

Nokia AirScale 部署

RAN Monitor 部署

前提

- 网络
 - 数据库
 - 部署 WebLM 部署
 - 部署 RAN Monitor
 - 部署 RAN Monitor
 - 部署
 - 部署
-

部署

部署 RAN Monitor 部署 Nokia AirScale 部署 RAN Monitor 部署 Nokia Web Element Manager (WebLM) 部署

部署 (PMCADM) 部署

部署

部署 Nokia AirScale 部署 **Omnitouch** 部署

- 部署
- 部署
- 部署 部署
- 部署

部署 **Omnitouch** 部署 **Omnitouch** 部署

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>/
```



https://<base-station-ip>/

3.

2

1. **Configuration**
2. **Configuration Management**
3. **Parameter Editor**

1 **OAM TLS**

OAM TLS `oamTls` `forced`

()

1. **Current BTS configuration**
2. **CURRENT_BTS_CONF-1**
3. **MRBTS-X** (X ID)
4. **oamTls**
5. `forced` `off`

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
oamTls	forced	off

項目	設定	説明
TLS 有効/無効	有効	RSA (2048ビット TLS)
監視 IP アドレス	RAN Monitor IP 10.179.2.139	監視 RAN Monitor IP (10.179.2.139)
監視ポート	RAN Monitor ポート 9076	9076 (ポート)
監視間隔	60s	60s (秒)
TLS 証明書	有効	false (無効)

設定

項目	設定	説明
SDL 有効/無効	有効	1
SDL Nonce	12345678	12345678
SDL 有効/無効	有効 0 有効	false (無効)

ステップ 4: 監視計画の作成

手順

- 監視計画 - 監視 IP アドレスを設定
- 監視計画
 - 監視計画 **Create Plan** をクリック
 - 監視計画を選択
 - 監視計画 ID を入力
- 監視計画

- **Validate Plan**
- ID
-
-

4.

- **Activate Plan**
- ID
-
-

XML

XML PMCADM

```
<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

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

- `tlsEnabled` - false
- `sdlPrimaryDestIp` - RAN Monitor IP

RAN Monitor IP ID MRBTS-132

LNBTS

LNBTS (LTE NodeB) PM

actL1PM - 1

	actL1PM
	NOKLTE:LNBTS
	(true/false)
	false
	RTWP (M8005C306-339)

1 () RTWP () RTWP

- M8005C306-309: AVG_RTWP_RX_ANT_1 AVG_RTWP_RX_ANT_4
- M8005C310-313: AVG_RTWP_RX_ANT_5 AVG_RTWP_RX_ANT_8
- M8005C314: MAX_RTWP
- M8005C317-325: UL_IOT_PUSCH
- M8005C326-335: AVG_SIR

XML

```
<managedObject class="NOKLTE:LNBT" distName="MRBT-XXX/LNB-XXX" operation="update">  
  <p name="actL1PM">true</p>  
</managedObject>
```

WebLM

- Configuration > Configuration Management > Parameter Editor
- MRBT-X > LNB-X
- actL1PM
- true
-

actPCMDReport -

Field	Value
actPCMDReport	
NOKLTE:LNBT	
	(true/false)
	false

(PCMD)

XML

```
<managedObject class="NOKLTE:LNBT" distName="MRBT-XXX/LNB-XXX" operation="update">  
  <p name="actPCMDReport">true</p>  
</managedObject>
```

PMCADM-1

PMCADM ()

-
-
-
-

PMCADM

reportingIntervalPm - PM

	reportingIntervalPm
	com.nokia.srbts.mnl:PMCADM
	5min, 15min, 30min, 60min
	5min 15min

- **5min** -
- **15min** -
- **60min** -

XML

```
<managedObject class="com.nokia.srbts.mnl:PMCADM" distName="MRBTS-XXXX/MNL-1/MNLENT-1/PMCADM-1" operation="update">
  <p name="reportingIntervalPm">5min</p>
</managedObject>
```

WebLM

- Configuration > Configuration Management > Parameter Editor
- MRBTS-X > MNL-1 > MNLENT-1 > PMCADM-1
- reportingIntervalPm
- 5min
-

certTypeForTlsAuth - TLS

- (RSA, DSA, ECDSA)
- TLS
- RSA
- tlsEnabled = true

rTpmCollEntityHost -

- IP (IPv4 or IPv6)
- IP
-
- 10.179.2.139
-

rTpmCollEntityPortNum -

- (1-65535)
- TCP

- 9076
- RAN Monitor

rTpmCollInterval -

- ()
-
- 15s, 30s, 60s, 300s, 900s, 1800s
- 60s
- 60s 15s

tlsEnabled - TLS

- (true/false)
-
- false
-

sdIMaxUploadFileNumber - SDL

-
-
- 1
-

sdINonce - SDL Nonce

- (8)
- SDL
- 12345678
-

suppressZeroValueCount -

- (true/false)
-
- false

- `status` `false` `status`
-



Configure RAN Monitor `status`

Configure RAN Monitor Web UI

1. RAN Monitor Web UI: `http://<ran-monitor-ip>:4000/`
2. `eNodeB Status`
3. `status`
4. `Connected` (`status`)
5. `Session` `Active`

Configure

- `Connected` (`status`)
- `Active`
- `status` `IP` `status`

- 0000 0000000

00000000 **Pending**

000000000000000000000000

0000000

- 0000 ID 0000000000
- RAN Monitor 0000000000
- 0000000

00000000000000

0000000 RAN Monitor

□□□□□□

- PMCADM □□□□ IP □□□□□□
- □□□□□□
- □□□□□□ 8080
- RAN Monitor □□□□□□

□□□□□□

□□ **InfluxDB** □□□

1. □ RAN Monitor Web UI □□□□□ **InfluxDB Status**
2. □□□□□□□□□□
3. □□ **Performance Metrics** □□□□□□□
4. □□ **Last Update** □□□□□□□

□□□□

- **Performance Metrics** □ □□□□□
- **Configuration** □ □□□□
- **Alarms** □ □□□□□□□□□□□ 0

□□□□□□

1. □□□ **Data Retention Policy** □□
2. □□□□□□
3. □□ **Performance Metrics** □ **Configuration** □ **Alarms** □□

□□□□

□□□□□ **RAN Monitor** □

□□□ □□□□□□ eNodeB □□□□

□□□□□

1. □□□□□□□

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

2. □□ **RAN Monitor** □□□

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

3. RAN Monitor

- o Live Logs
- o
- o

4.

- o WebLM PMCADM-1
- o RAN Monitor IP
- o 9076

“Pending”

“Pending”

1.

- o RAN Monitor ID
- o

2.

- o runtime.exs
- o

3.

- o 30-60
- o

“enetunreach”

1. 設定

- 設定 RAN Monitor 設定
- 設定
- 設定 VLAN/subnet 設定

2. 設定

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

3. 設定 RAN Monitor 設定

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

RTWP 設定 1 設定

設定 InfluxDB 設定 RTWP 設定 (M8005C306-339)

設定 LNBTS 設定 `actL1PM` 設定 `false` 設定 1 設定

設定

1. 設定

- 設定
- 設定 LNBTS 設定 `actL1PM`
- 設定 `false` 設定

2. 設定 1 PM

- 設定 WebLM
- 設定 MRBTS-X > LNBTS-X
- 設定 `actL1PM` 設定 `true`
- 設定

3. 設定

- 監視間隔 (単位 60 秒)
- InfluxDB 接続 M8005C306-309 設定
- RTWP 監視

設定

- `actPCMDReport` - 監視 (監視)

InfluxDB 設定

監視 監視

監視

1. 監視

- PMCADM-1 rTpmCollInterval 設定
- 監視
- InfluxDB 監視

2. InfluxDB 設定

- InfluxDB 監視
- “監視”監視
- 監視

3. RAN Monitor 監視

- InfluxDB 監視
- 監視
- API 監視

TLS/監視

監視 監視 TLS 監視

□□□□

1. □□□□□□□□

- □□□□□□□□□□
- □□ RAN Monitor □□□□ CA □□

2. □□□□□ **TLS**□

- □ tlsEnabled □□□ false
- □□□□□□□□□□
- □□□□□□□□□ TLS

3. □□□□□□□□

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

□□□□

□□□□

- [□□□□](#) - □□□ RAN Monitor □□□□
- [□□□□□□□□](#) - RAN Monitor □□□□
- [Nokia Counter Reference](#) - □□□□□□□□
- [Grafana](#) □□ - □□□□□□□□□□□□
- [API](#) □□ - □□□□□ REST API □□
- [□□□□□□□□](#) - □□□□□□□□□□

□□□□

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

□□

□□□□□□□□□□

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



□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□



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

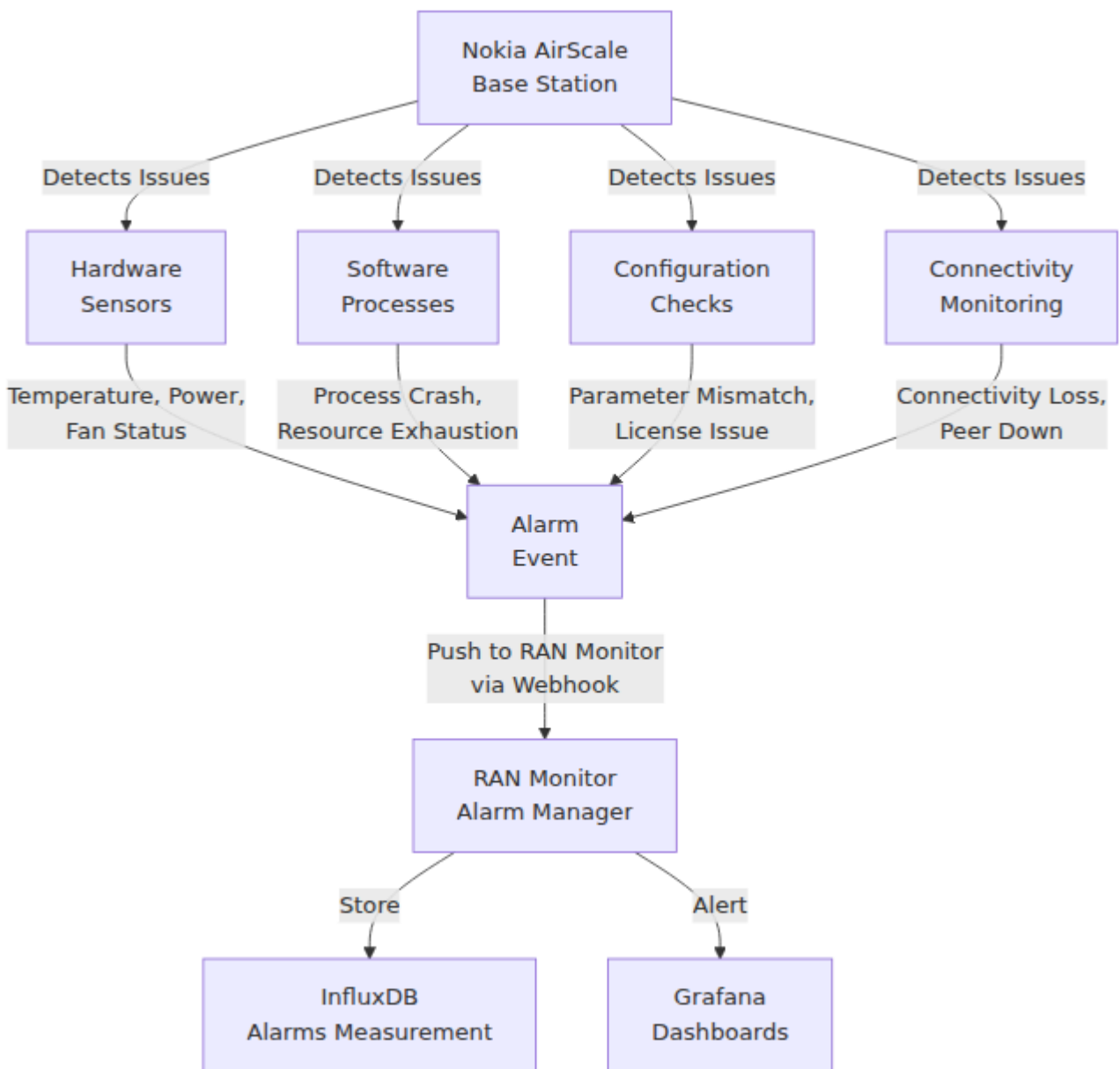


□□□□□□“□□”□□□□□□ AirScale □□□□□□□□□□□□ RAN Monitor □□□□□□□□□□□□□□□□
□□□□□□□□

□□□□□□□□

4G

□□□□



□□□□□□

□□□□□□

項目	内容	備考
識別ID	a1b2c3d4-e5f6-...	識別子
状態	正常	状態
名前	"テスト"	名前
場所	"S1 東京"	場所
位置情報	/BSC-1/BTS-23/Cell-A	位置情報DN
時刻	2025-12-10 14:23:45	時刻
タイプ	タイプ / 状態	タイプ

□□□□□□

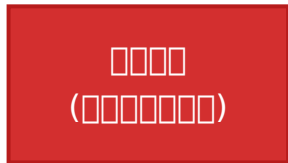
□□□□



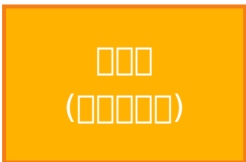
□□□□



□□□□



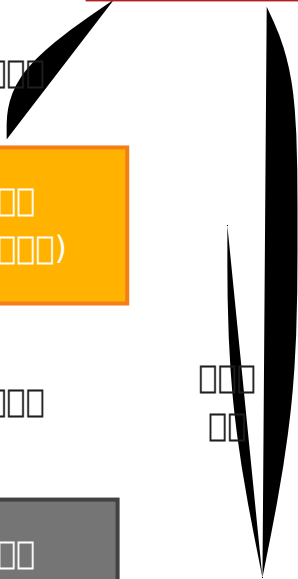
□□□□



□□□□



□□□
□□



□□□□□□

```
14:23:45 UTC - □□□□
  ↳ □□□□□□□□□□
  ↳ □□□□“S1 □□□□”□□□□

14:23:47 UTC - □□□□ RAN Monitor
  ↳ NE3S webhook □□□□
  ↳ □□□ InfluxDB
  ↳ □□□□□□

14:23:50 UTC - □□□□
  ↳ Grafana □□□□
  ↳ □ NOC □□□□ Slack □□
  ↳ □□ PagerDuty □□

14:24:15 UTC - □□□□□□
  ↳ NOC □□□□□□□□
  ↳ □□□□□□□□□□

14:28:35 UTC - □□000□□□□
  ↳ □□□□
  ↳ □□□□□□□
  ↳ RAN Monitor □□“□□□□”

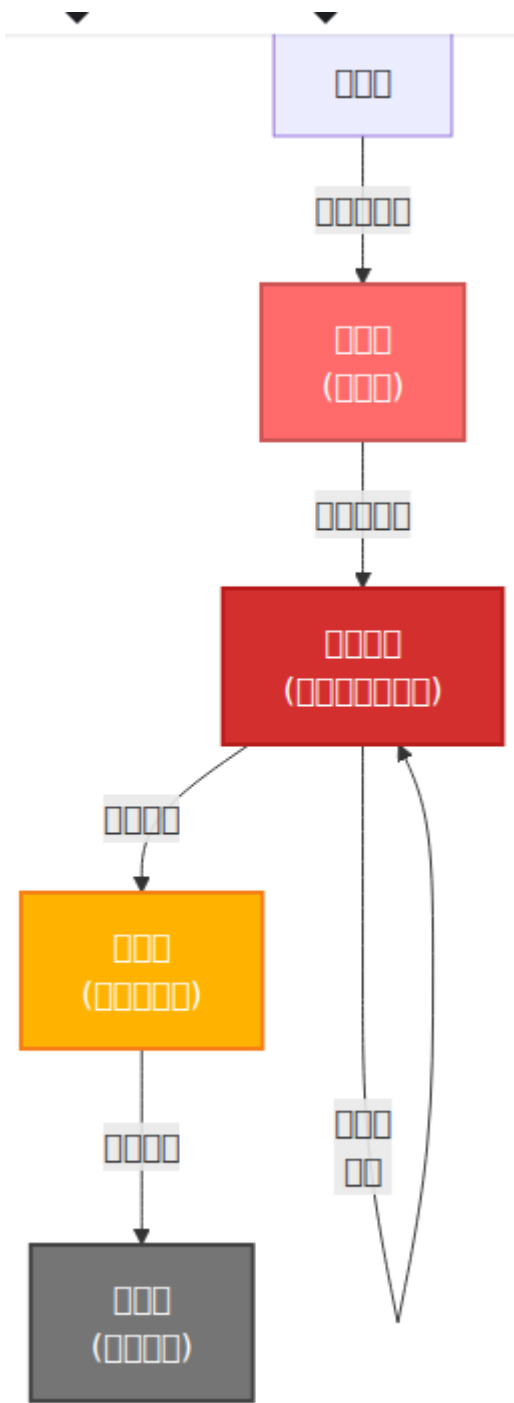
14:28:36 UTC - □□□□□□
  ↳ □□□□□□□5 □□ 51 □
  ↳ □□ SLA □□□□□□
  ↳ 30 □□□□
```

□□□□□□

RAN Monitor □□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□

□□□ □□□□□□□□□□□□□□



000

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

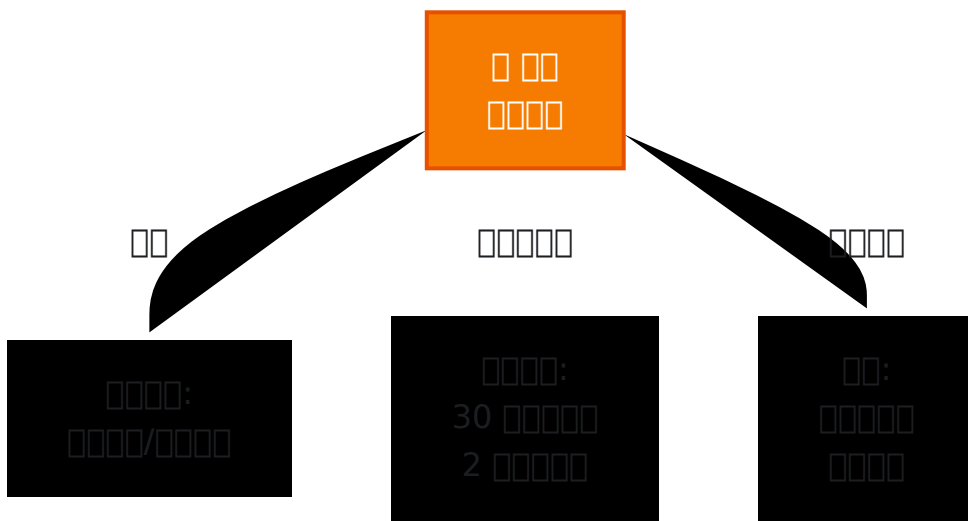
000

- 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



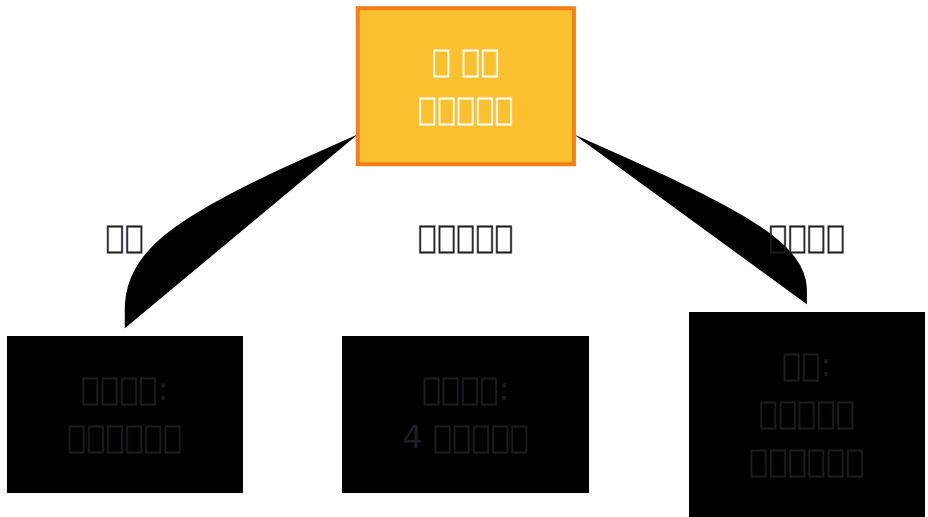
网络

- 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
- 网络性能 E1/T1 网络性能

网络

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

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. □□□□□□

□□□□

□□□□□□□

□□□□□

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

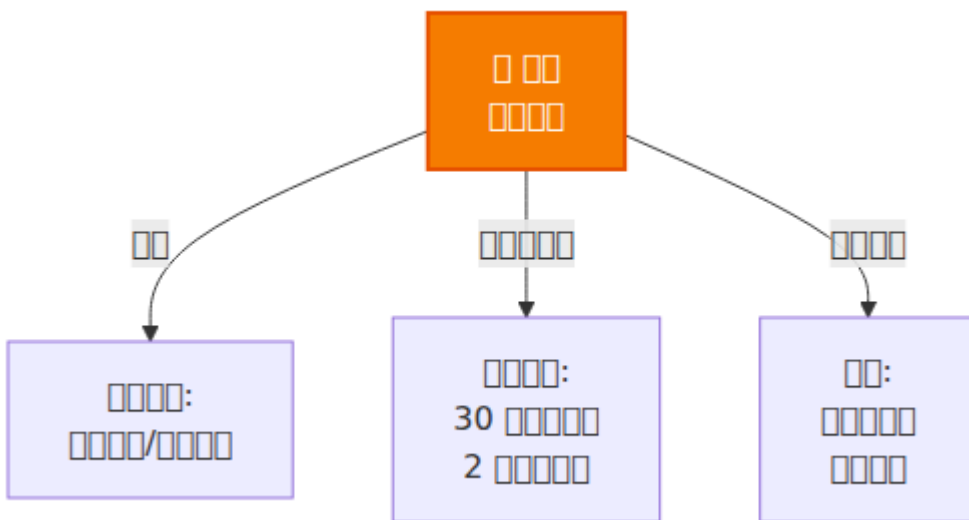
□□□□□□□□□□□

□□□

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 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
- 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 > 30 項目

項目項目項目

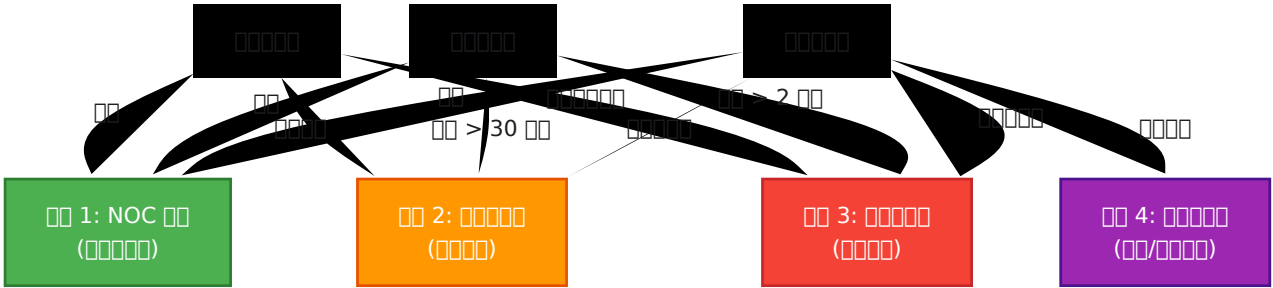
項目項目項目項目

項目

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

□□□□

□□□□



□□□□□□

□□□□□□□□□□□□□□□□ 2□

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

□□□□□□□□□□□□□□□□ 3□

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

□□□□□□□□□□□□□□□□ 4□□□□□□□□□□□□

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

0000

00000 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

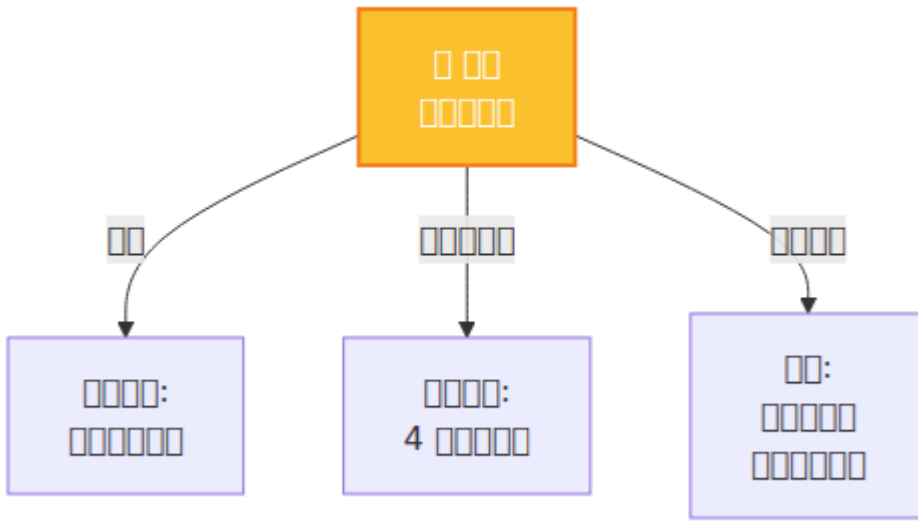
- SLA
- SLA
- SLA
- SLA
- MTTR SLA

SLA

- Grafana → SLA
- SLA

- 0000/00000 MTTR

00000000



000000

000000

1. 00000000

- 000000000000000000000000
- 000000000000000000000000
- 00000000
- 0000000000

2. 00000

- 000000000000
- 000000000000
- 000000000000
- 000000000000

3. 000000

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

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 監測ツール

このツールは RAN 環境向けの REST API

を監視するための API を RAN 環境に

目次

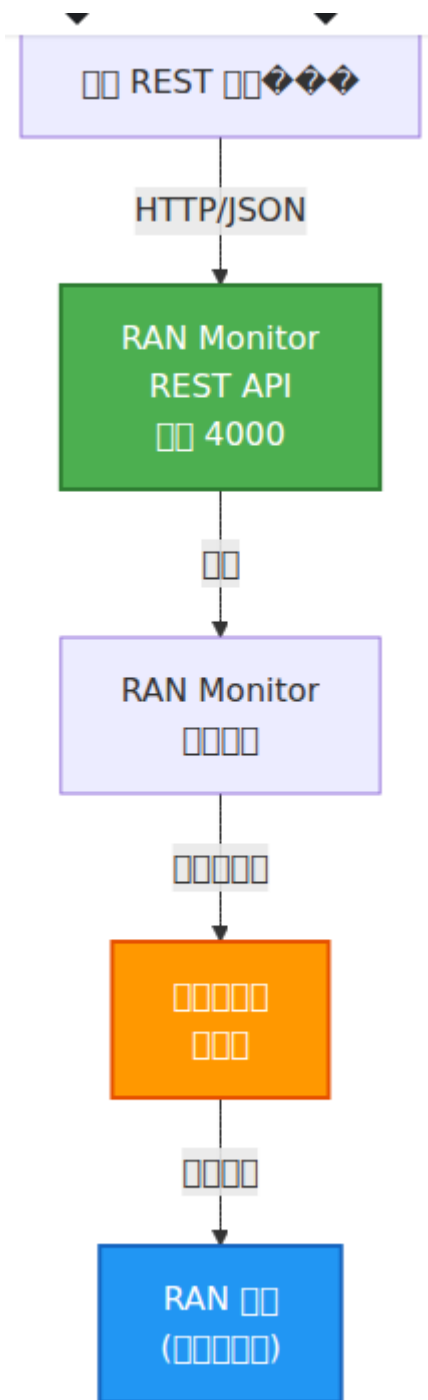
1. 概要
 2. API 一覧
 3. 監視対象
 4. 監視方法
 5. 監視結果
 6. 監視ログ
 7. 監視設定
 8. 監視エラー
 9. API 仕様
-

概要

RAN Monitor は REST API を通じて RAN 環境の監視を行うための API を提供します。

- 監視対象 - 監視対象の定義
- 監視方法 - 監視方法の定義
- 監視結果 - 監視結果の取得
- 監視ログ - 監視ログの取得
- 監視設定 - 監視設定の取得

API



API

- **RESTful** - HTTP (GET, POST, PUT, DELETE)
- **JSON** - JSON
- - API
- -

- `curl -X GET http://192.168.1.100:8080/health`
- `curl -X GET http://192.168.1.100:8080/status`

API

API

API RAN Monitor API RAN Monitor API

API

1. API
2. API
3. API
4. API

API

RAN Monitor API

- `curl -X GET http://192.168.1.100:8080/health`
 - `Authorization: Bearer <api-key>`
 - OAuth2
 - VPN
- `curl -X GET http://192.168.1.100:8080/status`
 -
 -
- `curl -X GET http://192.168.1.100:8080/health`
 - 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 00

Python 00000

```
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):
        """000000"""
        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):
        """000000"""
        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"):
        """000000"""
        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"Device registered: {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 環境構築ガイド





本ガイドは、AirScale を InfluxDB と XML を利用して環境構築するための手順を説明します。



- **前提条件** - インストール環境の確認
 - **インストール** - インストール手順
 - **設定** - 環境変数の設定
 - **起動** - サービスの起動 (100 MB ~ 690 MB)
 - **Web** 管理 - Web 管理画面のアクセス
 - **ログ** - ログの取得方法
 - **InfluxDB** 接続 - InfluxDB への接続設定
 - **更新** - バージョン更新の手順
-



本ガイドは AirScale のインストール手順を説明します。

- **インストール** 1. インストール
- **インストール** 2. インストール
- **インストール** 3. インストール  
- **インストール** 4. インストール

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 日 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 <AirScaleName>_config_YYYYMMDD_HHMMSS.xml

目次 AirScale 目次

目次

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

目次

目次

目次

- 目次 100 MB
- 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

0000

```
0000000 95 MB 655 000000 145KB
├─ ONS-Lab-Airscale_config_20240101_100000.xml  <- 000145KB
├─ ONS-Lab-Airscale_config_20240102_100000.xml  (145KB)
├─ ... (00 653 000)
└─ ONS-Lab-Airscale_config_20251230_100000.xml  <- 000145KB
```

0 2025-12-31 10:00:00 0000000145KB

000

- 000ONS-Lab-Airscale_config_20251231_100000.xml (145KB)
- 00000095 MB + 145KB = 95.14 MB 000 100 MB 0000
- 00000
- 000656 000000 95.14 MB

000000000000000000000000 500KB

000

- 000ONS-Lab-Airscale_config_20251231_150000.xml (500KB)
- 00000095.14 MB + 500KB = 95.64 MB 00000000
- 00000
- 000657 000000 95.64 MB

00000000000000

00000099.8 MB 685 0000

0000200KB

- 00000
- 000000100 MB 00000000
- 000000000000 < 100 MB
- 000000“00 3 000000 435 KB”
- 000682 000000 99.6 MB

0000

0000000

- 00000 0000000000 100 MB
- 0000 00000000
- 000000 000000
- 000000 0000000000000000000000
- 000000 000000000000

0000

0000000000000000

```
# 00000000
du -sh priv/airscale_configs/
# 000215M03 000000 70 MB

# 00000000
du -sh priv/airscale_configs/*/
# 000
# 95M     priv/airscale_configs/ons-lab-airscale/
# 68M     priv/airscale_configs/sector-1/
# 52M     priv/airscale_configs/sector-2/

# 00000000000000
find priv/airscale_configs -maxdepth 1 -type d -exec du -sm {} \;
| awk '$1 > 90 {print $2 " is at " $1 "MB (approaching 100MB
limit)}'
```

000000000000

00000000

Web UI 0000 **Nokia** → 0000

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 または MB
制限	“2” または “3”
その他	

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. その他

その他の設定

- 項目 - 説明
- 項目 - 説明
- 項目 - 説明

項目

項目

時刻は UTC を使用します。

項目

- 項目
- 項目
- 項目

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 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 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

ディレクトリ ディレクトリ

手順

1. ディレクトリ
2. ディレクトリ
3. ディレクトリ Web UI
4. ディレクトリ → ディレクトリ ID
5. ディレクトリ → ディレクトリ
6. ディレクトリ
7. ディレクトリ
8. ディレクトリ

確認

- ✓ ディレクトリ
- ✓ ディレクトリ
- ✓ ディレクトリ
- ✓ ディレクトリ
- ✓ ディレクトリ

結果

ディレクトリ “ディレクトリ”

□□□□

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>

```

特徴

データサイズ ~145 KB の XML を AirScale で

仕様

データサイズ	データサイズ	データサイズ 145KB	データサイズ
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

注意事項

- データサイズ 100 MB の XML
- データサイズ ~690 の XML 145KB
- データサイズ XML current.xml の 145KB
- データサイズ XML 100 MB の XML
- データサイズ XML の XML

□□□□

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

□□□□

□□□□

- □□□□□□ **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

- 000000“00000000”
- 000000 404 00

000000

1. 00000000

- 000000000000000000000000
- 00 `priv/airscale_configs/` 00000000

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“`设备名称-配置名称`”

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. 検索

- 検索
- 検索

検索

```
# 検索
grep "process_configuration" <log-file>

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

検索

検索

検索 検索

検索

```
#!/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/"

# Clean up old backups
# Keep 30 days
find "$BACKUP_DIR" -type d -mtime +30 -exec rm -rf {} +

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

How to use cron

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

FAQ

How to backup files

How to backup

1. Backup files
2. Backup files
3. Backup files

How to backup

```
# 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
```

```
#####
```

```
##### 10 11 12
```

```
#####
```

```
#####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`
- □□□□□□□□“□□”□□□□□□□□□□

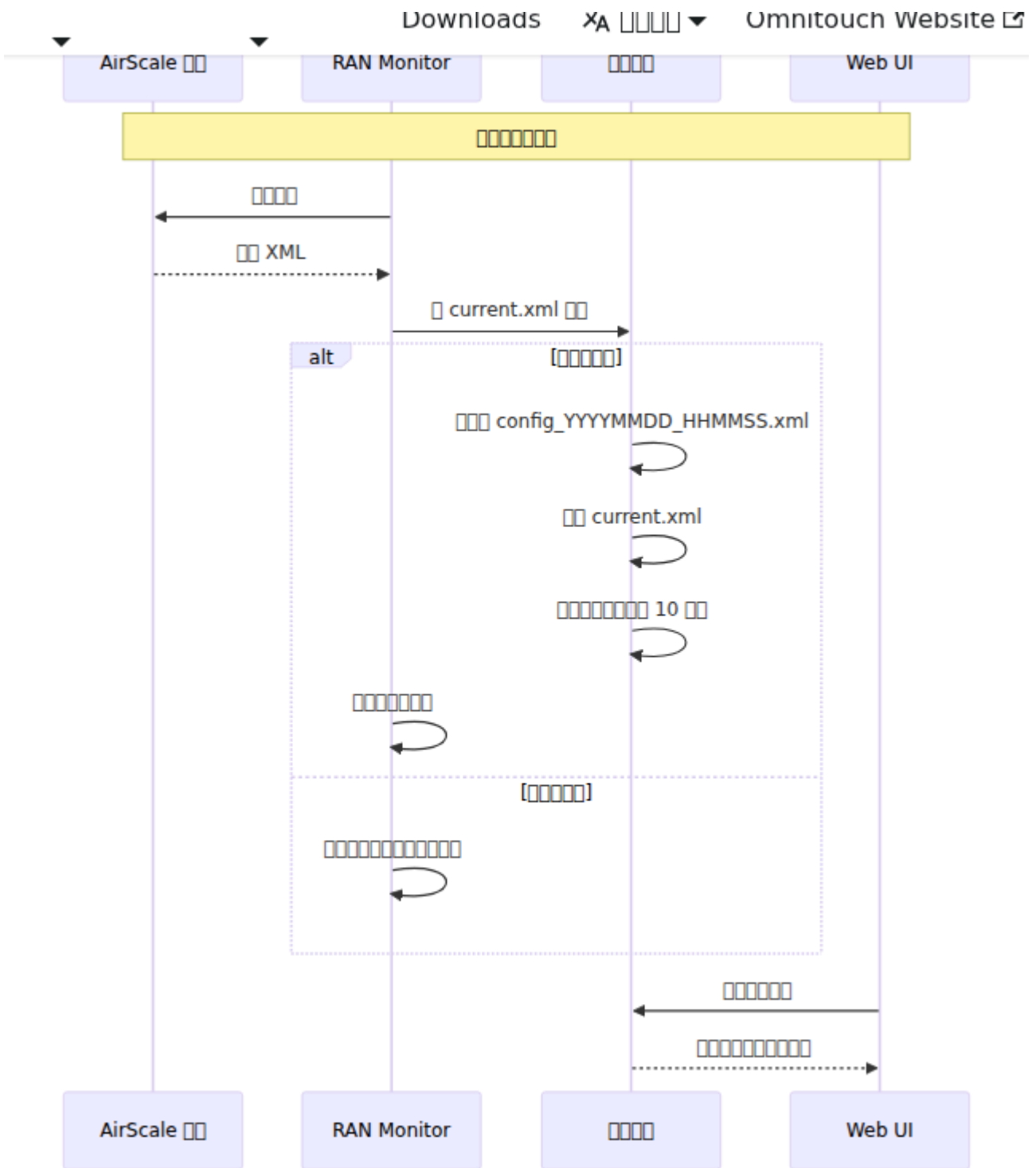
□□□□□□□□

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

□□□□□

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

□□□



API □□

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

□□□□

□□□□

```
curl -k "https://<server>:9443/download/config/<airscale-
name>/current.xml" \
-o current_config.xml
```

□□□□

```
curl -k "https://<server>:9443/download/config/<airscale-
name>/ONS-Lab-Airscale_config_20251230_143522.xml" \
-o ONS-Lab-Airscale_config_20251230_143522.xml
```

□□□ URL □□ AirScale □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□

□□□□□ **IEx** □□□□□

```
# 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** □□□□ - □□□□
- □□□□□□ - □□□□□□
- □□□□□□□□ - □□□□

📄📄📄📄📄📄📄

📄📄

RAN Monitor 📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄 InfluxDB 📄📄📄
📄📄 📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄

📄 📄📄📄📄

📄📄📄📄📄📄

1. 📄📄 📄📄📄: <https://localhost:9443>
2. 📄📄📄📄📄📄 📄📄📄
3. 📄📄📄📄📄📄📄 eNodeB 📄📄📄📄

📄📄📄📄📄📄

1. 📄📄📄📄 eNodeB
2. 📄📄“📄📄📄📄”📄📄📄📄📄📄📄
3. 📄📄📄📄📄
4. 📄📄📄📄📄📄📄📄📄📄

📄📄📄📄

1. 📄📄 📄📄📄📄 📄📄📄📄📄📄📄📄📄
2. 📄📄 📄📄📄📄📄 📄📄📄 eNodeB 📄📄📄📄📄📄📄📄

□□

□□□□□□□□□□ eNodeB □□□□□□□□□□

□ □□

□□□□□□

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

□□ **eNodeB** □□□

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

□□□□

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

□□□□□□

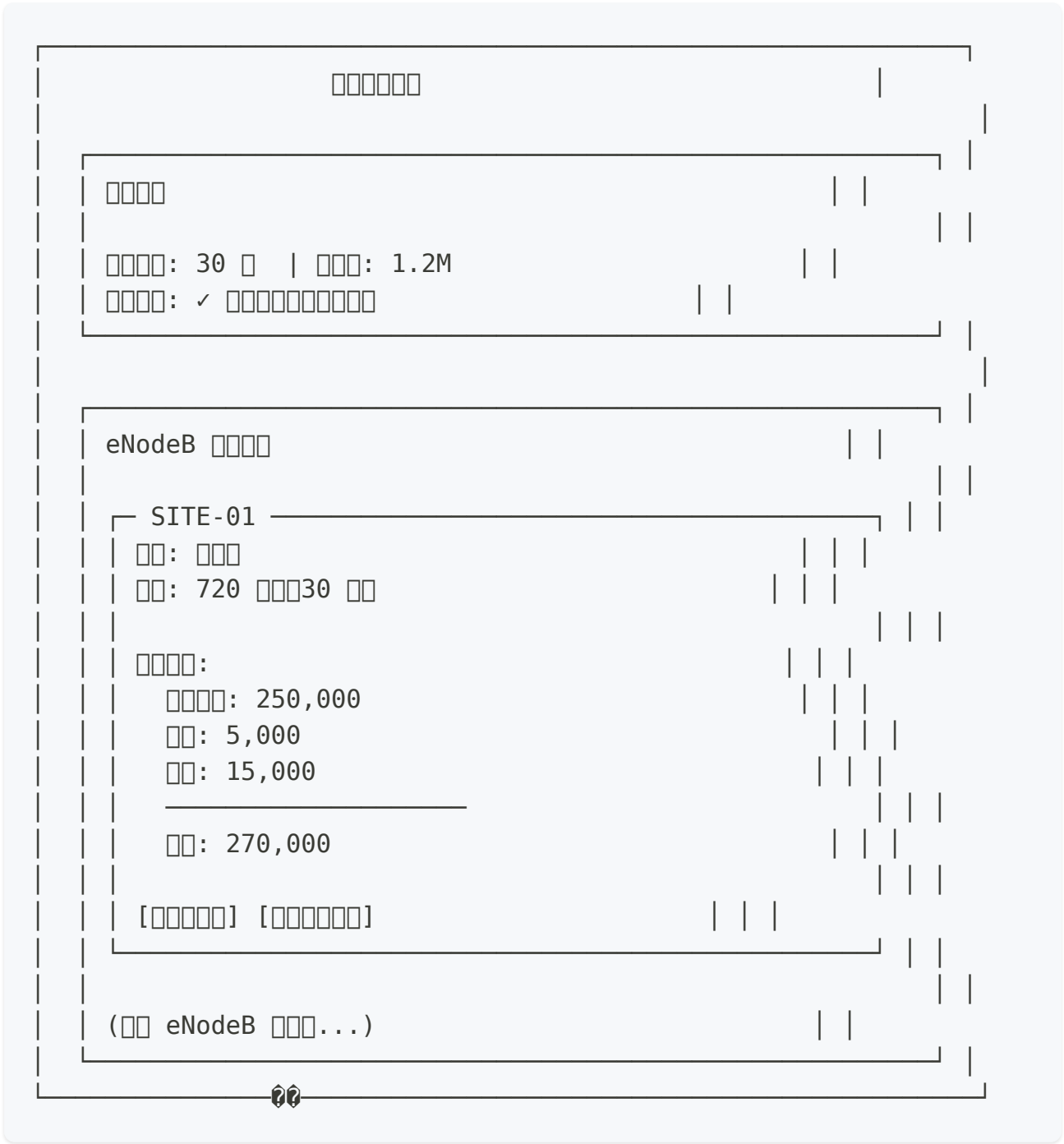
- □□□□□□: □□□□□□□□□□□□□□□□□□□
 - □□□□
 - □□
 - □□
 - □□□□: □□□□ eNodeB □□□□□□
 - □□: □□□□□□□□□□
-

□ □□□□

□□□□□

□□□□□□□□□□□□□□□□ *eNodeB* □□□□□□□□□□

□□□□□



000000

- 00 (✓): eNodeB 0000000000
- 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

= - (retention_hours * 3600)

30

└ : 2025-12-11 10:00:00

└ : 720 30

└ : 2025-11-11 10:00:00

└ : <

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. " " 5

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_CONFIG_PATH
- 环境变量: INFLUXDB_DATA_PATH

安装

1. 安装 **InfluxDB** 二进制文件
 2. 创建数据目录
 3. 配置 InfluxDB 配置文件
 4. 启动 InfluxDB 服务
 5. 验证 InfluxDB 安装成功
-

□ □□□□

□□□□□□□□

□□□

- □□□□□□□□□□□□□□
- □□□□□□□□

□□□□□

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

□□□□□□□□□□

□□□

- □□“□□□□□□”□□□□□□□□
- □□□□□□

□□□□□

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

□□□□□□□□□□

□□□

- □□□□□□
- □□□□□□

□□□□□

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** `enb`

`ui` `enb`

`ui`

`ui` `@cleanup_interval_ms`

`ui`

`ui`

Queries

Queries are used to retrieve data from InfluxDB.

Querying

Queries are executed using the following syntax:

- `SELECT <fields> FROM <measurement>`
- `SELECT <fields> FROM <measurement> WHERE <filter>`
- `SELECT <fields> FROM <measurement> ORDER BY <time>`

Querying Data

Queries can be executed using the `U` shell, `InfluxDB CLI`, or `API`.

Querying **InfluxDB** Data

Queries are used to retrieve data from InfluxDB.

Querying

Queries are executed using the following syntax:

□ □□□□

□□□□□

□□	□□
<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

□□□□

□□□ **AirScale** □□□□□

□□□□□□□□□□□□□□□□

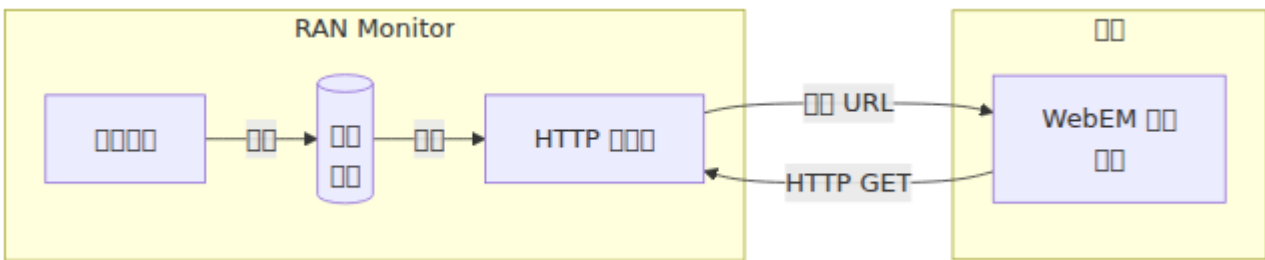
□□

□□□□□□□□□□□□□□□□ AirScale □□□□□□□□□□ RAN Monitor □□□□□□□ HTTP □□□□□□□ □□□□ WebEM □“□□□□□□”□□□□□□ RAN Monitor □□□□□□□□

□□□□

- □□□□□ - □□□□□ AirScale .zip □□□□
- **HTTP** □□□□ - □□□□□□□ HTTP URL □□□□□
- **MD5** □□□ - □□□□□□□□□□□□□□□□
- **URL** □□ - □□□□□ WebEM □□□□□□□ URL
- □□□□ - □□□□□□□□□□□□□□□□

□□□□□□



□□□□

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

root: ~ >

ls -la /var/firmware/nokia/

total 12

drwxr-xr-x 2 root root 4096 Nov 14 10:00

drwxr-xr-x 2 root root 4096 Nov 14 10:00 RAN Monitor

```
/var/firmware/nokia/
├── SBTS25R1_ENB_0000_001046_000000_release_BTSSM_downloadable_A54.zip
├── SBTS25R1_ENB_0000_001046_000000_release_BTSSM_downloadable_A54.zip
├── SBTS25R1_ENB_0000_001046_000000_release_BTSSM_downloadable_A54.zip.me
├── SBTS24R1_ENB_0000_001093_000000_release_BTSSM_downloadable_A52.zip
├── SBTS24R1_ENB_0000_001093_000000_release_BTSSM_downloadable_A52.zip
├── SBTS24R1_ENB_0000_001093_000000_release_BTSSM_downloadable_A52.zip.me
└── ...
```

ls -la /var/firmware/nokia/RAN Monitor/

- `.zip` -
- `.md5` - MD5
- `.metadata.json` -

ls -la /var/firmware/nokia/RAN Monitor/config/

`config/runtime.exs`

```
config :ran_monitor,
  firmware_storage_path: "/data/firmware/nokia"
```

項目	単位	単位	項目	備考
firmware_storage_path	文字列	文字列	/var/firmware/nokia	ファームウェアの保存先パス 指定する場合は、このパスに 指定してください

インストール

ファームウェアをインストールするには、`priv/firmware/nokia` に

Web UI

インストール

ファームウェアをインストールするには

項目	項目
ファームウェア	ファームウェア
パス	ファームウェア
ファームウェア	ファームウェア

インストール

ファームウェア AirScale インストール

1. ファームウェアを `.zip` に
2. ファームウェアをインストール
3. ファームウェアをインストール

ファームウェアをインストール

Download .zip file

15 GB

Details

- 15.37 MB/s
-
- /
- ETA

Download path

SBTS<release>_ENB_<build>_<variant>_release_BTSSM_downloadable_<suffix>

SBTS25R1_ENB_0000_001046_000000_release_BTSSM_downloadable_A54.zip

Download

Download path

□	□□
□□□	□□□□□□□□□□
□□	□□□□□□□□□□SBTS25R1□
□□□□	□□□□□□□□□□□□□□
□□	□□□□□□□□□□□□
□□	□□□□□□□□□□□□
MD5	MD5 □□□□□□□□□□
□□□□	□□□□□□□□□□
□□	□□ URL□□□□□□□□□□

□□□□□□

□□□□□□□□ □□ □□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

項目	説明
項目	項目名
項目	項目名 SBTS25R1
項目	項目名
項目	項目名
項目	項目名 RAN Monitor
項目	項目名 FTM_SBTS25R1_2025.05.27_0097
PS 項目	項目名
項目	Linux 項目名
LTE 項目	LTE 項目名 FL19A
WCDMA 項目	WCDMA 項目名 WBTSFP20C
項目	項目名 ASCE ASCF FCTJ
MD5 項目	項目名
項目 URL	項目名 URL

項目名


```
/var/firmware/nokia/  
├─ SBTS25R1_ENB_0000_001046_000000_release_BTSSM_downloadable_A54.zi  
└─  
SBTS25R1_ENB_0000_001046_000000_release_BTSSM_downloadable_A54.zip.mc
```

.md5

a1b2c3d4e5f6789012345678abcdef01

MD5

Linux/macOS:

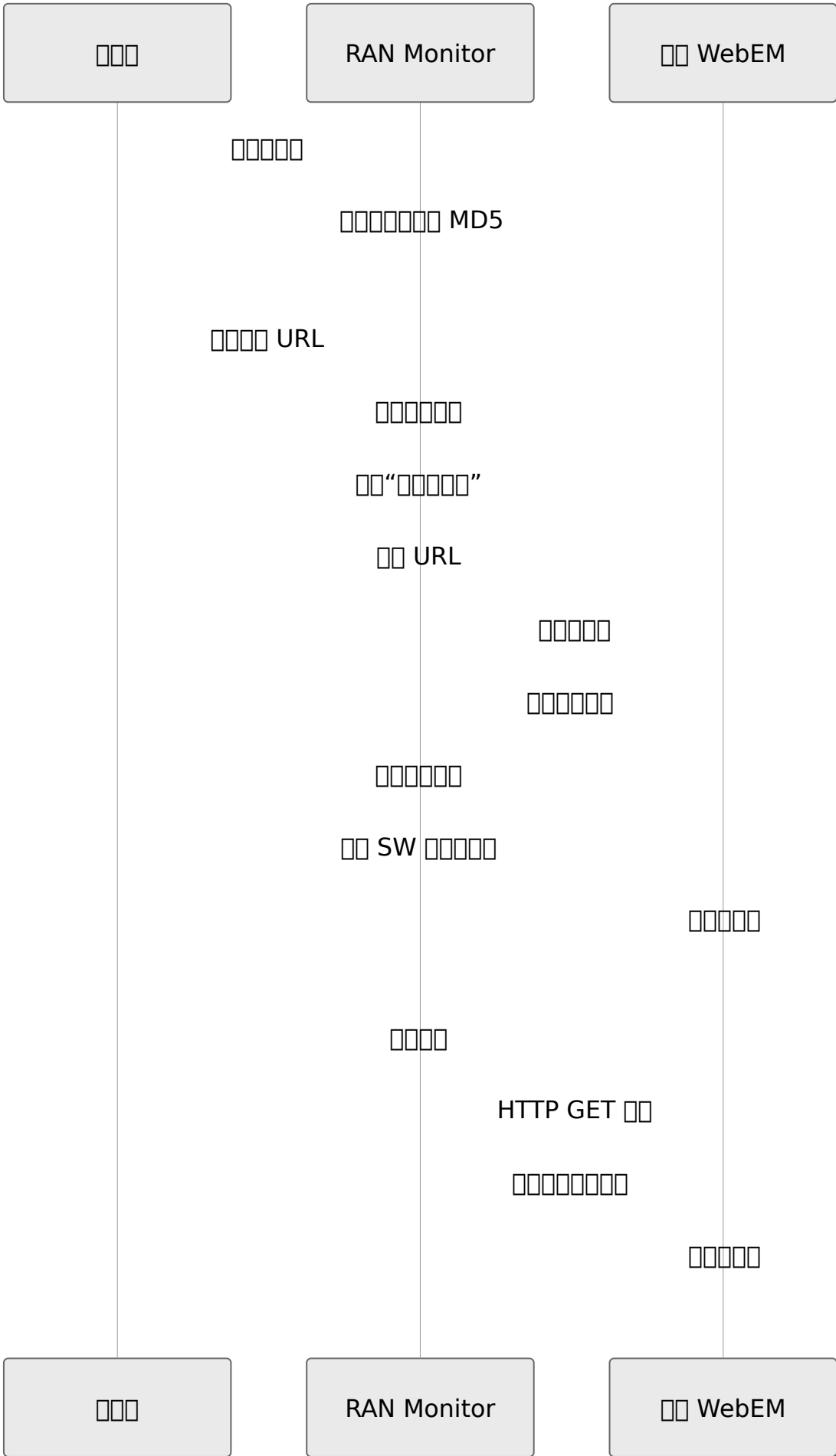
```
md5sum  
SBTS25R1_ENB_0000_001046_000000_release_BTSSM_downloadable_A54.zip
```

Windows PowerShell

```
Get-FileHash -Algorithm MD5  
SBTS25R1_ENB_0000_001046_000000_release_BTSSM_downloadable_A54.zip
```

- RAN Monitor
- RAN Monitor
- WebEM

□□□□





1. RAN Monitor

Accessing RAN Monitor

- Access `https://<ran-monitor-ip>:9443/nokia/firmware`
- Download `.zip` file
- Verify MD5

2. URL

Constructing the URL

```
http://<ran-monitor-ip>:4000/firmware/nokia/<filename>.zip
```

3. WebEM

Accessing WebEM

- Access IP > IP
- Access BTS

4. Configuration

WebEM Configuration

1. Access IP
2. Access URL `URL`
3. Access IP

WebEM Access RAN Monitor

5. Verification

1. Access IP
2. Access SW
3. Access IP

4. 4G LTE 網路效能

4G LTE

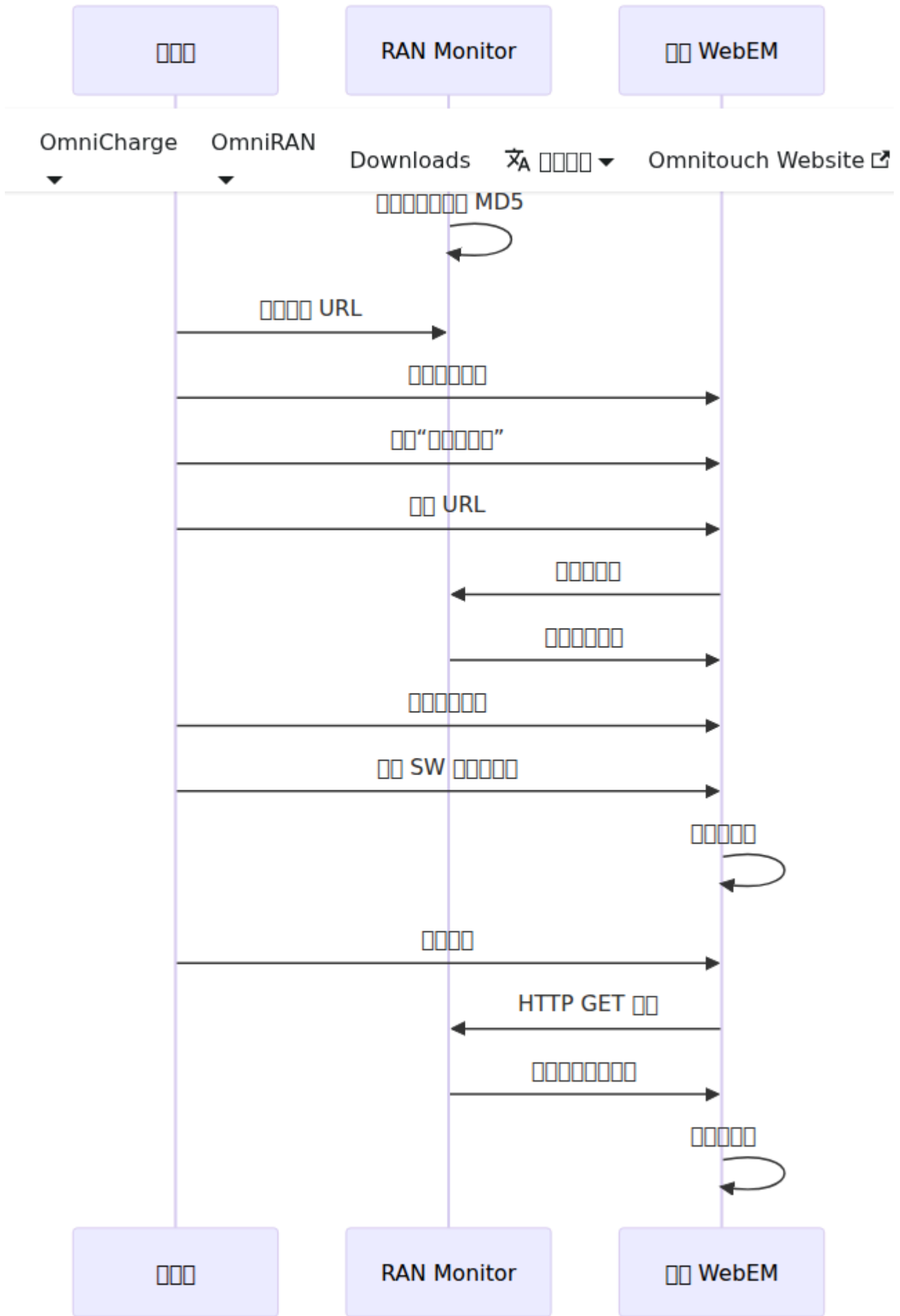
4G LTE

1. 4G LTE 網路效能 WebEM 效能 提升
2. 4G LTE RAN Monitor 效能提升
3. 4G LTE KPI 效能提升

4G LTE 網路效能 ? ? ? 提升

4G LTE RAN Monitor 效能提升 4G LTE WebEM

□□□□



□□□□

□□□□□□□□□□□□ XML □□□□□□□□□□□□□□□□□□ URL □□□□□

□□□ **sw_update.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<raml xmlns="raml21.xsd" version="2.1">
  <cmData type="plan" scope="changes">
    <managedObject class="com.nokia.srbts.btsswm:BTSSWM" distName="MRBTS-
    operation="update">
      <p name="swPkgUrl">http://{ran-monitor-
    ip}:4000/firmware/nokia/SBTS25R1_ENB_0000_001046_000000_release_BTSSM
      <p name="swPkgActivate">true</p>
    </managedObject>
  </cmData>
</raml>
```

□□□

- **{site_id}** - □□□□□ MRBTS ID□□□□□ 256□ 2057□
- **{ran-monitor-ip}** - □□□□□□□ RAN Monitor □ IP □□❓❓□□□

□□□□

1. □□□ RAN Monitor □□□□□□□□
2. □□ □□□□ □□□□□□□□
3. □□ **sw_update.xml** □□
4. □□ □□ □□□□□
5. □□ □□ □□□□□

□□□□□□□□□□□□□□

□□□□

□□□□□□□□□□□□□□□□□□ XML □□□□□□□□ MRBTS ID□□□□□ RAN Monitor □□□□□□□□□□ API □□□□□□□□

API

GET /api/firmware/nokia

REST API

GET /api/firmware/nokia

JSON

```
{
  "firmware": [
    {
      "filename": "SBTS25R1_ENB_0000_001046_000000_release_BTSSM_dowr",
      "size": 1073741824,
      "size_formatted": "1.0 GB",
      "modified": "2025-01-15T10:30:00Z",
      "download_url": "http://ran-monitor:4000/firmware/nokia/SBTS25R1_ENB_0000_001046_000000_release_E",
      "metadata": {
        "release": "SBTS25R1",
        "build": "0000_001046_000000",
        "variant": "release",
        "suffix": "A54",
        "type": "BTS"
      }
    }
  ],
  "stats": {
    "file_count": 3,
    "total_size": 3221225472,
    "total_size_formatted": "3.0 GB",
    "storage_path": "/var/firmware/nokia"
  }
}
```

GET /firmware/nokia/<filename>

GET /firmware/nokia/<filename>

Content-Type: application/zip

□□□□

URL

URL

```
config :ran_monitor,  
  firmware_base_url: "http://ran-monitor.example.com:4000"
```

□□	□ □	□ □	□□	□□
firmware_base_url	□ □ □	□	http://localhost:4000	□□□□ URL □□□□ □ URL□□□□□□ RAN Monitor □□□ □□□□□□□□/IP□

Web UI URL □ API URL□□□□□□□□□□□□□□□□ HTTP □□

□□□□

□□□□

□□□□□□□□□□□□□□

□□□□

- 15 GB □□□□
- .zip □□
-

- 0000000000

000000

1. 00000000 15 GB 00
2. 000000000 .zip
3. 0000000000 df -h /var/firmware/nokia
4. 000000000 ls -la /var/firmware

00000000

000 WebEM “000000”00000000

000000

- 000 RAN Monitor 0000000000
- 000000 HTTP 00 4000
- URL 00000000 IP 000000

000000

1. 0000000000000000 ping RAN Monitor
2. 0000000000 4000 00
3. 00 URL 000000 IP/0000000000
4. 0000000000000000 URL

MD5 00“000...”

000 MD5 000“000...”000000

000000

- 0000000000000000
- 0000000000000000 .md5 000000

000000

1. 0000000000000000000000

2. 2000 年 10 月 1 日

3. 2000 年 10 月 1 日

2000 年 10 月 1 日

2000 年 10 月 1 日

2000 年

1. 2000 年 10 月 1 日

2. 2000 年 10 月 1 日

3. 2000 年 10 月 1 日

📊 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", # 中国移动
    mnc: "001" # 中国移动
  },
  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 配置 SSL 证书

配置 HTTPS Web UI 需要 SSL 证书

```

# 生成证书/私钥
openssl req -newkey rsa:2048 -nodes -keyout
priv/cert/omnitouch.pem \
  -x509 -days 365 -out priv/cert/omnitouch.crt

```

生成 CA 证书

生成证书私钥

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
```

完了

- Web 管理
 - 監視
 - InfluxDB 管理
 - 設定
-

目次

第 4 章 監視

監視

4.1 Web UI 管理

監視

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

監視 RAN Monitor 管理

4.2 監視

Web UI 管理

1. 監視 管理
2. 監視
3. 監視“監視”管理
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. 000 0000
2. “”
3. 00000000

mix phx.server

4.5

Web UI

1. 000000000000
2. 000
 - o 0000000000

- 配置参数
 - 部署
 - 验证
-

部署

部署 RAN Monitor 部署

部署

1. 部署

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

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 `config/runtime.exs`

4. 部署

- 部署 Web UI 部署
- 部署 Grafana 部署
- 部署
- 部署 `config/runtime.exs`

□□□□

□□□□

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

□□□□

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

□□□□

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

□□□□

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

□□□□

□□□□

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

□□□□

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

□□□□

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

□□□□

□□□□□□

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

□□

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

□□□□□□

□□□□□□

- □□□□□□□□ping□
- □□□□□□□□

- 8080 8086
- 8086

InfluxDB 8086

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

Web UI 8086

- HTTPS 9443
- SSL
- Web
- 8086

8086

- 8086 - 8086
- **Web UI** 8086 - 8086
- 8086 - 8086
- 8086 - 8086
- **AirScale** 8086 - 8086
- **Grafana** 8086 - 8086
- 8086 - 8086
- 8086 - 8086
- 8086 - 8086

Grafana □□□□□□□□

□□□□ **RAN** □□□□□□□□□□

Grafana □□□□□□□□ KPI □□□□□□□□

□□

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

□□

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

概要

このドキュメントは、RAN Monitor/NOG (ONS) のパフォーマンス/KPI を可視化する Grafana ダッシュボードについて説明します。

前提条件

- RAN Monitor/NOG (ONS) が正常に稼働していること
- RAN Monitor/NOG (ONS) のパフォーマンス/KPI データが InfluxDB に保存されていること
- Grafana がインストールされていること
- RAN Monitor/NOG (ONS) の Nokia データソースが設定されていること

ONS データソース

- RAN Monitor/NOG (ONS) のパフォーマンス/KPI データソースを InfluxDB に接続します。
- データソースの名前を ONS と設定します。
- データソースの URL を `http://localhost:8080` と設定します。
- データソースの認証情報を設定します。
- データソースの認証情報を InfluxDB に保存します。

このドキュメントは、RAN Monitor/NOG (ONS) のパフォーマンス/KPI データを可視化するための Grafana ダッシュボードについて説明します。

Grafana と InfluxDB のインストール

インストール

インストール

- InfluxDB 2.0+ を RAN Monitor/NOG (ONS) にインストールします。
- InfluxDB API をインストールします。
- Grafana を InfluxDB にインストールします。

Docker Compose を使用する

```

version: '3.8'
services:
  influxdb:
    image: influxdb:2.7
    environment:
      INFLUXDB_DB: ran_metrics
      INFLUXDB_ADMIN_USER: admin
      INFLUXDB_ADMIN_PASSWORD: change_me
    ports:
      - "8086:8086"
    volumes:
      - influxdb_data:/var/lib/influxdb2

  grafana:
    image: grafana/grafana:latest
    environment:
      GF_SECURITY_ADMIN_PASSWORD: change_me
    ports:
      - "3000:3000"
    depends_on:
      - influxdb
    volumes:
      - grafana_data:/var/lib/grafana
      - ./provisioning:/etc/grafana/provisioning

volumes:
  influxdb_data:
  grafana_data:

```

📦 InfluxDB API 📦

1. 📦 InfluxDB UI 📦 8086
 2. 📦 API 📦
 3. 📦 📦 📦 📦 📦 📦 📦 📦 📦 📦 📦 📦
 - 📦 📦 📦 📦 ran_metrics 📦 📦 📦 📦 📦
 4. 📦 📦 📦 📦
 5. 📦 Grafana 📦 📦 📦 📦 📦 📦 📦
-

□□□□□

□ 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 □□)

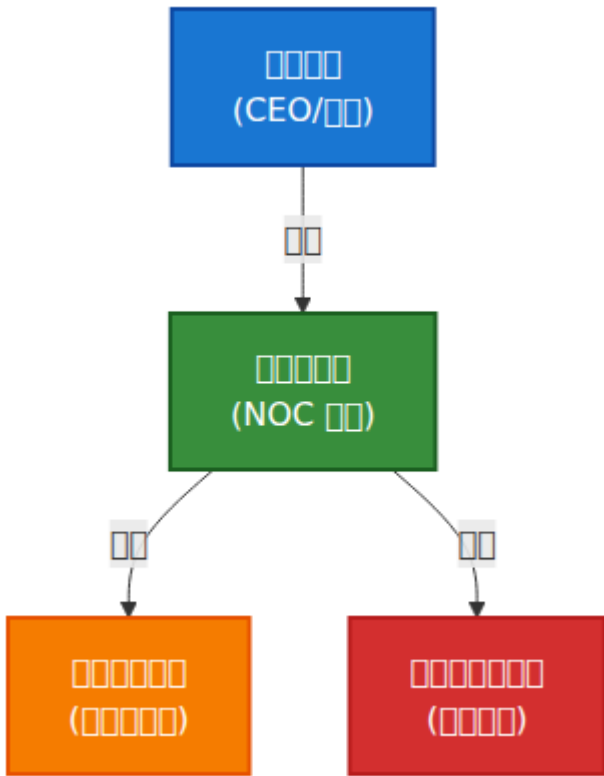
□□

□□

□□□□□□
(□□□□□)

□□□□□□□
(□□□□)

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 [2〇〇〇] 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

```

| 000007 00 |
| [0000000000/0000] |
| 000250 Gbps000000 2 00 |
| 00080 Gbps000000 3 00 |

```

000

- 000000
- 000000
- 000000
- 0000

🔗🔗0000000000000000

0000000

```

| 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

- 000000000000
- 000000000000

000000000000

```
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")
```

000

- 00000000
- 000 80% 0000000095% 00000000
- 000000000000

00000

000000000000000000 **24** 000

```
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")
```

000

- 00000000000000
- 00000000
- 00000000000000

00000000000000

```
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")

```

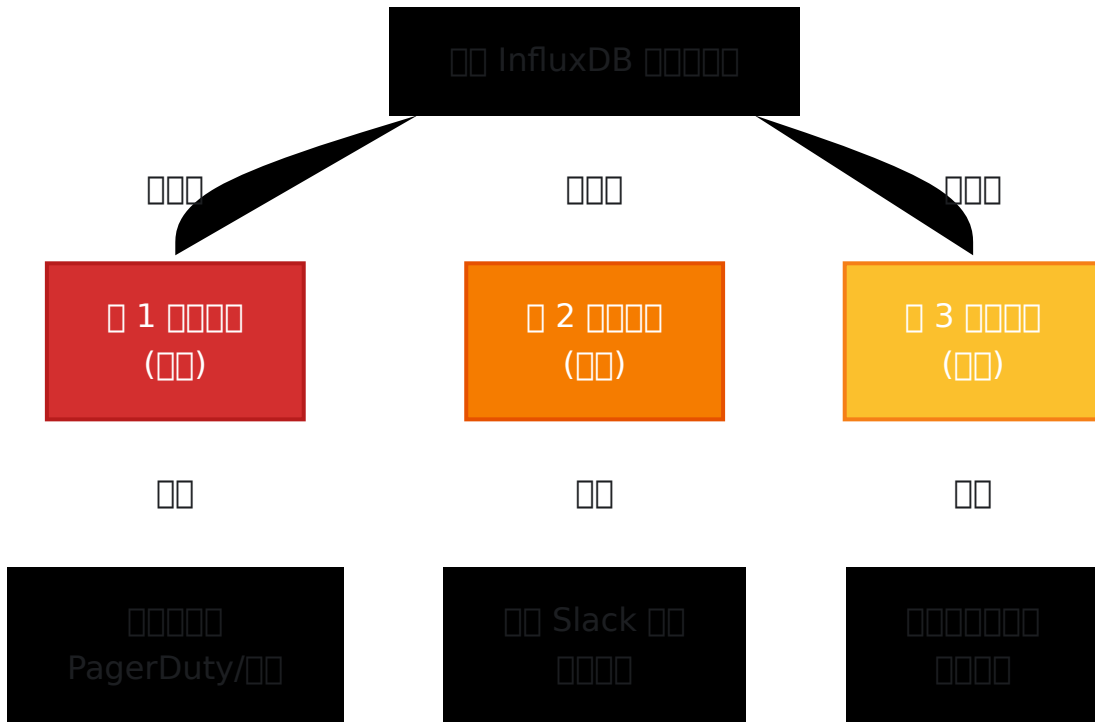
□□□

- □□□□□□□□□□□□
- □□□□□□□□

- `influxdb`

`influxdb`

`influxdb`



Grafana

1

- 1.
- 2.
3. →
4. → →

2

1

```
CellAvailability < 95%
```

```
duration 15 min
```

```
severity 1 min
```

```
cooldown 15 min
```

•

- 95% CellAvailability SLA 99.5%
- 15 min duration
- severity 1 min

• **2**

```
count(active_alarms) > 10
```

```
duration 5 min
```

```
severity 2 min
```

```
cooldown 5 min
```

•

- 10 active_alarms
- 5 min duration
- severity 2 min

• **3**

```
DLResourceUtilization > 90%
```

```
duration 30 min
```

```
severity 5 min
```

```
cooldown 30 min
```

•

- DLResourceUtilization
- 30 min duration
- 5 min severity

3. 3. 3. 3. 3. 3. 3. 3. 3. 3.

1. “ ”
2. Slack PagerDuty
- 3.

```

{{ .AlertRuleName }}
{{ .Severity }}
{{ .Labels.device }}
{{ .EvalMatches[0].Value }}
{{ .StartsAt }}

{{ .RuleUrl }}

```


1.

- SLA
- 1-5
- Slack + PagerDuty
-
- **SLA < 15**

2.

-
- 15-30
- Slack + PagerDuty
- NOC +
- **SLA < 30**

3.

-
- 1+

- 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 "Grafana" チャンネル
5. Grafana URL
6. Slack

Slack 連携内容

Grafana 連携 - SITE_A_BS1_Cell1
 連携チャンネル名
 連携時間 ~2000 分
 連携時間 2:15
 [チャンネル名] [チャンネル名] [チャンネル名]

PagerDuty 連携

1. Grafana PagerDuty 連携
2. Grafana 設定 → PagerDuty
3. Grafana "PagerDuty" チャンネル
4. Grafana URL
5. Grafana URL
 - Grafana → Grafana URL
 - Grafana → Grafana URL
 - Grafana → Grafana URL

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

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** 0000000000

000000000000

```
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"

// 1 PRB 100
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8011C24")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> map(fn: (r) => ({ r with _value: r._value / 10.0}))
  |> rename(columns: {"_value": "1 PRB 100"})

// 1000 PRB 100
from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["recordType"] == "performanceMetric")
  |> filter(fn: (r) => r["basebandName"] == "${Airscale}")
  |> filter(fn: (r) => r["metricCounter"] == "M8011C37")
  |> filter(fn: (r) => r._field == "counterValue")
  |> filter(fn: (r) => strings.containsStr(v: r["DN"], substr:
"${CellKey}"))
  |> group()
  |> map(fn: (r) => ({ r with _value: r._value / 10.0 }))
  |> rename(columns: {"_value": "1000 PRB 100"})

```

1 6

PDCP

```

import "strings"

// PDCP 7000000
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 7000000"})

// PDCP 7000000
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 7000000"})

```

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

-

Nokia Monitor $\square\square\square\square\square\square$ S1 $\square\square\square\square\square\square\square\square\square\square\square\square\square\square\square\square$ PRB $\square\square\square\square\square\square\square\square\square\square\square\square\square$

$\square\square\square\square$ - $\square\square\square\square$ **PRB** $\square\square\square\square$

$\square\square\square\square\square\square\square\square$ LNCel $\square\square$ LTE PRB $\square\square\square$ TTI $\square\square\square/\square\square\square\square$ PDCP $\square\square\square\square\square\square\square\square\square\square\square\square$

$\square\square\square\square$ - **RSSI** $\square\square\square\square\square\square$ **RRC**

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

000000

000

- 000000000
- 000000
- 000000

000

1. 設定する
2. 設定する
3. 設定する
4. 設定する

設定

- 設定する → 設定
- 設定する → 設定
- 設定する URL/設定
- Grafana 設定
- 設定

設定

設定

- 設定
- 設定
- 設定

設定

1. 設定
2. 設定
3. 設定/設定
4. 設定

設定

- InfluxDB 設定
- aggregateWindow 設定/設定/設定
- InfluxDB 設定
- 設定

□□□□□□

□□□

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

□□□□□

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

□□□□□

□□□

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

□□□□□

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

□□□□□

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

□□□□□□

□□ RAN □□□□□□

□□□□□□□□□□□□□□□□

□□

1. □□
 2. □□□□□
 3. □□□□
 4. □□□□□□
 5. □□□□□□
 6. □□□□□□
 7. □□□□□□
 8. □□□□
 9. □□□□□
 10. □□□□
-

□□

□□□□□□□□□□□□ RAN □□□□□□□□□□□□□□ NOC □□□□□□□□□□□□□□□□

□□□□

- RAN □□□□□□□□
- □□□□□□□□□□
- □□□□□ RAN □□□□□□
- □□□□□□□□□□

□□□□□□□□□□ □□□□□□

Task 1

Configure Nokia AirScale to connect to a base station

Step 1: Configuration

Configure the base station IP

```
# Configure the base station IP
ping <base-station-ip>

# Configure the telnet port
telnet <base-station-ip> 8080
```

Verify the configuration using ping and telnet

Steps:

- Configure the base station IP
- Configure the telnet port
- Verify the configuration

Step 2: Verification

Verify the configuration

項目	値	説明
IP アドレス	10.7.15.67	管理用 IP アドレス
ポート	8080	管理用ポート (Nokia AirScale) 8080
ホスト名	Site-B-Tower-1	管理用ホスト名
ユーザー名	admin	管理用ユーザー名
パスワード	password123	管理用パスワード

管理用メニュー

- 管理用メニュー
- 管理用メニュー
- 管理用メニュー
- 管理用メニュー: NYC-SiteA-BS1, LAX-Tower-Main, CHI-Indoor-DAS

ステップ 3: 管理用メニュー

管理用メニュー

1. 管理用 Web UI: `https://<ran-monitor-ip>:9443`
2. 管理用メニュー **eNodeBs** をクリック
3. 管理用メニュー IP アドレス ID
4. 管理用メニュー ID

管理用メニュー RAN 管理

ステップ 4: 管理用メニュー

管理用 `config/runtime.exe` を実行して `airscales` を確認

```

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 []
      }
    ]
  }
}

```

[] [] Elixir [] - []

[] **5** []

[]

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

[] []

[]

- []

- 0000000000 { 0000 [0000
- 0000000000000000
- 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] 0000000000Site-B-Tower-1
[info] 0000 Site-B-Tower-1
```

2. 00 Web UI

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

3. 0000000000

- 0000

- 設定する
- 設定 "設定" を実行する

4. InfluxDB のインストール

- InfluxDB のインストール
- 設定
- 接続

8. Grafana のインストール

インストール手順

1. 設定
2. インストール
3. 設定 "設定" を実行する
4. 接続

設定    Grafana のインストール

9. Grafana のインストール

Grafana のインストール

1. 設定
2. インストール
3. 設定

インストール手順 [Grafana のインストール](#)

インストール

インストール手順

00 1 000000000000

00000000000000000000

00 A 000000

- 000000000000
- 0000000000000000

00 B 000000

- 000000000000
- 00 InfluxDB 00
- 000 - 000000

00 2 000000000000

00000000000000

00 config/runtime.exe 00 airscales 000000000000000000

```
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"  
  # }  
]
```

Step 3: Configure the RAN

Configure the RAN

1. Access the Web UI
2. Configure the RAN
3. Save the configuration
4. Restart the RAN

Restart the RAN

Step 4: Restart RAN

Restart the RAN

```
systemctl restart ran_monitor  
#  
mix phx.server
```

Step 5: Verify

Verify

1. Verify the RAN status
 - o Check the RAN status
 - o Check the RAN logs
2. Verify the RAN configuration
 - o Check the RAN configuration
 - o Check the RAN logs
3. Verify InfluxDB
 - o Check the InfluxDB status
 - o Check the InfluxDB logs

06 Grafana 0000

0 Grafana 00000000

1. 0000000000
2. 00000000000000
3. 0000000000000000

00000000

000000000000 RAN 000000

00 1000000000

0000000000

1. 000000000000000000
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. Check logs

```
[info] Site-A-BS1  
[info] Site-A-BS1
```

2. Web UI

- Check "Site-A-BS1"
- Check
- "Site-A-BS1" status

Check

- Check
- Check
- Check
- Check

Conclusion

RAN configuration

□□□□□□

RAN □□□□□□□□□□□□□□□□

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

□□□□□□

□□□□□□□□□□

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

□□□□□□□□□□□□

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

□□ **1**□□□□□□

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

□□□□□□□□□□□□□□

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

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

OmniTouch

2

???

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

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

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

3

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

- 0000000000
4. 0000000000
- 0000000000
 - 0000000000
-

000000

0000 RAN 0000000000

000000

0000000000000000 Web UI 00 - 000000

000000

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

000000

000000

- 0000000000
- 0000000000
- 0000000000

000000

- 00000000000000
- 000000000000
- 00000000

部署前準備

- 環境構築
- ネットワーク
- 権限設定

インストール

- 依存関係の確認
- 実行 ID の取得
- 実行コマンド
- 実行結果の確認

確認

- ログの確認
- サービスの状態
- 実行時間 15-30 分
- エラーメッセージ

トラブルシューティング

よくある RAN 関連の問題

よくある問題

よくあるエラーメッセージ

1. Web UI へのアクセス

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

2. ログの確認

- ログファイルの場所

- 安装 Prometheus
- 配置 Prometheus

3. 安装 Grafana

- 安装 Grafana
- 配置 Grafana
- 启动 Grafana

4. 配置 Prometheus

- 配置 InfluxDB 数据库
- 配置 Prometheus 数据源
- 配置 Prometheus 告警

5. 配置 Grafana

- 配置 Grafana 数据源
- 配置 Grafana 仪表盘
- 配置 Grafana 告警

安装 Prometheus 和 Grafana 的 Web UI 界面 - Web UI 安装指南

安装 Grafana 仪表盘

LMNME S1 UEs PRB

□□□□□□

□□□□□□□□

1. □□□□□

- Grafana □□□□□□□□
- □□□□□□□□□□
- □□□□□□□□

2. □□□□□

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



3. □□□□□□

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

4. □□□□□□□

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

5. □□□□□

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

□□□□□ 30-45 □□

□□□□

□□□□□□

□□□□□□□□□□ □□□□□□□□

□□□□□

□□□□□□

- □□□□□□□□
- □□□□□□□□□□

□□□□□□

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

□□□□□□

- □□□□□□ "□□□□□" □□
- □□□□□□□□

□□□□□

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

□□□□

□□□□□□

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

4. 导出数据

使用 InfluxDB

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

使用 Grafana

- 配置
- 数据源
- 创建 "面板" - "数据源" - "CSV"

备份配置

配置

脚本

1. 备份

```
# 备份配置
cp config/runtime.exe backups/runtime.exe.$(date +%Y%m%d)

# 备份配置
tar -czf backups/config-$(date +%Y%m%d).tar.gz config/
```

2. 部署

- 部署 Web UI
- 配置
- 数据源/面板

3. MySQL 备份

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

4. InfluxDB 备份

```
# InfluxDB 1.x 备份  
influxd backup -portable -database nokia-monitor  
/backups/influx-$(date +%Y%m%d)  
  
# InfluxDB 2.x 备份  
influx backup /backups/influx-$(date +%Y%m%d)
```

5. SSL 证书

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

备份策略

备份

- MySQL 备份策略
- 备份策略

备份

- InfluxDB 备份策略
- 备份策略

备份策略

- 备份策略
- 备份策略
- 备份策略

□□□□

□□□□□□□□

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
- 000000000000
- 0000000000

4. 0000

- 00000000
- 0000000000
- 000000000000

0000

1. 000000

- 00000000
- 00000000
- 00000000

2. 0000

- 000000
- 000000
- 0000000000 SSL 00

3. 0000

- 00000000
- 0000000000
- 000000000000

000

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. 00000000

- 00000000
- 00000000
- 00000000

3. 00000000

- 00000000
- 00000000
- 0000000000000000

4. 000000

- 0000000000
- 00000000

- 資料庫

5. 資料庫

- 資料庫
- 資料庫
- 資料庫
- 資料庫

6. 資料庫

- 資料庫
- 資料庫
- 資料庫

資料庫

- 資料庫 - 資料庫
- **Web UI** 資料庫 - 資料庫
- 資料庫 - 資料庫
- **AirScale** 資料庫 - 資料庫
- 資料庫 - 資料庫
- **Grafana** 資料庫 - 資料庫
- 資料庫 - 資料庫
- 資料庫 - 資料庫
- 資料庫 - 資料庫
- 資料庫 - 資料庫

LTE 网络部署

部署 AirScale/FlexiRadio 网络

目录

1. 简介
 2. 网络架构
 3. 部署环境
 4. 网络配置
 5. 性能测试
 6. UE 部署
 7. 网络优化
 8. 故障排除
 9. 维护
 10. 网络 QoS 配置
 11. 网络监控
 12. Grafana 部署
 13. 附录
-

简介

本文档描述了 LTE 网络部署 AirScale/FlexiRadio 网络的详细步骤和配置。该网络旨在提供高速、低延迟的无线通信服务。

部署环境

部署环境包括以下组件： 部署环境

- 部署环境
- 部署环境

□□□□□□ (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	測定対象
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	測定対象
M5112	IP 電圧	112	電圧変動率
M5113	電圧 RX	21	電圧変動率

電圧 (M4xxxx)

項目	単位	測定時間	測定対象
M4001	電圧	電圧	VSWR電圧変動率RF 電圧
M4002	電圧	電圧	電圧

電圧変動率

電圧変動率

- M8013Cxx - □□□□

□□□□□

- M8020Cxx - □□□□
- M8049Cxx - □□□□

□□□□□

- M8005Cxx - □□□□□□
- M8001Cxx - □□□□
- M8013Cxx - □□□□

□□□□□□□□

M8011 - □□□□□□

□□□□□□□□□□PRB□□□□□□□□□□□□□□□□□□

□□□	□□	□□	□□	□ □	□□
M8011C24	□□ PRB □□□□ TTI □□	□□□□□□□□□□□□□□ PRB □□	0.1%	□ □	□□ 10 □□ □□□
M8011C37	□□ PRB □□□□ TTI □□	□□□□□□□□□□□□□□ PRB □□	0.1%	□ □	□□ 10 □□ □□□

□□ **PRB** □□□□

- □□□□□□**PRB** □ LTE □□□□□□□□□□□□□□
- **TTI**□□□□□□□□□□ = LTE □□ 1 □□
- □□□□□□□□ = □□□□□□□□□□□□□□
- 100% □□□□□□□□□□□□□□

□□□□

- 0-1000□□□ 0.0% □ 100.0%□
- □□□□□□□ 10 □□□□□
- □□□□□□□ 453 = 45.3% PRB □□□

□□□□□□□□□□

- < 50% - □□□□□□□□
- 50-70% - □□□□□□□
- 70-85% - □□□□□□□□□
- > 85% - □□□□□□□□□□□/□□

Grafana □□□□□

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

□□□□□□□□

M8012 - □□□□□□□□

□□□□□□□ PDCP□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

UE	UE	UE	UE	UE	UE
M8012C23	PDCP UE UE	UE PDCP UE	kbit/s	UE UE	UE UE UE PDCP SDU UE
M8012C26	PDCP UE UE	UE PDCP UE	kbit/s	UE UE	UE UE UE PDCP SDU UE

UE PDCP UE

- PDCP UE
- UE kbit/s UE
- UE
- UE

UE

- 1 UE
- UE
- UE VoLTE UE

UE

UE M8012C26 UE

- < 10 Mbps - UE/UE
- 10-50 Mbps - UE
- 50-100 Mbps - UE/UE
- > 100 Mbps - UE

UE M8012C23 UE

- < 5 Mbps - UE
- 5-20 Mbps - UE
- 20-40 Mbps - UE
- > 40 Mbps - UE

Grafana Queries

```

from(bucket: "nokia-monitor")
  |> range(start: v.timeRangeStart, stop: v.timeRangeStop)
  |> filter(fn: (r) => r["metricCounter"] == "M8012C26")
  |> filter(fn: (r) => r._field == "counterValue")
  |> map(fn: (r) => ({ r with _value: r._value / 1000.0})) // Mbps

```

UE Queries

M8018 - eNB Queries

Queries for eNB M8018 UE

<small>UE</small>	<small>eNB</small>	<small>UE</small>	<small>1</small>	<small>1</small>	<small>1</small>
M8018C1	<small>eNB UE</small>	<small>eNodeB UE</small>	<small>1</small>	<small>1</small>	<small>1</small>

UE

- UE** = SRB DRB
- 1
-

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

-
- AirScale 150-250 UE
- UE

M8020 -

M8020C3					~10
M8020C4					~10
M8020C6					~10

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

指標

- **M8020C3** - 平均回應時間
- **M8020C6** - 失敗率
- **M8020C4** - 平均延遲時間

SLA

SLA	等級	說明
> 99.9%	AAA	最高品質 SLA
99.0-99.9%	AA	高品質
95.0-99.0%	A	標準品質
< 95.0%	B	最低品質 - 不建議使用

Grafana 查詢

```
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,
    "SLA": 100.0 * r.M8020C3 / (r.M8020C6 - r.M8020C4)
  })))
```

□□□□□□□□

M8005 - □□□□□□

□□□□□□□□□□□□□□□□RSSI□□□□□□□□□□□□□□□□SINR□□□□□□□□□□□□□□

RSSI □□

□□□	□□	□□	□□	□□
M8005C0	PUCCH □□ RSSI	□□□□□□□□□□□□□□□□ RSSI	dBm	□□
M8005C1	PUCCH □□ RSSI	□□□□□□□□□□□□□□□□ RSSI	dBm	□□
M8005C2	PUCCH □□ RSSI	□□□□□□□□□□□□□□□□ RSSI	dBm	□□

□□ RSSI□

- **RSSI** = □□□□□□□□□□□□□□□□
- **PUCCH** = □□□□□□□□□□□□□□□□
- □ dBm□□□□□□□□□□□□
- □□□□ UE □□□□ RSSI □□□□

RSSI □□□□

RSSI □□	□□	□□
> -70 dBm	□□	□□□□□
-70 □ -85 dBm	□□	□□□□□□□□
-85 □ -100 dBm	□□	□□□□
-100 □ -110 dBm	□□	□□□□□□□□□□
< -110 dBm	□□□	□□□□□□□□□□□□

□□□

- □□□□ - □ RSSI □□□□□□
 - □□□□□□ - □□□ RSSI □□
 - **RF** □□ - □□□□□□□□
-

□□□□□□□□

M8013 - □□□□□□

□□□□□□ RRC□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□	□□	□□	□□	□□
M8013C5	□□□□□□□□	□□□ RRC □□□□	□□	□□
M8013C17	□□□□□□□□ MO-S	□□□□ - □□□□□□	□□	□□
M8013C18	□□□□□□□□ MT	□□□□ - □□□□	□□	□□
M8013C19	□□□□□□□□ MO-D	□□□□ - □□□□□□	□□	□□
M8013C21	□□□□□□□□□□	□□□□□□□□	□□	□□
M8013C31	□□□□□□□□□□□□	□□□□□□□□	□□	□□
M8013C34	□□□□□□□□□□□□	□□□□□□□□	□□	□□
M8013C91	□□□□□□□□ MO-V	□□□□ - □□□□□□	□□	□□
M8013C93	□□□□□□□□ MT-□□	□□□□ - MT □□	□□	□□

□□□□□□□□

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

□□□□

- **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 接口

网络接口 eNodeB 与 MME 之间的 S1 接口

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

S1 接口

- S1 接口 eNodeB 与 EPC 接口
- 接口名称 接口 UE 接口
- S1 接口 eNodeB 与 MME 接口
- NAS 接口 接口

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

000000000000—UE 000000000000000000000000

00000

00	00	000
00 eNB	000000000000	M8015
eNB 00	000000000000X20	M8014
000	000000000000	M8021
000	0000000000LTE→3G/2G0	M8017

00000

Scenario	Scenario	Scenario
M8009	Scenario	Scenario
M8014	Scenario X2 HO	Scenario
M8015	Scenario HO	Scenario
M8021	Scenario HO	Scenario

Scenario

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

Scenario

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

Scenario

- Scenario
- Scenario
- Scenario UE Scenario
- Scenario

Scenario

M8010 - CQI Scenario

Scenario UE Scenario CQI Scenario

Scenario CQI

- **CQI** = UE ↔ eNodeB 通信品質の指標
- 送信 CQI 0から15まで
- 送信 通信品質の指標 MCSの選択に使用
- 送信 通信品質の指標

CQI 通信品質

CQI 値	送信	通信品質	変調方式
0-3	送信	< 1 Mbps	QPSK
4-6	送信	1-5 Mbps	QPSK
7-9	送信	5-15 Mbps	16-QAM
10-12	送信	15-40 Mbps	64-QAM
13-15	送信	40-150 Mbps	64-QAM

M8010 送信

- M8010C0 - M8010C15の送信0-15の CQI 送信

送信

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

送信 CQI 送信

- 送信 CQI 10-15の送信品質の指標
 - 送信 CQI 7-9の送信品質の指標
 - 送信 CQI 0-6の送信品質の指標
-

CS 測試

M8016 - CS 測試

測試目的：驗證CSFB功能，LTE UE在2G/3G網絡中的性能。

CSFB 測試

- 測試 VoLTE 呼叫過程
- 測試 UE 在 LTE → 2G/3G → LTE 的切换過程
- 測試 2G/3G 網絡中的性能

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 網絡中的性能
2. 測試 UE 在 2G/3G 網絡中的性能

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在2G/3G網絡中的性能。

□□□□

- □□□□□□□□□□
- □ 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

$$X2 = 100 \times M8022C1 / M8022C0$$

> 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

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 %": 100.0 * r.M8013C5 / (r.M8013C17 + r.M8013C18 +
r.M8013C19 + r.M8013C34 + r.M8013C31 + r.M8013C21 + r.M8013C93 +
r.M8013C91)
  })))

```

4G 4G

4G 4G

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

4G 4G

- 4G 4G

- 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
- 网络拥塞

网络工具

- 网络性能 - RAN 性能
 - Grafana 网络 - Grafana 网络
 - 网络性能 - RAN 性能
 - 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 000 DL DTCH 00		
M8001C1	PDCP SDU 000 DL DTCH 00		
M8001C2	PDCP SDU 000 DL DTCH 00		
M8001C3	PDCP SDU 000 UL DTCH 00		
M8001C4	PDCP SDU 000 UL DTCH 00		
M8001C5	PDCP SDU 000 UL DTCH 00		
M8001C6	00 RACH 0000	00	
M8001C7	00 RACH 0000	00	
M8001C8	RACH 0000		0
M8001C9	0 PCH 0000 TB 0		
M8001C10	0 BCH 0000 TB 0		
M8001C11	0 DL-SCH 0000 TB 0		
M8001C12	DL-SCH 00 HARQ 00		
M8001C13	000000 UL-SCH TB		
M8001C14	000 UL-SCH TB 0000		
M8001C15	000 UL-SCH TB 00		
M8001C16	00 MCS0 0 PUSCH 00		
M8001C17	00 MCS1 0 PUSCH 00		
M8001C18	00 MCS2 0 PUSCH 00		
M8001C19	00 MCS3 0 PUSCH 00		
M8001C20	00 MCS4 0 PUSCH 00		
M8001C21	00 MCS5 0 PUSCH 00		
M8001C22	00 MCS6 0 PUSCH 00		
M8001C23	00 MCS7 0 PUSCH 00		
M8001C24	00 MCS8 0 PUSCH 00		
M8001C25	00 MCS9 0 PUSCH 00		
M8001C26	00 MCS10 0 PUSCH 00		
M8001C27	00 MCS11 0 PUSCH 00		
M8001C28	00 MCS12 0 PUSCH 00		
M8001C29	00 MCS13 0 PUSCH 00		
M8001C30	00 MCS14 0 PUSCH 00		
M8001C31	00 MCS15 0 PUSCH 00		
M8001C32	00 MCS16 0 PUSCH 00		
M8001C33	00 MCS17 0 PUSCH 00		
M8001C34	00 MCS18 0 PUSCH 00		
M8001C35	00 MCS19 0 PUSCH 00		
M8001C36	00 MCS20 0 PUSCH 00		
M8001C37	00 MCS21 0 PUSCH 00		
M8001C38	00 MCS22 0 PUSCH 00		
M8001C39	00 MCS23 0 PUSCH 00		
M8001C40	00 MCS24 0 PUSCH 00		
M8001C41	00 MCS25 0 PUSCH 00		
M8001C42	00 MCS26 0 PUSCH 00		

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

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-PRB	M51xxx	~500	5G PRB
5G-RRC	M40xxx	~250	5G RRC

目録

目録 `priv/pm_counters.csv` の項目

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

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 usage of a device.
3. It is used to identify the source and destination of the traffic.
4. It is used to identify the type of traffic.

Configuration

1. PM Counter is configured in the PM Counter configuration page.
2. It is used to monitor the network usage of a device.

Monitoring

Monitoring PM Counter

- PM Counter ID is used to identify the counter.
- PM Counter is used to monitor the network usage of a device.

Access

Access PM Counter via `priv/pm_counters.csv` to view the counter data.

PM Filter

PM Filter

1. PM Filter is configured in the PM Filter configuration page.
2. It is used to filter the traffic based on the source and destination.
3. It is used to filter the traffic based on the type of traffic.

InfluxDB is used to store the PM Counter data.

PM Counter

PM Counter	Radio Access Technology	Description
M8012C23	LTE	Radio Bearer
M8012C26	LTE	Radio Bearer
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,RA,RTT
M8012C23,RA,RTT
M8012C26,RA,RTT
M8001C2,RA,RTT
...
```

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>

RAN 目录结构

config/runtime.exs

目录

1. 目录
 2. 目录
 3. Web 目录
 4. 目录
 5. 目录
 6. InfluxDB 目录
 7. 目录
-

目录

config/runtime.exs 目录 RAN 目录结构

目录

- 目录MySQL
- Web 目录
- 目录
- InfluxDB 目录
- 目录
- 目录

目录

config/runtime.exs

📄📄📄📄📄

📄📄 📄📄 RAN 📄📄📄 **Omnitouch** 📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄

- 📄📄📄📄📄📄📄📄
- 📄📄📄📄📄📄📄📄📄📄📄
- 📄📄📄📄 📄📄📄📄📄📄
- 📄📄📄📄📄📄📄📄

📄📄📄 **Omnitouch** 📄📄📄📄📄📄 **Omnitouch** 📄📄📄📄📄📄📄📄

📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄 [📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄](#) [Grafana](#) 📄📄📄

📄📄📄📄📄

MySQL/MariaDB 📄📄

```
config :ran_monitor, RanMonitor.Repo,  
  username: "omnitouch",  
  password: "omnitouch2024",  
  hostname: "localhost",  
  database: "ran_monitor",  
  stacktrace: true,  
  show_sensitive_data_on_connection_error: true,  
  pool_size: 10
```

📄📄📄 📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄 MySQL 📄📄📄📄📄

📄📄📄📄

username (📄📄📄)

- 📄📄📄📄📄📄
- 📄📄📄📄 "omnitouch"
- 📄📄📄📄 📄📄📄📄 CREATE📄SELECT📄INSERT📄UPDATE📄DELETE 📄📄

- `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)

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

Web UI

RAN `enable` Web `enable`

SOAP/API

```
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)

- `ip` `0`
- `ip` `{0, 0, 0, 0}` `ip` `0`
- `ip` `0`

- `{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
-

環境変数

設定

1. 環境変数

```
# データベースパスワード  
password: "omnitouch2024"  
  
# デフォルトパスワード  
password: System.get_env("DB_PASSWORD") || "default_password"
```

2. 権限

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

3. 無視

- runtime.exs を無視する `.gitignore`
- 権限をリセットする
- 再起動

監視

1. 監視

- 監視する
- 監視する `pool_size`
- 監視する 2 分

2. 設定

- 監視する
- 60 分
- 監視する 15 分

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` の設定
- `info` の設定
- 設定の確認

3. RAN の起動

- 起動コマンド
- 起動の確認
- InfluxDB の接続
- 起動の完了

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
- インストールの完了
- 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 データを保存
- MySQL と InfluxDB の連携

□□□□□□□□

□□□□□□□□□□□□□□□□

```

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 ネットワークの RF トラフィック

トラフィック TCE

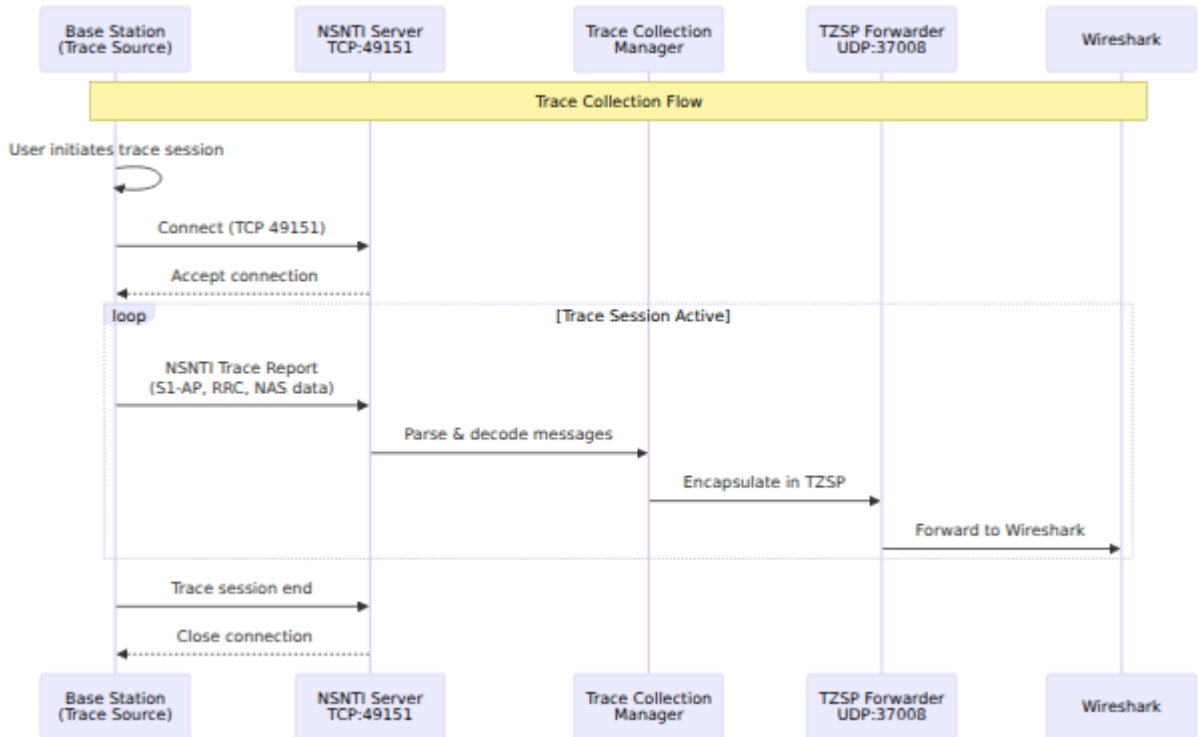
トラフィックを解析する Nokia AirScale ネットワーク

- **S1-AP** プロトコル - eNodeB と EPC 間のプロトコル
- **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 で解析
トラフィック	ASN.1	-	プロトコル S1-AP と RRC

□□□□



□□□□□□□□□□□□□□ 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	설명
36412	S1AP	-	S1AP 인터페이스 (eNodeB ↔ EPC)
37000	RRC	신호	RRC 신호
37001	RRC	DL-CCCH	DL-CCCH
37002	RRC	DL-DCCH	DL-DCCH
37003	RRC	BCCH-DL-SCH	BCCH-DL-SCH (MBSFN 영역)
37004	RRC	PCCH	Paging Control Channel
37008	TZSP	-	TZSP 신호
37011	RRC	UL-CCCH	UL-CCCH (RRC 신호)
37012	RRC	UL-DCCH	UL-DCCH (신호)
38000	MAC-LTE	신호	MAC-LTE 신호
38001	MAC-LTE	신호	신호
38002	MAC-LTE	BCH	BCH
38003	MAC-LTE	PCH	Paging Channel
38011	MAC-LTE	신호	신호
38012	MAC-LTE	RACH	신호

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

# 0000000
slap.HandoverRequired || slap.HandoverCommand

```

00

000000

- 0000000 RF 00
- 0000000
- 0000000 (RSRP, RSRQ, SINR)
- 0000000

000000

- 0000000000
- 0000000
- 0000000
- 0000000

RF 000

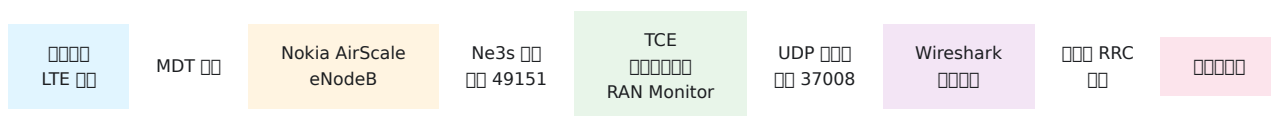
- PCI 00000
- 0000000

- `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

- Web UI: `https://<ran-monitor-ip>:9443`
-
-
-
-
- TCE
 - IP: `<RAN Monitor IP>`
 - : `49151`
-
-
-

AirScale ONS

2 AirScale MDT

MDT

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

ONS

3. 실험 환경

이 실험은 AirScale 기반의 MDT 테스트를 위해 TCE를 구성하는 데 중점을 둡니다.

Wireshark를 사용하여 MDT 테스트

Wireshark 설치

1. Wireshark 설치
2. OS에 따라 설치: Linux (lo), macOS (lo), Windows (Loopback)
3. 필터 설정: `udp port 37008`
4. 테스트 실행

Wireshark를 사용하여 S1AP 메시지 (InitialUEMessage, Attach request) 및 LTE RRC 메시지 (RRCConnectionReject, RRCConnectionReestablishment)를 TCE에서 캡처합니다.

MDT 테스트

이 실험은 MDT 테스트를 위한 환경을 구성하는 데 중점을 둡니다.

```
# 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 ()
```

MDT

MDT

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

3. Excel Python 篩選篩選

篩選

篩選 RSRP/RSRQ 篩選

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

篩選 UE 篩選

```
lte-rrc.MeasResultListEUTRA
```

篩選 RSRP 篩選 RSRQ 篩選

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

篩選

Wireshark 篩選

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

篩選

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

篩選 **AirScale** 篩選 **ONS**

□□□□□□□□

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

□□

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



RAN 配置

配置



1. 配置
 2. 配置
 3. 配置
 4. Web UI 配置
 5. 配置
 6. 配置
 7. 配置
 8. 配置
 9. 配置
-



配置 RAN 配置



1. 配置

- 配置
- 配置
- 配置

2. 配置

- 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

確認事項

確認事項

確認

- Web UI 確認

- ping
- telnet
- http

ping

1. ping

```
# ping
ping <device-ip>

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

ping telnet
-

2. Web UI

Web UI → IP → port → username/password

- IP
- port 8080
- username/password

config/runtime.exs

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

3. Web UI

Web UI → IP → port

□□□

- [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 秒
  }
}
```

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. 配置

```
# 配置 RAN 名称  
date
```

```
# 配置 IP 地址  
# 配置子网掩码
```

配置

配置

1. 配置 `reregister_interval` 为 30
2. 配置 RAN 名称
3. 配置 IP 地址

配置

1. 配置 IP 地址
2. 配置子网掩码
3. 配置网关/默认路由
4. 配置 DNS 服务器

配置

1. 配置 RAN 名称和 NTP
2. 配置 IP 地址
3. 配置子网掩码

配置

1. 配置 IP 地址
 2. 配置子网掩码
 3. 配置网关/默认路由
-

❓❓ 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

- InfluxDB
- `config/runtime.exs`
 -
 - 8086
 - /
 - /API
- RAN
- 8086
- RAN

InfluxDB

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

5. RAN 監視

InfluxDB 監視

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

監視

1. ディスク使用率の監視
2. ディスク使用率の webhook URL 設定
3. ディスク使用率の監視
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

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

2. 設定

Web UI → 設定 → ネットワーク → ネットワーク設定

- ネットワーク設定
- ネットワーク設定

3. ネットワーク InfluxDB 設定

ネットワーク設定 InfluxDB 設定

- ネットワーク InfluxDB 設定
- ネットワーク/設定

設定

RAN 設定

- ネットワーク設定
- ネットワーク設定
- ネットワーク設定

設定

- ネットワーク設定
- ネットワーク設定
- ネットワーク設定
- ネットワーク設定

InfluxDB 設定

- ネットワーク設定
- ネットワーク RAN ネットワーク設定
- ネットワーク InfluxDB 設定
- ネットワーク設定

設定

- ネットワーク RAN ネットワーク設定

- `docker run --rm -p 9443:9443`
 - `docker run --rm -p 9443:9443 --name ran-monitor`
 - `docker run --rm -p 9443:9443 --name ran-monitor --network ran-network`
-

Web UI

Web UI

Web UI

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

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
```

配置

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

配置

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

配置

1. 配置 MySQL 服务

2. 配置 MySQL 服务 3306

3. 配置 MySQL 服务

配置

1. MySQL max_connections
 2. pool_size
 3. RAN
-

MySQL

MySQL CPU

MySQL

- RAN CPU RAM
-
-
-

MySQL

1. MySQL

```
# CPU  
top -p $(pgrep -f ran_monitor)  
  
#  
ps aux | grep ran_monitor
```

2. MySQL

MySQL

- =
-

3. MySQL

MySQL

- 100 = 100 CPU/1000
- 1000 10 1000000

4. 10000000000

config/runtime.exs

```
pool_size: 10 # 1000000
```

100000

1000000000000000

1. 1000000000
2. 1000000000 CPU/RAM
3. 10000000000
4. 1000000000000000

1000000000

1. 1000000 pool_size
2. 1000000000 2 1000 + 5 1000 Web UI
3. 100 RAN 1000
4. 1000000000

100000

1. 1000000000
2. 1000000000000000000000
3. 1000000000 RAN 1000
4. 1000000000000000

InfluxDB 100000

1. 100 InfluxDB 1000000
2. 100 InfluxDB 100000
3. 100000 InfluxDB 1000
4. 1000000000000000

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

InfluxDB

1. InfluxDB
- 2.
3. InfluxDB
- 4.

Web UI

- 1.
- 2.
- 3.
- 4.

systemd

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

mix

-

- / -

- 00 - 00000000
- 00 - 0000
- 00 - 0000000
- 00 - 0000000

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

操作数据库

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

操作数据库

```
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 Web

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

-
-



RAN Web Web UI

Web UI Grafana

Web UI

-
-
-
-
-

Grafana

-
- KPI
-
-
-

Grafana Grafana

Web UI

HTTPS

URL:

: 9443

SSL

-

- 証明書 CA 証明書
- 証明書 `config/runtime.exe` 証明書

証明書証明書証明書 証明書証明書

証明書 証明書 5 証明書証明書証明書証明書

証明書

証明書 RAN 証明書証明書

証明書


証明書

- 証明書
- 証明書証明書証明書

証明書

- 証明書/証明書
- 証明書
- 証明書

証明書

- 証明書証明書
- 証明書  証明書証明書証明書
- 証明書

証明書

- 証明書
- 証明書
- 証明書

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

□□□

□	□□
□□	□□□□□□□□
□□	□□□□□“□□□”□□□□□“□□□”□□□□□
□□	□□ 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

□□□□□□

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

□□□□□□

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

□□□□

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

□□□□

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

□□□□

- □□□□□
- □□□□□□□
- □□□□□
- □□□□□□

□□□□

□□□□□□□□□□□□□□□□

□□□□

□□□□□□

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

□□□□□□□□

- □□□□
- □□□□
- □□□□□□□□ DN - Distinguished Name□
- □□□□□□□□□□□□□□□□

□□

□□□□□□

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

□□□□□□□□

- □□□□□□□□□□□□

- 数据源
- 数据源

数据源

- 数据源
- 数据源
- 数据源

数据源 数据源

数据源

Web UI 数据源

数据源

数据源

数据源

1. 数据源
2. 数据“数据源”
3. 数据源
4. 数据源 XML 数据源

数据源 数据源

数据源

数据源

数据源

1. 数据 XML 数据源
2. 数据“数据源”

3. 000000000000“00”0

4. 0000000000 ID 000000

000 000000000000 - 00000000

00000

000 0000000000000000

000

1. 00000000 ID

2. 00“00”

3. 0000000000

4. 0000000000000000

000 0000000000000000000000

00000

000 000000000000

000

1. 00000000 ID

2. 00“0000”

3. 0000000000

4. 0000000000/00

000 00000000000000000000 - 0000000000

000000

00000

1. 0000000000

2. 000000

3. 0000000000 ID0

4. 配置 AirScale 软件
5. 配置 AirScale 软件
6. 配置 AirScale 软件
7. 配置 AirScale 软件

配置 AirScale 软件

配置 eNodeBs 软件

配置 eNodeBs 软件

配置

配置 eNodeBs 软件

- 配置 AirScale 软件
- 配置 AirScale 软件
- 配置 AirScale 软件
- 配置 AirScale 软件

配置

配置 ID

- 配置 AirScale 软件
- 配置 AirScale 软件 ID

配置

- 配置 AirScale 软件
- 配置 AirScale 软件

配置

- 配置 AirScale 软件
- 配置 AirScale 软件

□□□□

- □□□□□□□□□□
- □□□□□□□□

□□□□□

□□

- □□□□□□□□□□
- □□□□□□□□□□

□□

- □□□□□□□□□□
- □□□□□□□□□□

□□□□

- □□□□□□□□□□
- □□□□□□

□□□□

□□□□□□□□□□□□□□□□□□

1. □□□□□□ **ID**
2. □□□□□□□□ `config/runtime.exe` □□

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

3. 關於 RAN 的 重要事項

關於 RAN 的 重要事項

關於

- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項

關於 RAN 的 重要事項

關於 RAN 的 重要事項

關於 RAN 的 重要事項

關於 RAN 的 重要事項

- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項
- 關於 RAN 的 重要事項

關於 RAN 的 重要事項

關於

關於 RAN 的 重要事項

- 關於 RAN 的 重要事項

- 500
-

- / -
- -
- -

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

-
- IP
-

-
-
-

-
-
-

- 部署環境の準備

インストール

- 依存パッケージのインストール
- 実行環境の準備
- サービスの起動

設定

- データベース接続の設定
 - ログ出力の設定
 - ログレベル
 - ログ出力先
 - ログ出力形式
 - サービスの再起動
 - サービスの停止
-

トラブルシューティング

よくある問題と InfluxDB のエラーメッセージ

FAQ

よくある質問

- 接続エラーの原因と対処法
- `config/config.exe` の設定方法
- サービスが起動しない場合の対処法

お問い合わせ

- お問い合わせ先
- 問い合わせの受付時間

□□□□□

- □□□□□□□□□□
- □□□□□□□

□□□□□

□□□□□□□□□□

□□□□□

- □□□□
- □□□□□□□□□□/□□□□
- □□□□□□□

□□□□□

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

□□□

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

□□□□□□□

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

XXXXXXXXXXXXXXXXXXXXXXXXXXXX InfluxDB XXXXX

XXXX

XX	XX
XXXX	XXXXXXXXXXXXXXXXXXXX
XX	XX InfluxDB XXXXX
InfluxDB XX	XXXXXXXXXXXX2.x

XXXXXX

XXXXXXXXXXXXXXXXXXXX

項目	説明
PM	PM 値
PM	PM 値
PM	PM 値
PM	PM 値
PM	PM 値

PM 値

InfluxDB での PM 値の取得方法

項目	説明
PM	PM 値 < 1000 < 10000 < 20000 > = 20000
PM	PM 値
PM	PM 値
PM	PM 値
PM	PM 値
PM	PM 値
PM	PM 値
PM	PM 値

PM 値

項目	内容
概要	このドキュメントは InfluxDB のインストールと設定に関する情報を提供します。
前提条件	インストールには Linux または macOS が必要です。
インストール	このセクションでは、InfluxDB をシステムにインストールする方法を説明します。
設定	インストール後、InfluxDB の設定を変更する方法を説明します。
起動	インストールと設定が完了したら、InfluxDB を起動する方法を説明します。
接続	InfluxDB に接続するためのクライアントのインストールと設定方法を説明します。
トラブルシューティング	インストールや設定中に発生する一般的な問題を解決するためのヒントを提供します。
お問い合わせ	追加のサポートが必要な場合は、お問い合わせください。

このドキュメントは InfluxDB のインストールと設定に関する情報を提供します。

目次

項目	内容
概要	このドキュメントの目的と範囲を説明します。
前提条件	インストールに必要なシステム要件を説明します。
インストール	InfluxDB をシステムにインストールする方法を説明します。
設定	InfluxDB の設定を変更する方法を説明します。
起動	InfluxDB を起動する方法を説明します。
接続	InfluxDB に接続するためのクライアントのインストールと設定方法を説明します。
トラブルシューティング	インストールや設定中に発生する一般的な問題を解決するためのヒントを提供します。
お問い合わせ	追加のサポートが必要な場合は、お問い合わせください。

□□□□□□

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

□□□□

□□□□□□□□□□

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


□□□□

□□□ 30 □□□□□□

□□□□

□□	□□
□□ + □□□□□□	□□□□□□
□□ + □□□	□□□□□□□ PM □□□□□
□□□	□□ InfluxDB □□□□□□□□□□□
□□□□□□□□/□□□	InfluxDB □□□□□□□□□□□
□□□□□	□□□□ - □□□□□□□□□□□□□□□
□□□□	□□ - □□□□□□□□□□

□□

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

□□□□□□□

□□□□ InfluxDB □□□□□□□□□□□□□□□□

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

InfluxDB *Airscale*

InfluxDB

	InfluxDB
	InfluxDB

Airscale InfluxDB

InfluxDB

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

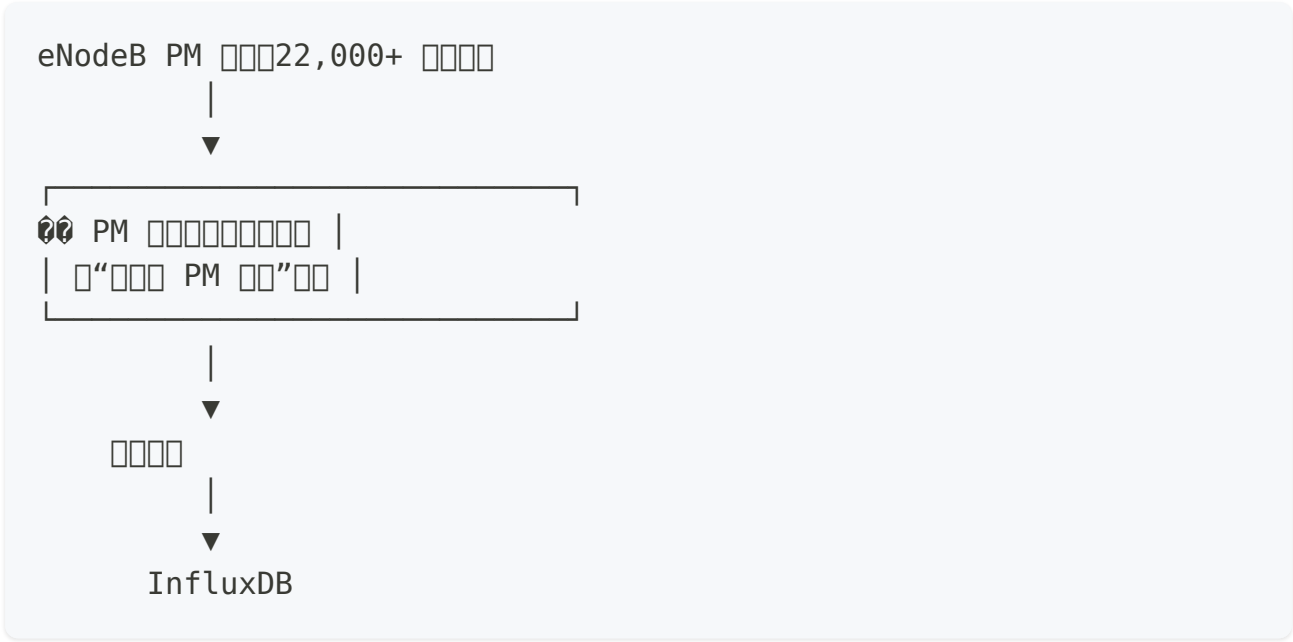
□□□□□□□□□□□□□□□□□□

□□□□

Erlang VM □□□□□□

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

PM 配置



PM configuration keys are used to store PM configuration data in InfluxDB.

Configuration

Item	Value
PM configuration	InfluxDB configuration
PM keys	22,000+ cells
PM keys	config_keys.csv

PM Configuration

PM configuration is stored in InfluxDB.

Configuration

- PM configuration ID
- PM configuration LTE/5G-NR
- PM configuration CSV

000

- 0000000000000000
- 0000000000000000
- 00000000 pm_counters.csv 000000

00000000

00000000 Nokia PM 0000000000

0000

- ID 00000000000000
- 000000000000

000

- 0000000000000000
- 000000000000000000000000 200 00

000000

00	0000	00
LTE	M8xxx	LTE L1/L2/L3 000
WCDMA	M5xxx	3G WCDMA 000
5G-NR	M55xxx	5G NR 000
5G-000	M51xxx	5G 00000
5G-00	M40xxx	5G 00000

000

000

- `priv/pm_filters.etf`
-
-

PM

InfluxDB

- **PM**
-

PM

PM

Nokia AirScale

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

PM

RAN WebEM " " HTTP URL

PM

	Nokia AirScale .zip
URL	WebEM HTTP URL
MD5	

□□□□□□

1. □□□□□□ RAN □□
2. □□ □□ **URL** □□□□ URL
3. □ WebEM □□□□□□□□ □□□□
4. □□ URL □□□ □□□□□
5. □□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□ □□□□□□

□□□□□□

□□□□□□□□□□□□□□□□

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

□□ ETS □□□□□□□□ InfluxDB □□□□□□□□□□□□□□

ETS 部署ガイド

このガイドは、ETS をインストールおよび設定するための手順を説明します。

OS	プラットフォーム
Nokia 9000	ARMv7
Nokia 9000	ARMv8
Ubuntu	ARMv7
InfluxDB 2.0	ARM InfluxDB 2.0

このガイドは ETS のインストールと設定に関する手順を説明しています。

前提条件

インストール前に以下の条件を満たす必要があります。

- 十分なディスク容量 (TAR ファイル)
- 十分なメモリ

インストールディレクトリとして `/tmp` を指定してください。

InfluxDB 2.0 のインストール

InfluxDB 2.0 をインストールするには、以下の手順に従ってください。

- インフラストラクチャの準備
- インストール
- 設定

このガイドは、InfluxDB 2.0 のインストールと設定に関する手順を説明しています。

このガイドは、InfluxDB 2.0 のインストールと設定に関する手順を説明しています。

□□□□□

□□□□□□□□□□□□□□□□□□□□ `config/runtime.exs` □□□□□□□ ETS □□

□□

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



Web UI □□□

□□ Web UI □□□□□□□□□□

□□□□□□□

□□□ □□□□□□□□□□□□□□

□□□

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

□□□ < 5 □□

□□□□□

□□□ □□□□□□□□

□□□

1. 11111111
2. 22222222222222
3. 33333333333333
4. 44444444
5. 5555555555
6. 66666666
7. 777777
8. 88888888

1111111111111111 11111111

1111

111 11111111

111

1. 1111111111
2. 22222222222222
3. 33333333
4. 4444444 ID
5. 5555 ID 1111
6. 66666666666666
7. 77777777
8. 88888888 15-30 11
9. 99999999999999

11111111

- 11111111111111
- 22222222222222
- 33333333
- 4444444

環境構築

環境構築の準備

環境

1. 仮想マシン eNodeBs を構築
2. 仮想マシン ID と IP を設定
3. 仮想マシンで `config/runtime.exs` を編集
4. 仮想マシン RAN を構築
5. 仮想マシンを起動
6. 仮想マシン間の接続を確認
7. 仮想マシンで InfluxDB をインストール
8. 仮想マシンで InfluxDB を起動
9. 仮想マシンで Grafana をインストール

環境構築の完了を確認

環境構築の完了

環境構築の完了を確認

環境

1. 仮想マシンを起動
2. 仮想マシン/ネットワークを確認
3. 仮想マシン間の接続を確認
4. 仮想マシンで ping を実行
5. 仮想マシンで ping を実行
6. 仮想マシンで ping を実行
7. 仮想マシンで ping を実行
8. 仮想マシンで ping を実行
9. 仮想マシンで ping を実行
10. 仮想マシンで ping を実行

環境構築の完了を確認

□□□□

- □□□□ - □□□□□□
- □□□□□□ - □□□□□□
- □□□□□□ - □□□□□□
- □□□□□□ - □□□□
- **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

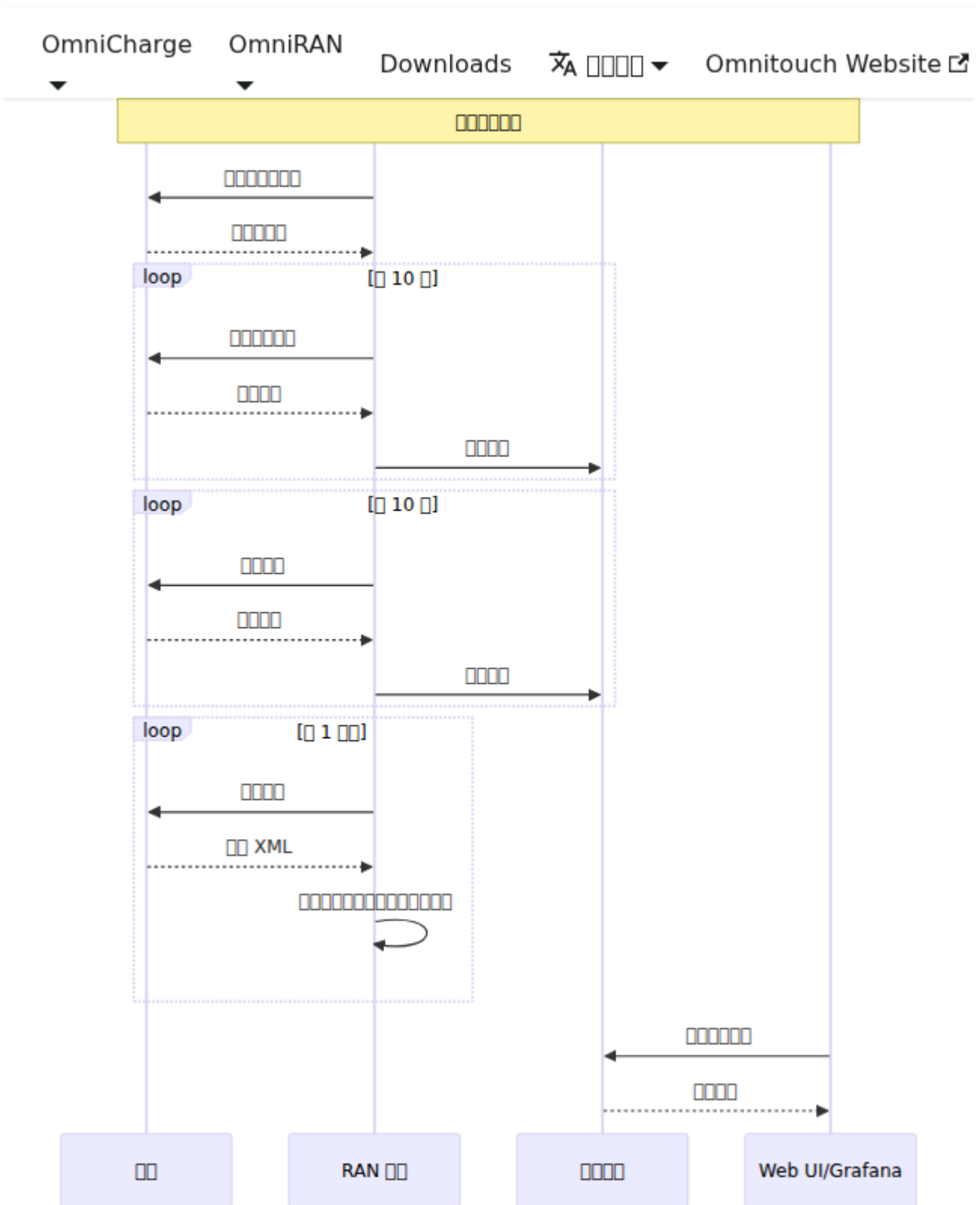
- 部署 - 部署指南
- 配置 - 部署指南
- 验证 - 部署指南
- **Web** 部署 - 部署 Web UI 部署指南
- **Grafana** 部署 - 部署指南

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

□□□



□□□□□

□□□□□

- 00000000000000
- 00000000/0000
- 000000PRB 0000
- 00000000
- 0000
- 00000000

000

- 000000000000000000000000
- 0000000000
- 0000000000
- 0000000000

000

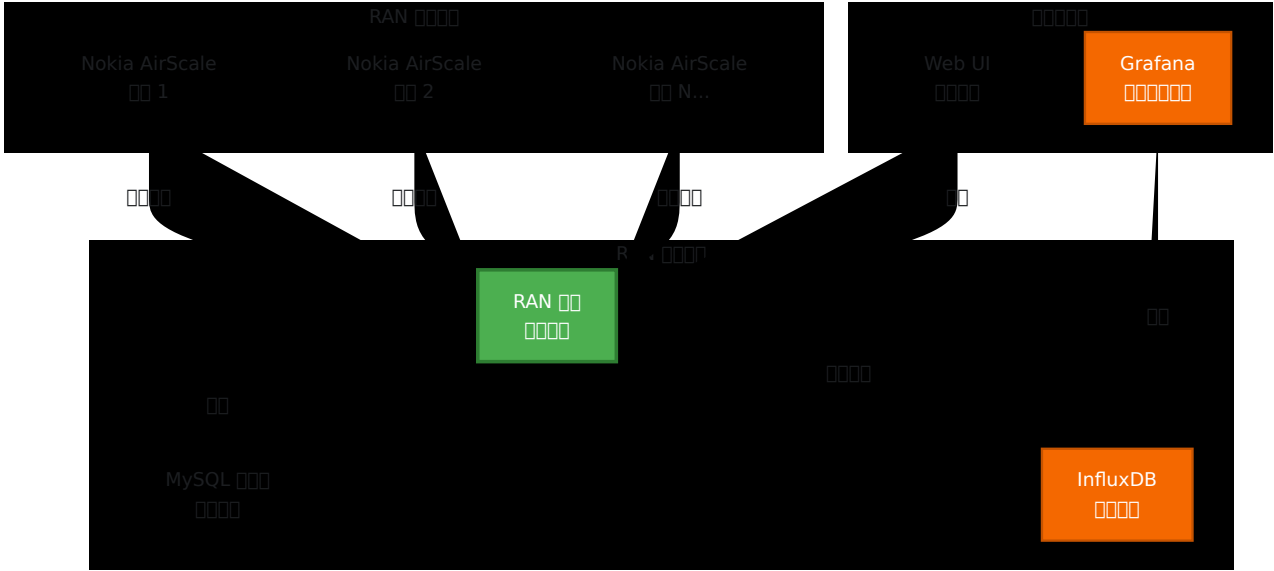
- 000 XML 0000000000000000
- 000000000000
- 0000000000
- 00000000 10 000

0000000000000000 00000000

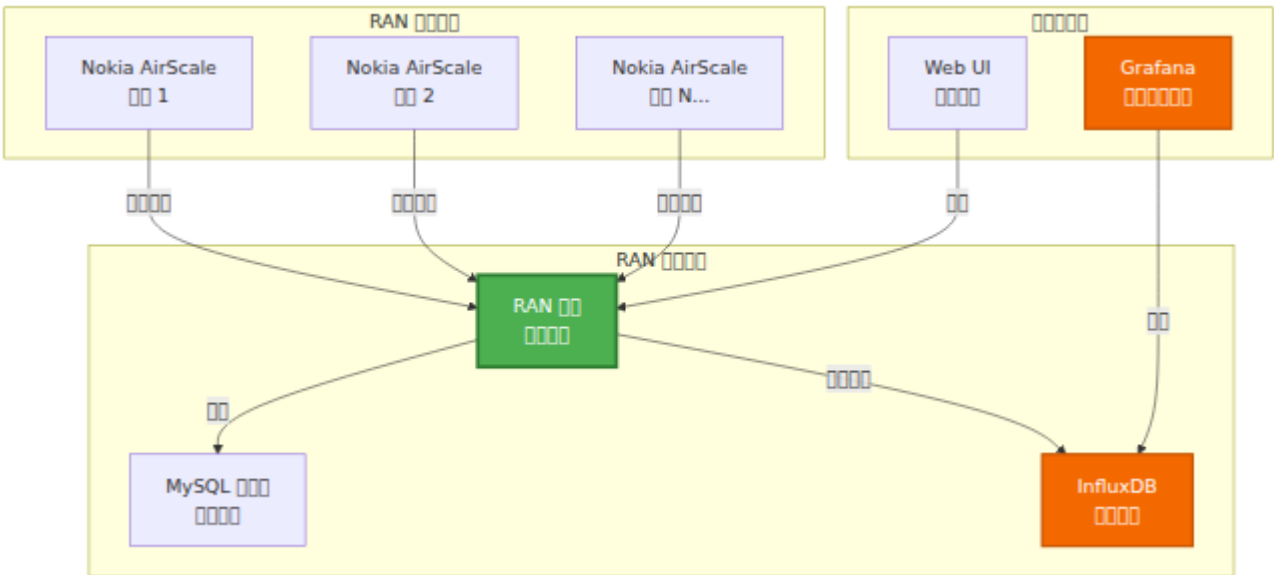
0000000000000000 Nokia 00000000

□□□□

□□□□□□



??□□□



□□□□□□□□□□□□□□ □□□□□□□□

Network (TCE)

RAN Network Architecture for LTE/5G Network RF Network

Network TCE

Network Architecture Nokia AirScale Network

- **S1-AP** - eNodeB - EPC Network
- **RRC** - Network
- **NAS** - Network
- Network - PDCP Network

Network

Network

- Network RF Network
- Network
- Network RSRP RSRQ SINR
- Network

Network

- Network
- Network
- Network
- Network

RF Network

- PCI Network
- Network
- Network
- Network

Network Wireshark Network **TCE MDT** Network

Web UI

RAN Web UI

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

Placeholder text

Placeholder text

Placeholder text

Placeholder text

- Placeholder text
- Placeholder text/Placeholder text
- Placeholder text
- Placeholder text

5 Placeholder text

Placeholder text

Placeholder text

- Placeholder text = Placeholder text = Placeholder text
- Placeholder text
- Placeholder text
- Placeholder text

Placeholder text

□□□□

□□□□□□□□□□

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

□□□□□□□□□□□□□□□□ □□□□□□

📄 eNodeB

📄📄📄📄📄📄📄📄

- 📄 ID📄📄📄📄📄📄
- 📄📄📄📄
- 📄📄📄📄
- 📄📄📄📄📄📄📄📄

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

□□□□

□□□□□□□□□□□□□□

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

□□□□□□□□□□□□□□□□ □□□□□□

□□□□□

□□□□ InfluxDB □□□□□

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

□□□□□□□□□□□□ □□□□□□□□

InfluxDB 简介

时间序列数据库

- 易于使用
- 易于集成
- 易于部署
- 易于扩展
- 5 年质保

核心特性

- 查询 + 写入 = 高性能
- 查询 + 写入 = 可扩展
- 查询 = 与 InfluxDB 集成

Web UI

Web UI

Web UI -

Grafana

Web UI Grafana

Grafana

Grafana

-
- KPI
-
-

- SLA
-

Web UI

-
-
-
-
-

Grafana

-
-
-
-
-

NOC

-
-
- 24
-
-

-
- SINR RSRP
- RLC RRC
-
-

Nokia AirScale

- PRB
- PDCP
- UE
-
-
- RSSI
- RRC
- VSWR
-

Grafana -

Nokia -

□□□□

□□□□

□□□□□□**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. `config/runtime.exs` 配置
3. 数据库配置
4. 部署 RAN 应用
5. 部署 Grafana 应用

部署环境

1. 部署环境
2. `config/runtime.exs` 配置
3. 部署 RAN 应用
4. 部署数据库

部署应用

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	Monitoring and configuration interface
Alerts	Notifications	Real-time alerts and notifications
Reports	Analytics	Performance reports and analytics
Configuration	Settings	System configuration and settings
Users	Roles	User management and roles
Integrations	APIs	External integrations and APIs
Security	Access	Security and access control

Tools and Integrations

Tool	Integration	Usage
Grafana	Monitoring	Visualizing metrics and logs
Nokia	Network	Network performance monitoring
AirScale	Network	Network performance monitoring
TCE MDT	MDT, Wireshark	Network traffic analysis
API	REST API	System integration

□□□□□□□□

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

□□□□

□□□ **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. □□□□□□ AirScale □□
 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. □□□□□□□□□□
-

