	0.1%
M8011C37	□□ PRB □□□□	□□□□	0.1%
M8012C23	PDCP □□□□□	□□□□	kbit/s
M8012C26	PDCP □□□□□	□□□□	kbit/s
M8018C1	□□□□ UE	□□□□	□□
M8020C3	□□□□□□	□□□□□	□□
M8020C6	□□□□□	□□□□□	□□
M8013C5	□□□□	□□□□□	□□
M8005C0	RSSI □□□	□□□□	dBm
M8005C2	RSSI □□	□□□□	dBm
M8001C2	PDCP □□□□	□□□□	□□
M40001C0	VSWR	□□□□	0.1 □□

□□□

- □□□ FlexiRadio LTE □□□□
- □□□ AirScale □□□□□□□
- 3GPP LTE □□□□□□

□□□□□□□□

□□□□□□ KPI □□□□□□□□ 1,186 □□□□ LTE □□□□□□□□□□□□

=====
LTE
=====

M8000

M8000C0 | | | |
M8000C1 | | | |
M8000C2 | | | |
M8000C3 | | | |
M8000C4 | | | |
M8000C5 | | | |
M8000C6 | S1 | | |
M8000C7 | S1 | | |
M8000C8 | S1 | | |
M8000C9 | S1 MME | | |
M8000C11 | S1 | | |
M8000C12 | UE S1- | | |
M8000C13 | eNB S1 | | |
M8000C14 | MME S1 | | |
M8000C15 | eNB S1 | | |
M8000C16 | MME S1 | | |
M8000C23 | UE | | |
M8000C24 | UE | | |
M8000C25 | UE | | |
M8000C29 | NAS | | |
M8000C30 | NAS | | |
M8000C31 | UE | | |
M8000C32 | IMS E-RAB | | |
M8000C33 | IMS E-RAB | | |
M8000C34 | IMS E-RAB | | |
M8000C35 | | | |
M8000C36 | | | |
M8000C37 | X2 IP | | |
M8000C38 | X2 IP | | |
M8000C39 | WRITE-REPLACE | | |
M8000C40 | WRITE-REPLACE | | |
M8000C41 | KILL | | |
M8000C42 | KILL | | |

M8001

M8001C0	PDCP SDU [][] DL DTCH [][]		
M8001C1	PDCP SDU [][] DL DTCH [][]		
M8001C2	PDCP SDU [][] DL DTCH [][]		
M8001C3	PDCP SDU [][] UL DTCH [][]		
M8001C4	PDCP SDU [][] UL DTCH [][]		
M8001C5	PDCP SDU [][] UL DTCH [][]		
M8001C6	[][] RACH [][][][]	[][]	
M8001C7	[][] RACH [][][][]	[][]	
M8001C8	RACH [][][][]		[]
M8001C9	[] PCH [][][][] TB []		
M8001C10	[] BCH [][][][] TB []		
M8001C11	[] DL-SCH [][][][] TB []		
M8001C12	DL-SCH [][] HARQ [][]		
M8001C13	[][][][][] UL-SCH TB		
M8001C14	[][][] UL-SCH TB [][][][]		
M8001C15	[][][] UL-SCH TB [][]		
M8001C16	[][] MCS0 [] PUSCH [][]		
M8001C17	[][] MCS1 [] PUSCH [][]		
M8001C18	[][] MCS2 [] PUSCH [][]		
M8001C19	[][] MCS3 [] PUSCH [][]		
M8001C20	[][] MCS4 [] PUSCH [][]		
M8001C21	[][] MCS5 [] PUSCH [][]		
M8001C22	[][] MCS6 [] PUSCH [][]		
M8001C23	[][] MCS7 [] PUSCH [][]		
M8001C24	[][] MCS8 [] PUSCH [][]		
M8001C25	[][] MCS9 [] PUSCH [][]		
M8001C26	[][] MCS10 [] PUSCH [][]		
M8001C27	[][] MCS11 [] PUSCH [][]		
M8001C28	[][] MCS12 [] PUSCH [][]		
M8001C29	[][] MCS13 [] PUSCH [][]		
M8001C30	[][] MCS14 [] PUSCH [][]		
M8001C31	[][] MCS15 [] PUSCH [][]		
M8001C32	[][] MCS16 [] PUSCH [][]		
M8001C33	[][] MCS17 [] PUSCH [][]		
M8001C34	[][] MCS18 [] PUSCH [][]		
M8001C35	[][] MCS19 [] PUSCH [][]		
M8001C36	[][] MCS20 [] PUSCH [][]		
M8001C37	[][] MCS21 [] PUSCH [][]		
M8001C38	[][] MCS22 [] PUSCH [][]		
M8001C39	[][] MCS23 [] PUSCH [][]		
M8001C40	[][] MCS24 [] PUSCH [][]		
M8001C41	[][] MCS25 [] PUSCH [][]		
M8001C42	[][] MCS26 [] PUSCH [][]		

M8001C43		□□	MCS27	□	PUSCH	□□
M8001C44		□□	MCS28	□	PUSCH	□□
M8001C45		□□	MCS29	□	PUSCH	□□
M8001C46		□□	MCS30	□	PUSCH	□□
M8001C47		□□	MCS31	□	PUSCH	□□
M8001C48		□□	MCS32	□	PUSCH	□□
M8001C49		□□	MCS33	□	PUSCH	□□
M8001C50		□□	MCS34	□	PUSCH	□□
M8001C51		□□	MCS35	□	PUSCH	□□
M8001C52		□□	MCS36	□	PUSCH	□□
M8001C53		□□	MCS37	□	PUSCH	□□
M8001C54		□□	MCS38	□	PUSCH	□□
M8001C55		□□	MCS39	□	PUSCH	□□
M8001C56		□□	MCS40	□	PUSCH	□□
M8001C57		□□	MCS41	□	PUSCH	□□
M8001C58		□□	MCS42	□	PUSCH	□□
M8001C59		□□	MCS43	□	PUSCH	□□
M8001C60		□□	MCS44	□	PUSCH	□□
M8001C61		□□	MCS45	□	PUSCH	□□
M8001C62		□□	MCS46	□	PUSCH	□□
M8001C63		□□	MCS47	□	PUSCH	□□
M8001C64		□□	MCS48	□	PUSCH	□□
M8001C65		□□	MCS49	□	PUSCH	□□
M8001C66		□□	MCS50	□	PUSCH	□□
M8001C67		□□	MCS51	□	PUSCH	□□
M8001C68		□□	MCS52	□	PUSCH	□□
M8001C69		□□	MCS53	□	PUSCH	□□
M8001C70		□□	MCS54	□	PUSCH	□□
M8001C71		□□	MCS55	□	PUSCH	□□
M8001C72		□□	MCS56	□	PUSCH	□□
M8001C73		□□	MCS57	□	PUSCH	□□
M8001C74		□□	MCS58	□	PUSCH	□□
M8001C75		□□	MCS59	□	PUSCH	□□
M8001C76		□□	MCS60	□	PUSCH	□□
M8001C77		□□	MCS61	□	PUSCH	□□
M8001C78		□□	MCS62	□	PUSCH	□□
M8001C79		□□	MCS63	□	PUSCH	□□
M8001C80		□□	MCS64	□	PUSCH	□□
M8001C81		□□	MCS65	□	PUSCH	□□
M8001C82		□□	MCS66	□	PUSCH	□□
M8001C83		□□	MCS67	□	PUSCH	□□
M8001C84		□□	MCS68	□	PUSCH	□□
M8001C85		□□	MCS69	□	PUSCH	□□
M8001C86		□□	MCS70	□	PUSCH	□□

M8001C87		□□	MCS71	□	PUSCH	□□
M8001C88		□□	MCS72	□	PUSCH	□□
M8001C89		□□	MCS73	□	PUSCH	□□
M8001C90		□□	MCS74	□	PUSCH	□□
M8001C91		□□	MCS75	□	PUSCH	□□
M8001C92		□□	MCS76	□	PUSCH	□□
M8001C93		□□	MCS77	□□□	PUSCH	□□
M8001C94		□□	MCS78	□	PUSCH	□□
M8001C95		□□	MCS79	□	PUSCH	□□
M8001C96		□□	MCS80	□	PUSCH	□□
M8001C97		□□	MCS81	□	PUSCH	□□
M8001C98		□□	MCS82	□	PUSCH	□□
M8001C99		□□	MCS83	□	PUSCH	□□
M8001C100		□□	MCS84	□	PUSCH	□□
M8001C101		□□	MCS85	□	PUSCH	□□
M8001C102		□□	MCS86	□	PUSCH	□□
M8001C103		□□	MCS87	□	PUSCH	□□
M8001C104		□□	MCS88	□	PUSCH	□□
M8001C105		□□	MCS89	□	PUSCH	□□
M8001C106		□□	MCS90	□	PUSCH	□□
M8001C107		□□	MCS91	□	PUSCH	□□
M8001C108		□□	MCS92	□	PUSCH	□□
M8001C109		□□	MCS93	□	PUSCH	□□
M8001C110		□□	MCS94	□	PUSCH	□□
M8001C111		□□	MCS95	□	PUSCH	□□
M8001C112		□□	MCS96	□	PUSCH	□□
M8001C113		□□	MCS97	□	PUSCH	□□
M8001C114		□□	MCS98	□	PUSCH	□□
M8001C115		□□	MCS99	□	PUSCH	□□
M8001C116		□□	MCS100	□	PUSCH	□□
M8001C117		□□	MCS101	□	PUSCH	□□
M8001C118		□□	MCS102	□	PUSCH	□□
M8001C119		□□	MCS103	□	PUSCH	□□
M8001C120		□□	MCS104	□	PUSCH	□□
M8001C121		□□	MCS105	□	PUSCH	□□
M8001C122		□□	MCS106	□	PUSCH	□□
M8001C123		□□	MCS107	□	PUSCH	□□
M8001C124		□□	MCS108	□	PUSCH	□□
M8001C125		□□	MCS109	□	PUSCH	□□
M8001C126		□□	MCS110	□	PUSCH	□□
M8001C127		□□	MCS111	□	PUSCH	□□
M8001C128		□□	MCS112	□	PUSCH	□□
M8001C129		□□	MCS113	□	PUSCH	□□
M8001C130		□□	MCS114	□	PUSCH	□□

M8001C131		□□ MCS115 □ PUSCH □□				
M8001C132		DL DTCH □□ RLC SDUs				
M8001C133		DL DCCH □□ RLC SDUs				
M8001C135		UL DTCH □□ RLC SDUs				
M8001C136		UL DCCH □□ RLC SDUs				
M8001C137		RLC PDU □□□□			□□	
M8001C138		RLC PDU □□			□□	
M8001C139		RLC PDU □□				
M8001C140		DL RLC C-PDUs □□□□□□			□□	
M8001C141		DL RLC □□ PDUs □□□□□□				
M8001C142		UL RLC PDUs □□				
M8001C143		□□□ UL RLC PDU □□				
M8001C144		UL RLC PDU □□□□			□□	
M8001C145		UL RLC PDUs □□				
M8001C146		DL RLC SDU □□ PDCP				
M8001C147		□□□□□ UE □□		□□		□□
M8001C148		□□□□□ UE □□		□□		
M8001C150		□□□□□ UE □□		□□		□□
M8001C151		□□□□□ UE □□		□□		
M8001C153		□□ PDCP SDUs				
M8001C154		□□ PDCP SDUs				
M8001C155		□□□□□ PDCP SDUs			□□	
M8001C156		□□□ PDSCH MCS0 □□			□	
M8001C157		□□□ PDSCH MCS1 □□			□	
M8001C158		□□□ PDSCH MCS2 □□			□	
M8001C159		□□□ PDSCH MCS3 □□			□	
M8001C160		□□□ PDSCH MCS4 □□			□	
M8001C161		□□□ PDSCH MCS5 □□			□	
M8001C162		□□□ PDSCH MCS6 □□			□	
M8001C163		□□□ PDSCH MCS7 □□			□	
M8001C164		□□□ PDSCH MCS8 □□			□	
M8001C165		□□□ PDSCH MCS9 □□			□	
M8001C166		□□□ PDSCH MCS10 □□			□	
M8001C167		□□□ PDSCH MCS11 □□			□	
M8001C168		□□□ PDSCH MCS12 □□			□	
M8001C169		□□□ PDSCH MCS13 □□			□	
M8001C170		□□□ PDSCH MCS14 □□			□	
M8001C171		□□□ PDSCH MCS15 □□			□	
M8001C172		□□□ PDSCH MCS16 □□			□	
M8001C173		□□□ PDSCH MCS17 □□			□	
M8001C174		□□□ PDSCH MCS18 □□			□	
M8001C175		□□□ PDSCH MCS19 □□			□	
M8001C176		□□□ PDSCH MCS20 □□			□	
M8001C177		□□□ PDSCH MCS21 □□			□	

M8001C178		□□□	PDSCH	MCS22	□□		□
M8001C179		□□□	PDSCH	MCS23	□□		□
M8001C180		□□□	PDSCH	MCS24	□□		□
M8001C181		□□□	PDSCH	MCS25	□□		□
M8001C182		□□□	PDSCH	MCS26	□□		□
M8001C183		□□□	PDSCH	MCS27	□□		□
M8001C184		□□□	PDSCH	MCS28	□□		□
M8001C185		□□□	PDSCH	MCS29	□□		□
M8001C186		□□□	PDSCH	MCS30	□□		□
M8001C187		□□□	PDSCH	MCS31	□□		□
M8001C188		□□□	PDSCH	MCS32	□□		□
M8001C189		□□□	PDSCH	MCS33	□□		□
M8001C190		□□□	PDSCH	MCS34	□□		□
M8001C191		□□□	PDSCH	MCS35	□□		□
M8001C192		□□□	PDSCH	MCS36	□□		□
M8001C193		□□□	PDSCH	MCS			□

PM 環境構築

概要

PM 環境構築には InfluxDB と PM 環境構築 AirScale をインストールし、**22,000** の PM 環境構築環境を構築します。

環境構築環境を構築します。

環境構築

PM 環境構築

- 環境構築 `https://localhost:9443`
- 環境構築 環境構築
- 環境構築 PM 環境構築

環境構築

環境構築環境構築

環境構築	環境構築
環境構築 PM 環境構築	環境構築環境構築 InfluxDB 環境構築
環境構築環境構築	環境構築 22,000+ 環境構築環境構築

環境構築

PM 環境構築環境構築環境構築

種別	識別子	数	説明
LTE	M8xxx	~5,900	LTE L1/L2/L3 ERAB/RRC
WCDMA	M5xxx	~885	3G WCDMA MAC CQI/HSDPA
5G-NR	M55xxx	~14,500	5G NR MIMO
5G-UE	M51xxx	~500	5G UE
5G-UE	M40xxx	~250	5G UE/UE

目録

目録 `priv/pm_counters.csv` の項目

- UE
- UE
- UE
- RRC UE/UE
- **PRB**
- UE/UE
- **RRC** RRC
- **ERAB** E-RAB
- **PDCP** PDCP
- UE
- UE

PM Counter

Overview

1. PM Counter is a counter that counts the number of packets sent and received by a device.
2. It is used to monitor the network performance and to identify the devices that are causing network congestion.
3. It is used to monitor the network performance and to identify the devices that are causing network congestion.
4. It is used to monitor the network performance and to identify the devices that are causing network congestion.

Configuration

1. PM Counter is configured in the PM Counter configuration page.
2. It is used to monitor the network performance and to identify the devices that are causing network congestion.

Monitoring

Monitoring PM Counter

- PM Counter ID is used to identify the device.
- PM Counter is used to monitor the network performance and to identify the devices that are causing network congestion.

Access

PM Counter is accessed via `priv/pm_counters.csv` endpoint.

PM Filter

PM Filter

1. PM Filter is configured in the PM Filter configuration page.
2. It is used to filter the network traffic based on the PM Counter ID.
3. It is used to filter the network traffic based on the PM Counter ID.

InfluxDB is used to store the PM Counter data.

PM Counter

PM Counter	Technology	Description
M8012C23	LTE	PM Counter
M8012C26	LTE	PM Counter
M8001C2	LTE	PDCP SDU DL
M8011C24	LTE	PRB
M8011C37	LTE	PRB
M8013C17	LTE	RRC
M8020C3	LTE	
M40001C0	5G	

PM Counter

pm_counters.csv

PM Counter

```
# PM Counter,Technology,Description
M8012C23,LTE,PM Counter
M8012C26,LTE,PM Counter
M8001C2,LTE,PDCP SDU DL
...
```

PM Counter `priv/pm_counters.csv`

pm_metrics.csv

□□□□□□□□□□□□

```
# □□□PM_Code,□□,□□  
M8000C6,LTE,S1_SETUP_ATT  
M8000C7,LTE,S1_SETUP_SUCC  
...
```

□□□ `priv/pm_metrics.csv`

□□□□

□□□□□□□

1. □□□□□□□□“□□□ PM □□”□□□□
2. □□ eNodeB □□□□□□ PM □□□□□□ InfluxDB □□□□□
3. □□□□□ ID □□□□□□□□□□□□□□

□□□□□□

1. □□□□□□□□□□□□□□□□
2. □□□□□ **eNodeB** □□□□ **PM** □□□□□□□□□□ 15 □□□□□□
3. □□□□□□□□□□□ □□ `[PmFilterStore]` □□
4. □□□□□□□□□□□□□□□□

□□□□□□□

1. □□□□□□□□□□□□□□□□ `priv/pm_metrics.csv`
 2. □□□□□□□□□□□□□□□□
 3. □□ UTF-8 □□□□□
-

□□□□

- □□□□□ - PM □□□□□□□□
 - Grafana □□ - □□ PM □□□□□□□□
 - InfluxDB □□ - □□ PM □□
-

□□□

- **PM** □□□□ <https://localhost:9443/nokia/pm-filters>
- □□□□ <https://localhost:9443/nokia/retention>
- **InfluxDB** □□□ <https://localhost:9443/nokia/influx>

- `password` (string)

password (string)

- `password`
- `password` "omnitouch2024"
- `password` `password`
- `password` `password`

hostname (string)

- `hostname`
- `hostname` "localhost"
- `hostname`
 - "localhost" - `hostname` `?``?``?`
 - "127.0.0.1" - TCP `hostname`
 - "10.179.2.135" - `hostname` IP
 - "db.example.com" - `hostname`

database (string)

- `database`
- `database` "ran_monitor"
- `database` `database` RAN `database`
- `database` `CREATE DATABASE ran_monitor CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;`

stacktrace (boolean)

- `stacktrace`
- `stacktrace` true
- `stacktrace` true - `stacktrace`
- `stacktrace` false - `stacktrace`

show_sensitive_data_on_connection_error (boolean)

- `show_sensitive_data_on_connection_error`

- `enable` `true`
- `enable` `true` - `enable` `true`
- `enable` `false` - `enable` `false`

pool_size (int)

- `enable` `true`
- `enable` `10`
- `enable`
 - 1-5 `pool_size: 5`
 - 6-20 `pool_size: 10`
 - 21-50 `pool_size: 15`
 - 50+ `pool_size: 20`
- `enable` `2` `enable` + 5 `enable` Web UI

Web UI

RAN `enable` Web `enable`

`enable` SOAP/API `enable`

```
config :ran_monitor, RanMonitor.Web.Endpoint,
  http: [ip: {0, 0, 0, 0}, port: 8080],
  check_origin: false,
  secret_key_base:
    "v5t0S1/QRonjw0ky7adGGfkBbrJmiJyXhpesJy/jvSZhqLZkREV+rlo1/pR8lkbu",
  server: true
```

`enable` `enable` NE3S `enable` SOAP `enable`

ip (int)

- `enable`
- `enable` `{0, 0, 0, 0}` `enable`
- `enable`

- `{0, 0, 0, 0}` - 本地 IP 地址
- `{127, 0, 0, 1}` - 本地 IP 地址
- `{10, 179, 2, 135}` - 本地 IP 地址

port (端口)

- TCP 端口
- 默认值 `8080`
- 端口号必须在 1024 到 65535 之间
- 端口号必须是 IP 地址

check_origin (检查)

- 检查 WebSocket/HTTP 请求
- 默认值 `false`
- 检查是否来自 SOAP API 请求 `false` 检查是否来自 Web UI

secret_key_base (密钥)

- 密钥长度
- 默认值 64 字节
- 默认值 `mix phx.gen.secret`
- 密钥必须在 1024 到 65535 之间
- 密钥必须是 IP 地址

server (服务器)

- 服务器地址
- 默认值 `true`
- 检查是否来自 runtime.exs 请求 `true`

Web UI

```
# HTTPS port 9443
https_port =
String.to_integer(System.get_env("CONTROL_PANEL_HTTPS_PORT") ||
"9443")

config :control_panel, ControlPanelWeb.Endpoint,
  url: [host: "0.0.0.0", port: https_port, scheme: "https"],
  https: [
    ip: {0, 0, 0, 0},
    port: https_port,
    keyfile: "priv/cert/omnitouch.pem",
    certfile: "priv/cert/omnitouch.crt"
  ]
```

Web UI HTTPS

- **CONTROL_PANEL_HTTPS_PORT** - HTTPS 9443
 - HTTPS
 - `export CONTROL_PANEL_HTTPS_PORT=8443`

url (Endpoint)

- URL
- **host:** "0.0.0.0" -
- **port:** `https_port` `CONTROL_PANEL_HTTPS_PORT`
- **scheme:** "https" - HTTPS

https (SSL)

- HTTPS
- **ip:** {0, 0, 0, 0} -
- **port:** `https_port` url
- **keyfile:** SSL
- **certfile:** SSL

SSL 証明書

- 証明書 SSL/TLS 用
- 証明書生成
- 証明書 CA 証明書
- 証明書

```
openssl req -newkey rsa:2048 -nodes -keyout omnitouch.pem -x509  
-days 365 -out omnitouch.crt
```

AirScale Webhook 設定

```
config :ran_monitor, RanMonitor.Web.Nokia.Airscale.Endpoint,  
  url: [host: "0.0.0.0"],  
  http: [ip: {0, 0, 0, 0}, port: 9076],  
  server: true
```

設定 AirScale 証明書

port (設定)

- 9076
- PMCADM 証明書 rTpmCollEntityPortNum
- WebLM 証明書

ログ

```
config :logger,  
  level: :info  
  
config :logger, :console,  
  format: "$time $metadata[$level] $message\n",  
  metadata: [:request_id]
```

□□□□

level (□□)

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

□□□□□□□□

- □□□ :debug - □□□□□□□□□□
- □□□ :info - □□□□□□□□□□□□□□
- □□□□□ □□□□□ :debug □□□□□□
- □□□□□□ :warning - □□□□□□□□□□□□

□□□□□□

format (□□□)

- □□□□□□□□□□
- □□□□ "\$time \$metadata[\$level] \$message\n"
- □□□
 - \$time - □□□□
 - \$metadata - □□□□□□
 - \$level - □□□□□□info□error □□
 - \$message - □□□□□□□□

metadata (□□□□)

- □□□□□□□□□□
- □□□□ [:request_id]
- **request_id:** □□□□□□□□□□ HTTP □□

□□□□□

□□□□ RAN □□□□□□□□□□□□□□

```
config :ran_monitor,
  general: %{
    mcc: "505",
    mnc: "57"
  },
  nokia: %{
    ne3s: %{
      webhook_url: "http://10.5.198.200:9076/webhook",
      private_key: Path.join(Application.app_dir(:ran_monitor,
"priv"), "external/nokia/ne.key.pem"),
      public_key: Path.join(Application.app_dir(:ran_monitor,
"priv"), "external/nokia/ne.cert.der"),
      reregister_interval: 30
    },
    airscales: [
      %{
        address: "10.7.15.67",
        name: "ONS-Lab-Airscale",
        port: "8080",
        web_username: "Nemuadmin",
        web_password: "nemuuser"
      }
    ]
  }
}
```

□□□□

mcc (□□□)

- □□□□□
- □□□□ "505"
- □□□ □□ 3GPP □□□□□
- □□□ 3 □□□

- ITU-T E.212

mnc ()

-
- "57"
-
- 2 3

NE3S ? ? ? ? ? NE3S

webhook_url ()

- URL
- "http://10.5.198.200:9076/webhook"
- http://<ran-monitor-ip>:<port>/webhook
- IP RAN IP
- RanMonitor.Web.Nokia.Airscale.Endpoint 9076
- /webhook

private_key (-)

-
- priv/external/nokia/ne.key.pem
- PEM
-
- OpenSSL

public_key (-)

-
- priv/external/nokia/ne.cert.der
- DER
-
- private_key

reregister_interval ()

- 30
- 30-300
- 30

AirScale

airscales (air)

- AirScale
-

address (ip)

- IP
- "10.7.15.66"
- IPv4
- RAN
- ping 10.7.15.66

name (name)

-
- "ONS-Lab-Airscale"
- Web UI InfluxDB
-
- - "NYC-Site-A-BS1"
 - "LAX-Tower-Main"
 - "TestLab-Airscale-01"

port (port)

-

- 8080
- AirScale 8080
-
-

web_username ()

- WebLM
- "Nemuadmin"
- API
- /
-

web_password ()

- WebLM
- "nemuuser"
-
-

airscapes

```
airscales: [  
  {%  
    address: "10.7.15.66",  
    name: "ONS-Lab-Airscale",  
    port: "8080",  
    web_username: "Nemuadmin",  
    web_password: "nemuuser"  
  },  
  {%  
    address: "10.7.15.67",  
    name: "Site-A-Tower-1",  
    port: "8080",  
    web_username: "admin",  
    web_password: "password123"  
  },  
  {%  
    address: "192.168.100.50",  
    name: "Site-B-Indoor",  
    port: "8080",  
    web_username: "admin",  
    web_password: "different_password"  
  }  
]
```

InfluxDB

```
config :ran_monitor, RanMonitor.InfluxDbConnection,  
  auth: [  
    username: "monitor",  
    password: "sideunderTexasgalaxyview_61"  
  ],  
  database: "nokia-monitor",  
  host: "10.179.2.135"
```

  InfluxDB               

□□□□

auth (□□□□□)

- InfluxDB □□□□□□□□
- **username:** InfluxDB □□□□□ "monitor" □
- **password:** InfluxDB □□□ "sideunderTexasgalaxyview_61" □
- □□□ □□ InfluxDB 2.x□□□□□□ API □□

database (□□□□)

- InfluxDB □□□/□□□□□□□□
- □□□□ "nokia-monitor"
- **InfluxDB 1.x** □ □□□□□□□
- **InfluxDB 2.x** □ □□□
- □□□ □□□□□ RAN □□□□□□□□

```
# InfluxDB 1.x
influx -execute 'CREATE DATABASE "nokia-monitor"'

# InfluxDB 2.x
influx bucket create -n nokia-monitor -o your-org
```

host (□□□□)

- InfluxDB □□□□□□
- □□□□ "10.179.2.135"
- □□□ IP □□□□□□□□
- □□□ □□ InfluxDB □□□8086□□□□
- □□□
 - "localhost" - □ RAN □□□□□□□□□□
 - "10.179.2.135" - □□ InfluxDB □□□□
 - "influxdb.example.com" - □□□□

InfluxDB □□□□□□□□

□□□□□□

- RAN 8086 InfluxDB
- `curl http://10.179.2.135:8086/ping`

- Web UI
- 30 720
-

- InfluxDB 60
 -
 - InfluxDB
-

配置数据库

配置

1. 配置数据库

```
# 数据库密码  
password: "omnitouch2024"  
  
# 数据库连接  
password: System.get_env("DB_PASSWORD") || "default_password"
```

2. 配置权限

```
chmod 600 config/runtime.exs  
chown ran_monitor:ran_monitor config/runtime.exs
```

3. 配置忽略文件

- 将 runtime.exs 添加到 `.gitignore`
- 配置数据库连接
- 配置密码

配置

1. 配置数据库

- 配置数据库
- 配置数据库连接池大小 `pool_size`
- 配置数据库连接池大小 2 个

2. 配置

- 配置数据库
- 60 秒
- 配置数据库 15 秒

3. InfluxDB 設定

- 設定ファイルを確認
- InfluxDB のインストール
- InfluxDB の起動

確認

1. インストール

- インストールディレクトリ IP 設定
- RAN のインストール
- インストールの確認
- インストールの完了

2. 設定

- `debug` 設定の確認
- `info` 設定の確認
- 設定ファイルの確認

3. RAN の起動

- インストールの確認
- インストールの確認
- InfluxDB のインストール
- インストールの確認

確認

1. インストール

- `runtime.exe` のインストール
- インストールの確認
- インストールの確認
- RAN のインストール

2. 設定

```
# 1. 設定 runtime.exs
vim config/runtime.exs

# 2. Elixir を起動
elixir -c config/runtime.exs

# 3. サービスを再起動
systemctl restart ran_monitor
```

3. 検証

- CPU使用率が低い
- メモリ使用率が低い
- 50+ の InfluxDB データが蓄積されている
- MySQL と InfluxDB が正常に動作している

□□□□□□□□

□□□□□□□□□□□□□□□□□□□□

```

import Config

#
=====
# 
#
=====

config :ran_monitor, RanMonitor.Repo,
  username: System.get_env("DB_USERNAME") || "ran_monitor_user",
  password: System.get_env("DB_PASSWORD") || "change_this_password",
  hostname: System.get_env("DB_HOST") || "localhost",
  database: "ran_monitor",
  stacktrace: false, # 
  show_sensitive_data_on_connection_error: false, # 
  pool_size: 15 # 6 * 2 + 3 

#
=====
# Web 
#
=====

config :ran_monitor, RanMonitor.Web.Endpoint,
  http: [ip: {0, 0, 0, 0}, port: 8080],
  check_origin: false,
  secret_key_base: System.get_env("SECRET_KEY_BASE") ||
"generate_with_mix_phx_gen_secret",
  server: true

config :control_panel, ControlPanelWeb.Endpoint,
  url: [host: "0.0.0.0", port: 9443, scheme: "https"],
  https: [
    ip: {0, 0, 0, 0},
    port: 9443,
    keyfile: "priv/cert/server.key",
    certfile: "priv/cert/server.crt"
  ]

config :ran_monitor, RanMonitor.Web.Nokia.Airscale.Endpoint,
  url: [host: "0.0.0.0"],
  http: [ip: {0, 0, 0, 0}, port: 9076],
  server: true

```

```

#
=====
#   [] [] [] []
#
=====

config :logger,
  level: :info #   [] [] []

config :logger, :console,
  format: "$time $metadata[$level] $message\n",
  metadata: [:request_id]

#
=====
#   [] [] [] []
#
=====

config :ran_monitor,
  general: %{
    mcc: "001",
    mnc: "001"
  },
  nokia: %{
    ne3s: %{
      webhook_url: "http://10.179.2.135:9076/webhook",
      private_key: Path.join(Application.app_dir(:ran_monitor, "priv'
"external/nokia/ne.key.pem"),
      public_key: Path.join(Application.app_dir(:ran_monitor, "priv")
"external/nokia/ne.cert.der"),
      reregister_interval: 30
    },
    airscales: [
      #   [] A -   []
      %{
        address: "10.7.15.66",
        name: "Site-A-Main-Tower",
        port: "8080",
        web_username: "admin",
        web_password: System.get_env("BS_SITE_A_PASSWORD") || "passwo
      },

```

```
# Site A - Tower
%{
  address: "10.7.15.67",
  name: "Site-A-Backup-Tower",
  port: "8080",
  web_username: "admin",
  web_password: System.get_env("BS_SITE_A_PASSWORD") || "password",
},

# Site B - Indoor
%{
  address: "10.7.16.10",
  name: "Site-B-Indoor-DAS",
  port: "8080",
  web_username: "admin",
  web_password: System.get_env("BS_SITE_B_PASSWORD") || "password",
},

# Site C - Rooftop
%{
  address: "192.168.100.50",
  name: "Site-C-Rooftop",
  port: "8080",
  web_username: "admin",
  web_password: System.get_env("BS_SITE_C_PASSWORD") || "password",
},

# Lab - Airscale
%{
  address: "10.5.198.100",
  name: "Lab-Test-Airscale-01",
  port: "8080",
  web_username: "Nemuadmin",
  web_password: "nemuuser"
},

# Lab - Airscale
%{
  address: "10.5.198.101",
  name: "Lab-Dev-Airscale-02",
  port: "8080",
  web_username: "Nemuadmin",
  web_password: "nemuuser"
}
```

```

    ]
  }

#
=====
# InfluxDB
#
=====

config :ran_monitor, RanMonitor.InfluxDbConnection,
  auth: [
    username: System.get_env("INFLUX_USERNAME") || "monitor",
    password: System.get_env("INFLUX_PASSWORD") || "change_this_passw
  ],
  database: "nokia-monitor",
  host: System.get_env("INFLUX_HOST") || "10.179.2.135"

```

□□□□

- □□□□ - □□□□
- **AirScale** □□□□ - □□□□
- □□□□□□□□ - □□□□□□□□
- **Grafana** □□ - □□□□□□□□
- **API** □□ - REST API □□
- □□□□□□ - □□□□□□□□

MDT トラフィック TCE

トラフィック (TCE)

RAN Monitor トラフィックを解析する LTE/5G ネットワーク RF トラフィック

トラフィック TCE

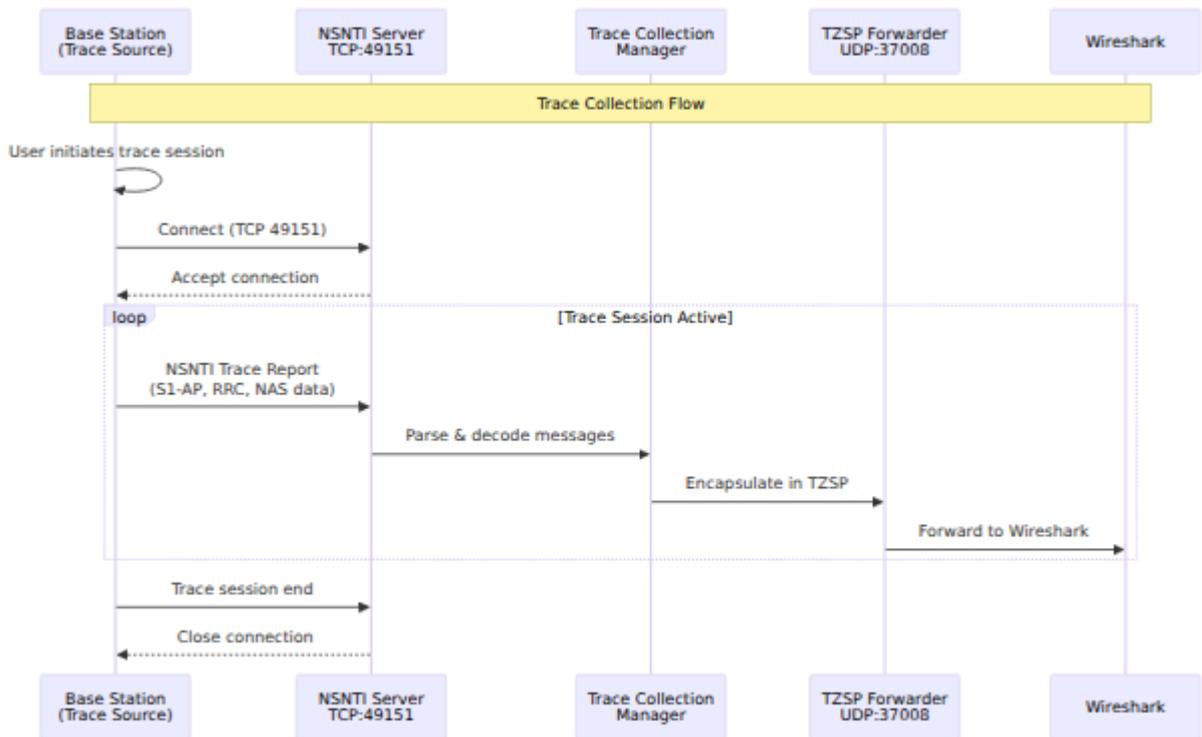
トラフィックを解析する Nokia AirScale ネットワーク

- **S1-AP** プロトコル - eNodeB と EPC 間の通信
- **RRC** プロトコル - 無線リソース制御
- **NAS** プロトコル - ネットワークアクセス層
- **パケット** - PDCP 層のデータ

TCE トラフィック

プロトコル	ポート	ポート	説明
NSNTI プロトコル	TCP	49151	トラフィック
TZSP プロトコル	UDP	37008	トラフィック Wireshark トラフィック
トラフィック	ASN.1	-	トラフィック S1-AP と RRC トラフィック

□□□□



□□□□□□□□□□□□□□□□ NSNTI □□□□ (49151) □ TZSP □□□□□□□□□□□□

□□□□□□□□

1. □□ TCE □□□□□□□□

```

ss -tlnp | grep 49151
# □□□□LISTEN 0.0.0.0:49151
  
```

2. □□□□□□□□

- □□□□□ IP □□□ RAN Monitor □□□
- □□□□□□□□□□□□□□ 49151
- □□□□□□□□□□□□□□ (S1-AP□RRC□NAS)
- □□□□□□□□

3. □□ Wireshark□

□□□□□□

- □□□ TZSP □□□□□□□□□□□□□□

- `udp port 37008`

□□□□□□

RAN Monitor □□□□ UDP □□□□□□□□□□ RRC □□□□ Wireshark □“□□□”□□□□
□□□□□□

□□ **1**□□□ **Wireshark GUI**

- **Analyze → Decode As...**
- + □□□□□□□□
-

Protocol	Port	Priority	QoS	Service
udp.port	36412	0	(none)	SCTP
sctp.port	36412	0	(none)	S1AP
udp.port	37000	0	(none)	TZSP
udp.port	37001	0	(none)	LTE RRC (DL-CCCH)
udp.port	37002	0	(none)	LTE RRC (DL-DCCH)
udp.port	37003	0	(none)	LTE RRC (BCCH)
udp.port	37004	0	(none)	LTE RRC (PCCH)
udp.port	37008	0	(none)	TZSP
udp.port	37011	0	(none)	LTE RRC (UL-CCCH)
udp.port	37012	0	(none)	LTE RRC (UL-DCCH)
udp.port	38000	0	(none)	MAC-LTE
udp.port	38001	0	(none)	MAC-LTE (DL)
udp.port	38002	0	(none)	MAC-LTE (BCH)
udp.port	38003	0	(none)	MAC-LTE (PCH)
udp.port	38011	0	(none)	MAC-LTE (UL)
udp.port	38012	0	(none)	MAC-LTE (RACH)

2 decode_as_entries

□□□□ □ ~/.config/wireshark/decode_as_entries (Linux/Mac) □

□%APPDATA%\Wireshark\decode_as_entries (Windows)□

```
# RAN Monitor TZSP □□□□
decode_as_entry: udp.port,36412,(none),SCTP
decode_as_entry: sctp.port,36412,(none),SIAP
decode_as_entry: udp.port,37000,(none),TZSP
decode_as_entry: udp.port,37001,(none),LTE RRC
decode_as_entry: udp.port,37002,(none),LTE RRC
decode_as_entry: udp.port,37003,(none),LTE RRC
decode_as_entry: udp.port,37004,(none),LTE RRC
decode_as_entry: udp.port,37008,(none),TZSP
decode_as_entry: udp.port,37011,(none),LTE RRC
decode_as_entry: udp.port,37012,(none),LTE RRC
decode_as_entry: udp.port,38000,(none),MAC-LTE
decode_as_entry: udp.port,38001,(none),MAC-LTE
decode_as_entry: udp.port,38002,(none),MAC-LTE
decode_as_entry: udp.port,38003,(none),MAC-LTE
decode_as_entry: udp.port,38011,(none),MAC-LTE
decode_as_entry: udp.port,38012,(none),MAC-LTE
```

□□□□□□□□

번호	계층	채널/신호	기능
36412	S1AP	-	네트워크 S1AP 인터페이스 (eNodeB ↔ EPC)
37000	RRC	신호	RRC 계층의 신호
37001	RRC	DL-CCCH	다운링크 공통 제어 채널
37002	RRC	DL-DCCH	다운링크 전용 제어 채널
37003	RRC	BCCH-DL-SCH	방송 채널 (다운링크) (MBSFN 전송 가능)
37004	RRC	PCCH	Paging Control Channel
37008	TZSP	-	테라지그비 전송 신호
37011	RRC	UL-CCCH	업링크 공통 제어 채널 (RRC 신호)
37012	RRC	UL-DCCH	업링크 전용 제어 채널 (신호)
38000	MAC-LTE	신호	MAC 계층의 신호
38001	MAC-LTE	신호	신호
38002	MAC-LTE	BCH	방송 채널
38003	MAC-LTE	PCH	Paging Channel
38011	MAC-LTE	신호	신호
38012	MAC-LTE	RACH	랜덤 액세스 채널

신호

```

# 0000 TZSP 000
tzsp

# 000000
slap || rrc || mac-lte

# 00000 RRC 00
udp.port == 37011 || udp.port == 37012

# 00000 RRC 00
udp.port == 37001 || udp.port == 37002

# 00 RRC 0000
rrc.rrcConnectionRequest || rrc.rrcConnectionSetup

# 000000
slap.HandoverRequired || slap.HandoverCommand

```

00

00000

- 000000 RF 00
- 000000
- 000000 (RSRP, RSRQ, SINR)
- 000000

00000

- 000000000
- 000000
- 000000
- 000000

RF 000

- PCI 0000
- 000000

- `rsrp-rsrq`
- `rsrp-rsrq`

00

MDT (MDT) UE (RSRP,RSRQ) Omnitouch RAN Monitor Web UI Nokia AirScale MDT Wireshark

00



TCE Omnitouch RAN Monitor Nokia Ne3s Wireshark

0000

00000

Nokia Airscale ONS

ONS

0000

- Omnitouch RAN Monitor TCE
- Wireshark 3.0
- Wireshark TCE Lua (TCE README)
- AirScale

MDT

RAN Monitor TCE Nokia Wireshark

1

RAN Monitor Web UI TCE

1. Web UI: `https://<ran-monitor-ip>:9443`
- 2.
- 3.
- 4.
- 5.
6. TCE
 - o IP: `<RAN Monitor IP>`
 - o : `49151`
- 7.
- 8.
- 9.

AirScale ONS

2 AirScale MDT

MDT

- MDT MDT
- PLMN
- UE 5000ms
- RSRP RSRQ
-

ONS

3. 실험 환경

이 실험은 AirScale를 사용하여 MDT를 생성하고 TCE를 실행하는 환경에서 수행됩니다.

Wireshark를 사용하여 MDT를 분석

Wireshark를 설치

1. Wireshark를 설치
2. Linux에서는 `lo`, macOS에서는 `lo0`, Windows에서는 `Loopback` 인터페이스를 선택
3. 필터를 `udp port 37008`로 설정
4. 캡처 시작

Wireshark를 사용하여 S1AP 메시지 (InitialUEMessage, Attach request)와 LTE RRC 메시지 (RRCConnectionReject, RRCConnectionReestablishment)를 분석하고 TCE를 실행합니다.

MDT를 분석

이 실험은 AirScale를 사용하여 MDT를 생성하고 TCE를 실행하는 환경에서 수행됩니다.

```
# RRC
lte-rrc.measurementReport

# RRC
udp.dstport >= 37011 && udp.dstport <= 37012

# RSRP (< -100 dBm)
lte-rrc.rsrpResult < 40

# RSRQ (< -12 dB)
lte-rrc.rsrqResult < 22
```

MDT

MDT RRC MeasurementReport

- RSRP RSRQ
- RSRP RSRQ
- ID ID
- **GPS** UE

Wireshark RRC

```
(RRC)
└─ UL-DCCH-Message
    └─ message: measurementReport
        └─ MeasurementReport
            └─ measResults
                ├── measResultServCell (RSRP/RSRQ)
                └─ measResultNeighCells ()
```

Wireshark

Wireshark

1. → → **CSV**
2. lte-rrc.rsrpResult lte-rrc.rsrqResult lte-rrc.physCellId

3. Excel Python

□□□□

□□□□□ RSRP/RSRQ □□□□

```
lte-rrc.rsrpResult < 40 || lte-rrc.rsrqResult < 22
```

□□□□□ UE □□□□□

```
lte-rrc.MeasResultListEUTRA
```

□□□□□ RSRP □□ RSRQ □□□□

```
lte-rrc.rsrpResult > 50 && lte-rrc.rsrqResult < 20
```

□□□□

Wireshark □□□□□

- □□ TCE □□□□□□ `ps aux | grep beam`
- □□ Wireshark □□□□□□ `udp port 37008` □□□□
- □□ AirScale □□□□□□□□□□□□
- □□□□□□□□ TCE IP/□□□□□□□□

□□□□□□

- □□□□□□□□□□ (MDT + □□□□□□)
- □□□□□□□□□□□□
- □□ UE □□ MDT (LTE Release 10+)

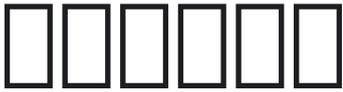
□□□□□□□□□□□□ **AirScale** □□□□□□□□ **ONS**□

□□□□□□□□

- MDT □□□□□□□□□□□□
- AirScale □□□ TCE IP (RAN Monitor IP) □□□ 49151
- RAN Monitor □□□□□□ TCE
- MDT □□□□
- Wireshark □□□□□□□□ `udp port 37008`
- `lte-rrc.measurementReport`
-

□□

- **Omnitouch** □□□□ (**ONS**)□□□□ AirScale □□□□□□□□□□□□



RAN 00000000

0000000000000000



1. 00
 2. 000000
 3. 000000
 4. Web UI 00
 5. 000000
 6. 0000
 7. 0000
 8. 0000
 9. 0000
-



000000000000 RAN 00



1. 0000
 - 000000000000
 - 00000000
 - 00000000
2. 0000

- 部署環境
- Web UI 環境
- 監視環境
- 通知環境

3. 監視環境

- 監視対象
- 監視方法
- 監視ツール
- 監視結果

4. 通知環境

- 通知対象
- 通知方法
- 通知ツール
- 通知結果

確認

確認事項

- RAN 監視環境 (ps aux | grep ran_monitor)
- データベース (MySQL, InfluxDB)
- 監視ツール
- 通知環境

確認結果

確認結果

確認

- Web UI 確認結果

- ping 测试
- telnet 测试
- 配置 Web UI

配置

1. ping 测试

```
# ping 测试
ping <device-ip>

# telnet 测试
telnet <device-ip> 8080
```

ping 测试 telnet 测试
 测试成功 - 配置成功

2. 配置

Web UI → 配置 → 配置 → 配置

- IP 地址
- 端口 8080
- 用户名

config/runtime.exs

```
%{
  address: "10.7.15.66", # IP
  name: "Site-A-BS1",
  port: "8080", # 端口
  web_username: "admin", # 用户名
  web_password: "password" # 密码
}
```

3. 验证

Web UI → 配置 → 配置

□□□

- [error] Authentication failed → □□□□□
- [error] Connection refused → □□/□□□□□
- [error] Timeout → □□□□□□
- [error] Certificate error → □□□□□/□□□□□

□□□□□

□□□□□

1. □□□□□□□□□□□□□□
2. □□ RAN □□□□□□□□□□□□□□
3. □□□□□□□□□□
 - RAN □□□ → □□□□ 8080
 - □□ → RAN □□□□□ 9076□webhooks□
4. □ RAN □□□□□□□□□□□□

□□□□□□

1. □□□□□□□□ WebLM □□□□□□□□
2. □□ config/runtime.exs □□□□□
3. □□ RAN □□□□
4. □□□□□□□□□□□□

□□/□□□□□□

1. □□□□□□□□□□□□□□
2. □□□□□□□□□□□□
3. □□□□□□□□□ telnet <device-ip> 8080
4. □□□□□□□□□□□□

□□□□□□/□□□□□□

1. □□□□□□□□□□□□
 - priv/external/nokia/ne.key.pem
 - priv/external/nokia/ne.cert.der
2. □□□□□□□□□□□□

3. 配置 Nokia 设备

4. 配置 Nokia 设备

配置 Nokia 设备

配置

- 配置 Nokia 设备
- 配置 Nokia 设备
- Web UI 配置 Nokia 设备
- 配置 Nokia 设备

配置

1. 配置 Nokia 设备

Web UI → 配置 → 配置 → 配置

- 配置 Nokia 设备
- 配置 Nokia 设备
- 配置 Nokia 设备

2. 配置 Nokia 设备

配置 `config/runtime.exs` 配置

```
nokia: %{  
  ne3s: %{  
    reregister_interval: 30 # 配置 30-60  
  }  
}
```

3. 配置 Nokia 设备

- 配置 Nokia 设备
- 配置 Nokia 设备 `ping <device-ip> -c 100`
- 配置 Nokia 设备

4. 設定

```
# RAN 設定  
date
```

```
# 設定  
# 設定
```

設定

設定

1. `reregister_interval` 30
2. RAN 設定
3. 設定

設定

1. 設定
2. 設定
3. 設定/設定
4. 設定

設定

1. RAN 設定 NTP
2. 設定
3. 設定

設定

1. 設定
 2. 設定
 3. 設定
-

❓❓ 問題問題問題

問題

- 問題 Web UI 問題問題“問題”問題
- 問題 InfluxDB 問題問題
- Grafana 問題問題
- InfluxDB 問題問題問題

問題

1. 問題問題

Web UI → 問題

- 問題問題
- 問題問題問題問題
- 問題問題問題

2. 問題 InfluxDB 問題

Web UI → InfluxDB 問題

- 問題問題
- RAN 問題問題 InfluxDB

問題

```
# 問題 RAN 問題問題  
curl http://<influxdb-host>:8086/ping
```

3. 問題問題

問題

- [error] InfluxDB write failed → 問題問題
- [error] Failed to collect metrics → 問題問題
- [info] Metrics collected: 0 → 問題問題

4. InfluxDB

InfluxDB

```
# InfluxDB 1.x
influx -database 'nokia-monitor' -execute '
  SELECT COUNT(*) FROM PerformanceMetrics
  WHERE basebandName=''Site-A-BS1''
  AND time > now() - 1h
'

# InfluxDB 2.x
influx query 'from(bucket:"nokia-monitor")
  |> range(start: -1h)
  |> filter(fn: (r) => r.basebandName == "Site-A-BS1")
  |> filter(fn: (r) => r._measurement == "PerformanceMetrics")
  |> count()'
```

InfluxDB

1. InfluxDB
2. `config/runtime.exs`
 -
 - 8086
 - /
 - /API
3. RAN
4. 8086
5. RAN

InfluxDB

1. /
2. InfluxDB
3. API
4. `config/runtime.exs`

5. RAN 監視

InfluxDB 監視

1. ディスク使用率 `df -h`
2. ディスク使用率
3. ディスク使用率
4. RAN 監視

監視

1. ディスク使用率
2. ディスク使用率 webhook URL 設定
3. ディスク使用率
4. RAN 監視 webhook 設定 (9076)

監視

監視

監視

- Grafana 監視
- ディスク使用率
- InfluxDB 監視

監視

1. ディスク使用率

🔍 🔍 ディスク使用率

```
# ディスク使用率  
journalctl -u ran_monitor --since "2025-12-29" --until "2025-12-30"
```

2. 設定

Web UI → 設定 → 設定 → 設定“設定”

- 設定
- 設定

3. 設定 InfluxDB 設定

設定 InfluxDB 設定

- 設定 InfluxDB 設定
- 設定/設定

設定

RAN 設定

- 設定
- 設定
- 設定

設定

- 設定
- 設定
- 設定
- 設定

InfluxDB 設定

- 設定
- 設定 RAN 設定
- 設定 InfluxDB 設定
- 設定

設定

- 設定 RAN 設定

- `docker run --rm -p 9443:9443`
- `docker run --rm -p 9443:9443`
- `docker run --rm -p 9443:9443`

Web UI

Web UI

Web UI

- `https://<ran-monitor-ip>:9443`
- `https://<ran-monitor-ip>:9443`
- SSL `https://<ran-monitor-ip>:9443`

Web UI

1. Web UI

Web UI

```
[info] Running ControlPanelWeb.Endpoint with cowboy
```

Web UI

```
ps aux | grep control_panel  
netstat -tulpn | grep 9443
```

2. Web UI

Web UI

```
telnet <ran-monitor-ip> 9443
```

Web UI

```
curl -k https://localhost:9443
```

3. 確認

```
# 確認
sudo iptables -L -n | grep 9443

# 
sudo firewall-cmd --list-ports
```

確認

確認

1. 確認

```
sudo firewall-cmd --add-port=9443/tcp --permanent
sudo firewall-cmd --reload
```

2. 確認

Web UI 確認

1. `config/runtime.exs` Web 確認
2. SSL 確認
3. 確認
4. RAN 確認

SSL 確認

1. 確認

```
ls -l priv/cert/omnitouch.pem
ls -l priv/cert/omnitouch.crt
```

2. 確認

```
openssl x509 -in priv/cert/omnitouch.crt -text -noout
```

3. `airmon-ng` 실행
4. `airmon-ng` RAN 실행

확인

1. `config/runtime.exe` 실행
2. `airmon-ng` 실행
3. `CONTROL_PANEL_HTTPS_PORT` 확인

Web UI 확인

확인

- Web UI 확인
- `airmon-ng` 실행
- `airmon-ng` 실행

확인

1. 확인

`config/runtime.exe` 실행

```
airscales: [  
  #   
]
```

2. 확인

MySQL 확인

```
mysql -u ran_monitor_user -p ran_monitor -e "SELECT * FROM  
airscales;"
```

3. 확인

확인

□□□□

□□□□□

1. □ `config/runtime.exs` □□□□
2. □□ RAN □□□
3. □□□□□□ Web UI □

□□□□□□□

1. □□ MySQL □□□□□□
 2. □□ `config/runtime.exs` □□□□□□
 3. □□□□□□□
 4. □□ RAN □□□
-

□□□□□

□□□MySQL □□□□

□□□

- □□□□□□□□□□□□□□□□
- Web UI □□□□□□□□□□
- “□□□□□□□□”□□

□□□□□

1. □□ MySQL □□□□□□

```
systemctl status mysql
# □
systemctl status mariadb
```

2. □□□□

□ RAN □□□□□□□□

```
mysql -h <mysql-host> -u <username> -p <database>
```

3. 配置

□ `config/runtime.exs` □

```
config :ran_monitor, RanMonitor.Repo,  
  username: "ran_monitor_user",  
  password: "password",  
  hostname: "localhost",  
  database: "ran_monitor",  
  pool_size: 10
```

□□□□

MySQL 配置

1. 安装 MySQL 服务

```
systemctl start mysql
```

2. 配置 MySQL 服务

3. RAN 配置 MySQL 服务

□□□□□□

1. 配置 MySQL 服务

2. 配置 MySQL 服务

3. 配置 MySQL 服务 `config/runtime.exs`

4. 配置 RAN 服务

□□□□

1. 配置 MySQL 服务

2. 配置 MySQL 服务 3306

3. 配置 MySQL 服务

□□□□

1. MySQL max_connections
 2. pool_size
 3. RAN
-

□□□□

□□□□ CPU □□□□□□

□□□

- RAN CPU RAM
- □□□□□□□□
- □□□□□□
- □□□□□□

□□□□□

1. □□□□□□□□

```
# CPU
top -p $(pgrep -f ran_monitor)

# □□□□□□
ps aux | grep ran_monitor
```

2. □□□□□□□□

□□□□□□□□

- □□□□ = □□□□□□□□
- □□□□□□□□□□□□

3. □□□□□□

□□□□□□□□□□

- CPU = CPU/메모리
- 메모리 10 배수

4. 모니터링

config/runtime.exs

```
pool_size: 10 #
```

모니터링

모니터링

1. 모니터링
2. 모니터링(CPU/RAM)
3. 모니터링
4. 모니터링

모니터링

1. 모니터링 pool_size
2. 모니터링 2 + 5 Web UI
3. RAN
4. 모니터링

모니터링

1. 모니터링
2. 모니터링
3. 모니터링 RAN
4. 모니터링

InfluxDB

1. InfluxDB
2. InfluxDB
3. InfluxDB
4. 모니터링

Web UI

- Web UI
-
-

1.

RAN

```
top
free -h
df -h
```

2.

```
# MySQL
mysql -u root -p -e "SHOW VARIABLES LIKE 'slow_query_log%';"
```

3.

- 1.
- 2.
- 3.

□□□□□

1. □□ MySQL □□
2. □□□□□□□□□□□□□□□□
3. □□□□□□□□□□

□□□□□

1. □□□□□□
 2. □□□□□□□□□□
 3. □□□□□□□□□□□□
-

□□□□□

□□□□□□□□□□

□□□

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

□□□□□

1. □□□□□□□□□□

□□□□□□□□□□□□□□□□□□

2. □□ **Webhook** □□□

Webhook □□□□□□□□□□

```
netstat -tulpn | grep 9076
```

□□□

```
tcp 0 0.0.0.0:9076 0.0.0.0:* LISTEN
```

3. Webhook

webhook URL RAN

```
http://<ran-monitor-ip>:9076/webhook
```

4.

webhook

5. InfluxDB

```
influx -database 'nokia-monitor' -execute '
SELECT COUNT(*) FROM Alarms WHERE time > now() - 1h
'
```

Webhook

1. `config/runtime.exe` webhook
2. 9076
3. RAN
- 4.

- 1.
2. webhook URL
- 3.

1. 配置 RAN 端口 9076
2. 配置
3. 配置 `telnet <ran-monitor-ip> 9076`

InfluxDB 配置

1. 安装 InfluxDB
 2. 配置
 3. 安装 InfluxDB 数据库
 4. 配置
-

配置

配置

安装 Web UI

1. 配置
2. 配置
3. 配置
4. 配置

配置

配置 systemd 配置

```
journalctl -u ran_monitor -f
```

配置 mix 配置

- 配置

配置

- 配置/配置/配置 - 配置

- 00 - 00000000
- 00 - 0000
- 00 - 00000000
- 00 - 00000000

00000000

- 00000000 "Site-A-BS1"
- "error" 0 "failed"
- "InfluxDB" 0 "MySQL"
- "registration" 0 "session"

InfluxDB 00

00000000

```
influx -database 'nokia-monitor' -execute '
SELECT * FROM PerformanceMetrics
WHERE basebandName=''Site-A-BS1''
AND time > now() - 5m
LIMIT 10
'
```

00000000

```
influx -database 'nokia-monitor' -execute '
SELECT COUNT(*) FROM PerformanceMetrics
GROUP BY basebandName
'
```

000000

```
influx -database 'nokia-monitor' -execute '
SELECT * FROM Alarms
WHERE time > now() - 1h
'
```

MySQL 操作

操作

```
SELECT name, address, port, registration_status  
FROM airscales;
```

操作

```
mysql -u ran_monitor_user -p ran_monitor -e "SHOW PROCESSLIST;"
```

操作

操作

```
# 操作  
ping <device-ip>  
  
# 操作  
telnet <device-ip> 8080  
nc -zv <device-ip> 8080  
  
# 操作  
traceroute <device-ip>
```

操作

```
# 操作  
sudo iptables -L -n -v  
  
# 操作  
sudo iptables -L -n | grep 8080
```

□□□□

□□□□□□

□□□□□□

1. □□□□

- □□□□□□□□
- □□□□□□
- □□□□□□□□

2. □□□□

- □□□□□□□□□□□□
- □□□□□□
- □□□□□□□□

3. □□□□

- RAN □□□□□□
- □□□□□□□□
- □□□□□□MySQL□InfluxDB□
- □□□□□□□□

4. □□□□

- □□□□□□□□□□
- □□□□□□□□
- □□□□□□□□□□

5. □□

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

□□□□

- **Web UI** □□ - □□□□□□
- □□□□□□ - □□□□
- □□□□□□□□ - □□□□
- **Grafana** □□□□ - □□□□
- □□□□□□□ - □□□□
- □□□□□□□□ - □□□□
- □□□□ - □□□□

□□□

□□□□□□□□

- Web UI □□□□□□□□□□
- □□□□ `journalctl -u ran_monitor`
- □□□□□□

□□□□□□□□

- □□□□□□
- □□□□/□□
- □□□□
- □□□□

□□□□□□□□

- □□□□ Web UI □□
- □□□□□□□□□□
- InfluxDB □□□□□□
- □□□□□□□□

□□

□□□□□□□□□□

Web UI

RAN -

RAN Monitor Web

- - Web UI
 -
 -
 -
 -
 -
 - eNodeB
 -
 -
 - InfluxDB
 -
 - PM
 -
 - Web UI
-

RAN Monitor Web Web UI

Web UI Grafana

Web UI

- Grafana
- Grafana
- Grafana
- Grafana
- Grafana

Grafana

- Grafana
- Grafana KPI
- Grafana
- Grafana
- Grafana

Grafana Grafana

Web UI

HTTPS

URL: `https://<ran-monitor-ip>:9443`

9443 `CONTROL_PANEL_HTTPS_PORT`

SSL

- Grafana
- Grafana CA
- Grafana `config/runtime.exs`

Grafana

□□□□

□□□□□ RAN □□□□□□□□□□

□□□□

□□□□

- □□□□□□
- □□□□□□□□□□□□

□□□□

- □□/□□□□□□□□
- □□□□□□
- □□□□□□□□

□□□□

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

□□□□

- □□□□□□□□
- □□□□
- □□□□□

□□

- □ 5 □□□□□□
- □□□□□□□□□□□□□□ = □□□□□□ = □□□□
- □□□□□□□□□□

- □□□□□□

□□□□

□□□□□□□□□□□□□□□□□□

URL: `https://<ran-monitor-ip>:9443/nokia/enodeb`

□□□□□□□□□□□□□□□□□□□□ NOKIA eNodeB □□□□□

□□□□

□□□□□□□□□□□□

□□	□□
□□□	□□□□□□□□
□□□	□□□□□□□□□□
□□□	□□□□□□□□
□□□	□□□□□□□□□□

□□□

□	□□
□□	□□□□□□□□
□□	□□□□□“□□□”□□□□□“□□□”□□□□□
□□	□□ IP □□□□□
□□	□□□□□“□□”□□□□□“□□□”□□□□□
□□	□□□□□□□

□□□□

□□□□□□□□□□□□

項目	説明
Ping	接続確認
項目	説明
項目	説明
項目	説明

接続確認

接続確認

項目	説明
項目 ID	説明
項目 ID	説明
項目	説明 COMA
項目	説明 Nokia

接続確認

- 項目
- 項目 IP 項目
- 項目

接続確認

接続確認

□□□□□□

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

□□□□□□

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

□□□□

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

□□□□

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

□□□□

- □□□□□
- □□□□□□□
- □□□□□
- □□□□□□

- 00000000
- 00000000

000000

- 000000000000000000000000
- 000000000000
- 0000000000

000000000000000000000000 00000000

000000

Web UI 000000000000000000000000

000000

000 000000000000

000

1. 000000000000
2. 00“0000”
3. 0000000000
4. 00000000 XML 00

000000 0000000000 0000000000

000000

000 000000000000

000

1. 00 XML 000000
2. 00“0000”

3. 000000000000“00”0
4. 000000000000 ID

000 0000000000 - 0000000

0000

000 00000000000000

000

1. 00000000 ID
2. 00“00”
3. 0000000000
4. 0000000000000000

000 00000000000000000000

0000

000 0000000000

000

1. 00000000 ID
2. 00“0000”
3. 0000000000
4. 0000000000/00

000 0000000000000000 - 00000000

0000000

00000

1. 0000000000
2. 000000
3. 0000000000 ID

4. 配置 eNodeBs
5. 配置 MME
6. 配置 HSS
7. 配置 DNS

配置 eNodeBs 使用 **AirScale** 设备

配置 eNodeBs

配置 eNodeBs 的步骤如下：

1.

配置 eNodeBs 的 IP 地址

- 配置 eNodeBs 的 IP 地址
- 配置 eNodeBs 的 DNS 服务器
- 配置 eNodeBs 的 NTP 服务器
- 配置 eNodeBs 的日志文件

2.

配置 ID

- 配置 eNodeBs 的 PLMN ID
- 配置 eNodeBs 的 Cell ID

3.

- 配置 eNodeBs 的频点
- 配置 eNodeBs 的带宽

4.

- 配置 eNodeBs 的功率
- 配置 eNodeBs 的邻区

3. RAN Monitor

...

...

- ...
- ...
- ...
- ...

...

...

...

...

- ...
- ...
- ...
- ...
- ...
- ...
- ...
- ...
- ...

...

...

...

- ...

- 500
-

- / -
- -
- -

- - / /
- -
- -
- -
- -
- -

-
- IP
-

-
-
-

-
-
-

- 安装依赖包

安装

- 安装依赖包
- 安装依赖包
- 安装依赖包

配置

- 配置数据库连接
- 配置日志
- 配置其他参数
- 配置其他参数
- 配置其他参数

部署

安装 InfluxDB 数据库

配置

安装

- 安装依赖包/库
- 在 `config/config.exs` 中配置
- 设置端口 720 和 30

运行

- 运行命令
- 运行命令

□□□□□

- □□□□□□□□
- □□□□□□

□□□□□

□□□□□□□□□□

□□□□□

- □□□□
- □□□□□□□□/□□□□
- □□□□□□□

□□□□□

- □□□□ - □□□□ PM □□□□□
- □□ - □□□□□□□□□□
- □□ - □□□□□□□□□□□□
- □□ - □□□□□□□□□□□□

□□□

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

□□□□□□□

1. □□□□ - □□□□□□□□□□□□□□□□
2. □□□□□ - □□□□□□□□□□□□□□□□
3. □□□□ - □□□□□□□□□□□□□□□□
4. □□□□ - □□“□□□□□□”□□□□□□□

□□□□□

- **720** □□□**30** □□ - □□□□□□
- **2160** □□□**90** □□ - □□□□□□□□□□
- **4320** □□□**180** □□ - □□□□□□□□□□
- **8760** □□□**365** □□ - □□□□□□

□□

- □□□□□□□□□□□□□□
- □□□□□□□□□□□□□□□□
- □□□□□□□□□□□□
- □□ InfluxDB □□□□□□

□□□ □□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□ 

InfluxDB □□□□

□□□□ InfluxDB □□□□□□□□□□□□□□

URL: `https://<ran-monitor-ip>:9443/nokia/influx`

項目	内容
項目	項目項目項目 InfluxDB 項目項目
項目	項目項目項目項目項目
項目	項目項目項目項目項目項目項目
項目項目	項目 InfluxDB 項目項目
項目	項目項目項目項目KB/s 項目 MB/s
項目項目項目項目	項目項目項目項目項目項目項目
項目	項目項目項目項目項目

項目項目 項目項目項目項目項目項目項目項目項目項目項目項目項目項目項目項目項目

項目項目

項目	内容
項目	項目項目項目項目項目項目
項目	項目項目項目項目項目項目項目
項目	項目項目項目

□□□□□□

□□	□□
□□	InfluxDB □□□□□□
□□	InfluxDB □□□□□□□□8086□
□	InfluxDB □□□
□□	□□□□□□
□□	□□□□□□□□3□PerformanceMetrics□Configuration□Alarms□

□□□□

□□□□□□□□□□

- **InfluxDB** □□□ - □□□□□□□□□□
- □□□□ - □□□□□□□□□□
- □□□□ - □□□□□□□□□□
- □□□□ - □□□□□□□□

□□□□

□□□ 30 □□□□□□□□□□

Queries

Query	Description
Overall Health	Overall Health
Overall Health + Alerts	Overall Health + Alerts
Overall Health + PM	Overall Health + PM
Alerts	Alerts InfluxDB
Alerts/PM	InfluxDB Alerts
Alerts	Alerts - Overall Health
Alerts	Alerts - Overall Health

Alerts

- Alerts InfluxDB
- Alerts
- Alerts
- Alerts
- Alerts

Alerts

InfluxDB Alerts

URL: <https://<ran-monitor-ip>:9443/nokia/metrics>

InfluxDB *Airscale*

InfluxDB

	InfluxDB
	InfluxDB

Airscale InfluxDB

InfluxDB

□	□□
Airscale	□□□□
□□□□	□□□ PM □□□□□
□□	□□□□□□□□□□
□□	□□□□□□□□□□
□□□	□□□□□□□□□□□
□□□□	□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□

□□□□

Erlang VM □□□□□□

□□	□□
□□□	□□□ Erlang VM □□□□
□□□□	Erlang □□□□□□□□
□□□□□	□□□□□□□□□□□XML□JSON □□□
□□□□	□□□□□□□□
□□□□	□□ Erlang □□□□□
□□□□	□□ CPU □□□□□□□□□□0 □□□□

- 导出数据 CSV

数据

- 导出数据
- 导出数据
- 导出数据 `pm_counters.csv` 导出数据

数据

导出数据 Nokia PM 导出数据

数据

- 导出数据 ID 导出数据
- 导出数据

数据

- 导出数据
- 导出数据 200 数据

数据

数据	数据	数据
LTE	M8xxx	LTE L1/L2/L3 导出数据
WCDMA	M5xxx	3G WCDMA 数据
5G-NR	M55xxx	5G NR 数据
5G-数据	M51xxx	5G 数据
5G-数据	M40xxx	5G 数据

□□□

□□□

- □□□□□ □ `priv/pm_filters.etf` □
- □□□□□□□□□□
- □□□□□□□□□□

□□□□□□

InfluxDB □□□□□□□□□□□□

- **PM** □□□□□□ PM □□□□□□□□□□
- □□□□□□□□□□□□/□□□□□□□□

□□□□□□□□□□□□□□□□ **PM** □□□□□□□□

□□□□□□□□

□□□□□□□□□□□□□□□□

URL: `https://<ran-monitor-ip>:9443/nokia/data`

[[ETS]] InfluxDB

ETS

Nokia	
Nokia	
InfluxDB	InfluxDB

ETS

□□□

1. □□□□□□
2. □□□□□□□□□□□□
3. □□□□□□□□□□
4. □□□□□□□/□□□□□□
5. □□□□□□□□□□
6. □□□□□□□□□□□□

□□□ < 5 □□

□□□□□

□□□ □□□□□□□□

□□□

1. □□□□□□□
2. □□□□□□□□□□□□
3. □□□□□□□□□□□□□□
4. □□□□□□□□□□
5. □□□□□□□□□□□□
6. □□□□□□□□□
7. □□□□□□□
8. □□□□□□□□□□

□□□□□□□□□□□□□□□□ □□□□□□□□

□□□□□

□□□ □□□□□□□□□□

□□□

1. □□□□□□□□□□
2. □□□□□□□□□□□□□□
3. □□□□□□□□□□

4. 設定 ID
5. ID 設定
6. 設定
7. 設定
8. 設定 15-30
9. 設定

設定

- 設定
- 設定
- 設定
- 設定

設定

設定 設定

設定

1. 設定 eNodeBs 設定
2. ID IP 設定
3. 設定 `config/runtime.exs`
4. RAN Monitor 設定
5. 設定
6. 設定
7. 設定 InfluxDB
8. 設定
9. 設定 Grafana 設定

設定 設定

設定

設定 設定

📌

1. 📌
2. 📌/📌
3. 📌
4. 📌
5. 📌ping 📌
6. 📌
7. 📌
8. 📌
9. 📌
10. 📌

📌 📌

📌📌📌

- 📌 - 📌
- 📌 - 📌
- 📌 - 📌
- 📌 - 📌
- **Grafana** 📌 - 📌
- 📌 - 📌
- 📌 - 📌

RAN 部署指南

部署指南 (RAN) 部署指南

OmniTouch 部署指南

目录

1. 简介
2. RAN 部署
3. 配置
4. Web UI 部署
5. Grafana 部署
6. 验证
7. 故障排除
8. 附录
9. 索引

简介

RAN 部署指南 Nokia AirScale 部署 3GPP LTE 及 5G 部署指南。RAN 部署指南

部署

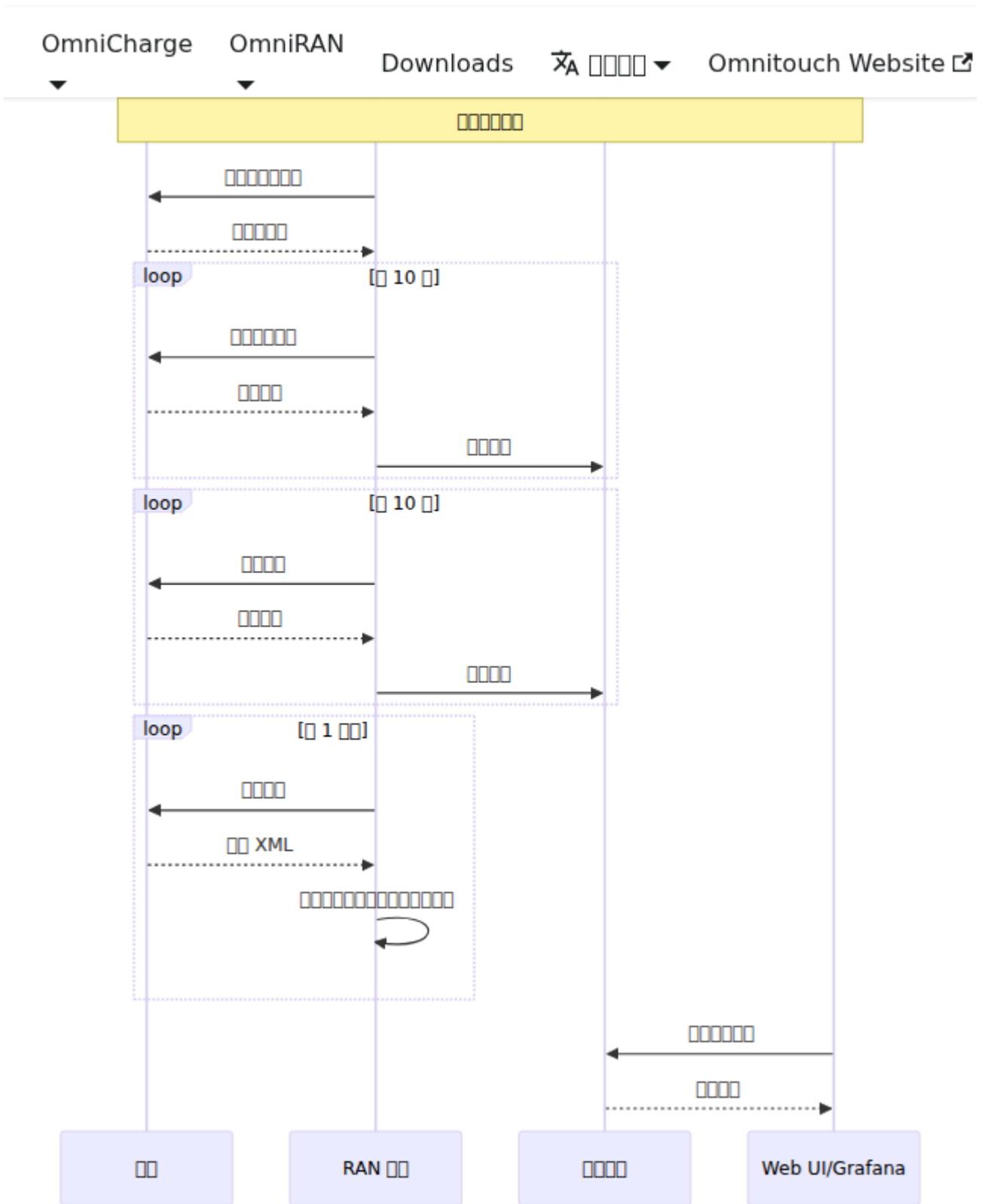
- 部署 - 部署指南
- 部署 - 部署指南
- 部署 - 部署指南
- **Web** 部署 - 部署 Web UI 部署指南
- **Grafana** 部署 - 部署指南

RAN 000000

RAN 000000000000

1. 000000 - 0000 Nokia 0000000000
2. 0000000 - 0 10 000000 KPI000000
3. 00000 - 000000000000
4. 00000 - 000000000000
5. 0000000 - 00000000000000
6. 000000 - 00 Web UI 0 Grafana 000000

□□□

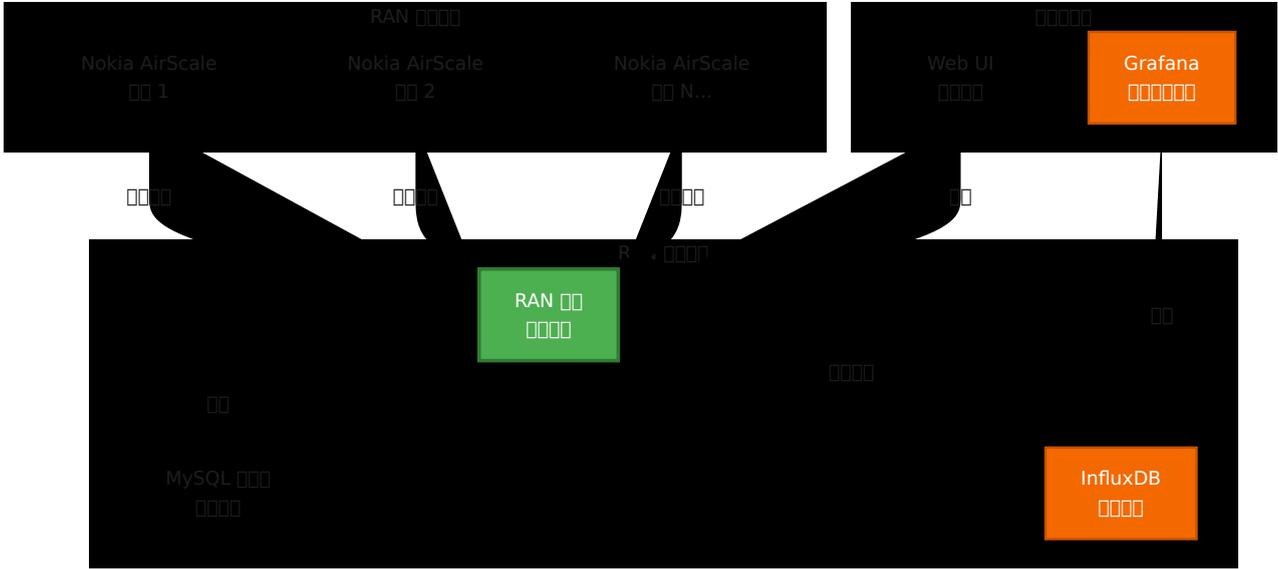


□□□□□

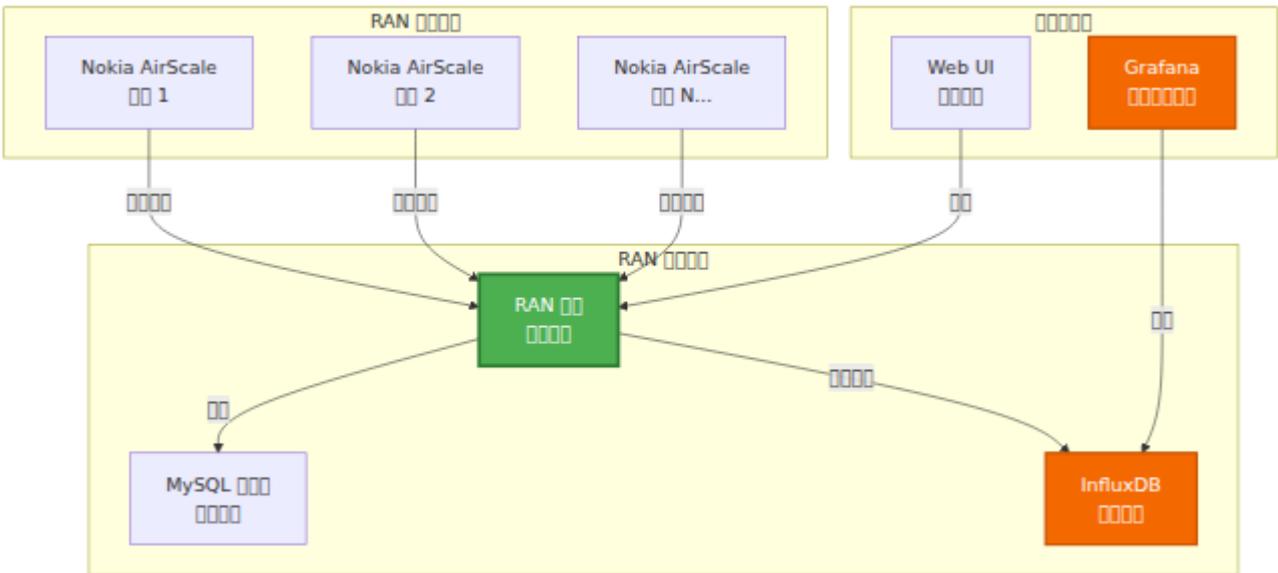
□□□□□

□□□□

□□□□□□



??□□□



□□□□□□□□□□□□□□ □□□□□□□□



Network (TCE)

RAN Network Architecture for LTE/5G

Network TCE

Network Architecture Nokia AirScale

- **S1-AP** - eNodeB to EPC
- **RRC** - RRC
- **NAS** - NAS
- **PD** - PDCP

Network

Network

- Network
- Network
- Network RSRP, RSRQ, SINR
- Network

Network

- Network
- Network
- Network
- Network

Network

- PCI
- Network
- Network
- Network

Network Wireshark Network TCE MDT

Web UI

RAN Web UI

<https://<ran-monitor-ip>:9443>

-
-
-
-

5

- = =
-
-
-

□□□□

□□□□□□□□□□

- □□□□□□□□□□□□□□ = □□□□□ = □□□□□ = □□□□□ = □□□□□ = □□□□
- □□□□□□□□□□□□□□
- □□□□□□□□□□
- □□□□□□□□□□

□□□□□□□□□□□□□□□□ □□□□□□□□

📄 eNodeB

📄📄📄📄📄📄📄📄

- 📄 ID 📄📄📄📄📄📄📄
- 📄📄📄📄📄
- 📄📄📄📄
- 📄📄📄📄📄📄📄📄

📄📄 📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄 ID 📄📄📄📄 `config/runtime.exs` 📄

□□□□

□□□□□□□□□□□□□□

- □□□□□□□□□□□□□□
- □□□□□□□□
- □□/□□□□□□
- □□□□□□□□□□
- □□□□□□□□

□□□□□□□□□□□□□□ □□□□□□

□□□□□

□□□□ InfluxDB □□□□□

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

□□□□□□□□□□□□ □□□□□□□□

InfluxDB 简介

时间序列数据库

- 分布式
- 可扩展
- 易用
- 高性能
- 支持 5 种数据类型

特点

- 查询 + 写入 = 高性能
- 查询 + 写入 = 可扩展
- 易用 = 易用 InfluxDB 接口

Web UI

Web UI

Web UI -

Grafana

Web UI Grafana

Grafana

Grafana

-
- KPI
-

- 配置管理
- 可用性 SLA
- 性能管理

Web UI 管理

- 配置管理
- 可用性
- 性能
- 告警
- 日志

Grafana 仪表盘

仪表盘

仪表盘

- 配置管理
- 可用性
- 性能
- 告警

□□□□

□□□□

□□□□□□ **5-10** □□□□

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

□□□□□□□□□□ **Web UI** □□ - □□□□□□

□□□□□

1. □□□□□□□□□□□□
2. □□□□□□□□□□□□
3. □□□□□□□□□□
4. □□□□□□□
5. □□□□□□□□□□□□

□□□□□□□□□□□□□□□□ □□□□□□□□

□□□□□

□□□□□□

1. □□□□□□□□□□□□
2. □□□□□□□ eNodeB □□□□□□□□
3. □□□□□□□ `config/runtime.exs`
4. □□ RAN □□□
5. □□□□□□□□□□□□□□
6. □□□□□□□ InfluxDB

□□□□□

1. 部署环境准备
2. 在 `config/runtime.exs` 中配置
3. 部署应用
4. 配置 RAN 服务
5. 配置 Grafana 监控

部署应用

1. 部署应用
2. 在 `config/runtime.exs` 中配置
3. 配置 RAN 服务
4. 部署应用

部署应用

部署应用 - 部署应用

Network Operations Center (NOC) Overview

Category	Item	Description
Web UI	Dashboard	Real-time network status overview
Alerts	Configuration	Alert threshold and notification settings
Alerts	Management	Alert suppression and acknowledgment
Alerts	Reporting	Alert history and trend analysis
Alerts	Integration	Integration with external monitoring tools
Alerts	Automation	Automated alert response workflows
Alerts	Documentation	Alert handling procedures and playbooks

Integration Tools

Tool	Integration	Capabilities
Grafana	Visualization	Real-time network performance dashboards
Nokia	Monitoring	Network health and performance monitoring
AirScale	Configuration	Network configuration and management
TCE MDT	MDT, Wireshark	Network traffic analysis and packet capture
API	REST API	Automated network management and reporting

□□□□□□

□□	□□	□□□□
□□□□□□	□□□□□□	□□□□□□□□

□□□□

□□□□ **RAN** □□□

1. □ **Web UI** □□ □□□□□□
2. □□ □□□□□□ □□□□□□□□
3. □□ □□□□□□ □□□□□□
4. □ □□□□□□□ □□□□□□□□□□

□□□□□□

1. □□ **Grafana** □□□□ □□□□□□□□
2. □□ **Nokia** □□□□□□ □□□□□□
3. □□ □□□□□□□□ □□□□□□

□□□□□□

□□□□

□□	URL	□□
Web UI □□□□	https://<server>:9443	□□□□□□□□
Grafana	http://<server>:3000	□□□□□□□□
InfluxDB	http://<server>:8086	□□□□□□□□□□□□□□

□□□□

□□	□□
<code>config/runtime.exs</code>	□□□□□□□□□□□□□□□□
<code>priv/cert/</code>	HTTPS Web UI □ SSL □□
<code>priv/external/nokia/</code>	□□□□□□□
<code>priv/airscale_configs/</code>	□□□□□□□ XML □□□

□□□□

□□□□

- RAN □□□□□□□□□□
- □□□□□□□□□□□□□□
- □□□□□□□□□□□□□□ 30 □□
- □□□□□□□ MySQL □□□□

□□□□

- □ 10 □□□□□□□□□□□□□□
- □ 10 □□□□□ + □□ Webhook □□□□
- □ 1 □□□□□□□□□□□□□□□□□□
- □□□□□□□□□ InfluxDB □□□□□□□□

□□□□

- □□□□□720 □□□30 □□
- □□□□□□□
- □□□□□□□□□□
- □□ Web UI □□□□□□

□□□□□□□□□□□□ □□□□□□□□□□

□□□□□□

□□□□□□

1. □□ Web UI → □□□
2. □□□□□□□□□□□□
3. □□□□□□
4. □□□□□□

□□□□□□

1. Web UI → □□ → □□□□□□
2. □□□□□□□□□□□□
3. □□□□□□
4. □□□□□□□□□□□□
5. □□□□□□
6. □□□□□□□□

□□□□□□

1. □□□□□□
 2. □□ `config/runtime.exe`
 3. □□□□□□ airscapes □□
 4. □□ RAN □□
 5. □□□□□□□□□□□□
-

□□

□□□□□□

□□	□□
□□□□□□	□□□□□□□□
□□□□□□	□□□□□□□□
□□□□□□	□□□□□□□□
InfluxDB □□□□	□□□□□□

□□□□□□

□□□□□□

1. □□□□□□ → □□□□
2. □□□□□□□ ping <device-ip>
3. □□ config/runtime.exe □□□□
4. □□□□□□□□□□

Grafana □□□□□□

1. □□□□□□□□□□□□□□
2. □□ InfluxDB □□□□□□□□□□
3. □□ InfluxDB □□□
4. □□ Grafana □□□□□

Web UI □□□□□

1. □□□□ 9443 □□□□□
2. □□□□□□□□□□ HTTPS □□
3. □□ SSL □□□□□□
4. □□□□□□□□□□ Web UI □□□□

□□□□□□□□□□□□ □□□□□□

□□□□

□□□□□□□□

□□□□□□□□

- □□□□□□□□□□
- □□□□□□□□□□
- □□□□□□□□□□/IP□
- □□□□□□□□
- RAN □□□□□□□□□□

□□□□

□□ RAN □□□□□□□□

- Omnitouch □□□□□□□□
- □□□□□□□□□□
- □□□□□□□□□□□□□□
- □□□□□□□□□□

□□□□□□

1. □□ □□□□□□□□
 2. □□□□□□□□□□□□□□
 3. □□□□□□□□□□
 4. □□□□□□□□□□□□
 5. □□□□□□□□□□
-

□ □ □ □

