

センサー情報相互運用配信システム 配信設定マニュアル

Ver. 1.0.0

2015/9/1

国立研究開発法人 防災科学技術研究所

変更履歴

Version	変更日付	変更内容
1.0	2015/9/1	初版作成

目次

1	概要	1
2	SOS 配信システム概要	1
2.1	システム構成	1
2.2	センサーデータの処理概要	1
2.2.1	CSV ファイル形式のセンサーデータの処理概要	1
2.2.2	メール形式によるセンサーデータの処理概要	2
3	SOS 配信設定	2
3.1	設定フロー	2
3.1.1	プロパティファイルの作成	2
3.1.2	取得するセンサーのデータベーステーブル作成	3
3.2	対応するデータ形式詳細	4
3.2.1	CSV ファイル形式	4
3.2.2	メール形式	4
3.3	具体的な設定要領	4
3.3.1	静岡県のファイルフォーマット詳細	5
3.3.2	プロパティファイルの作成	5
3.3.3	取得するセンサーのデータベーステーブル作成	7

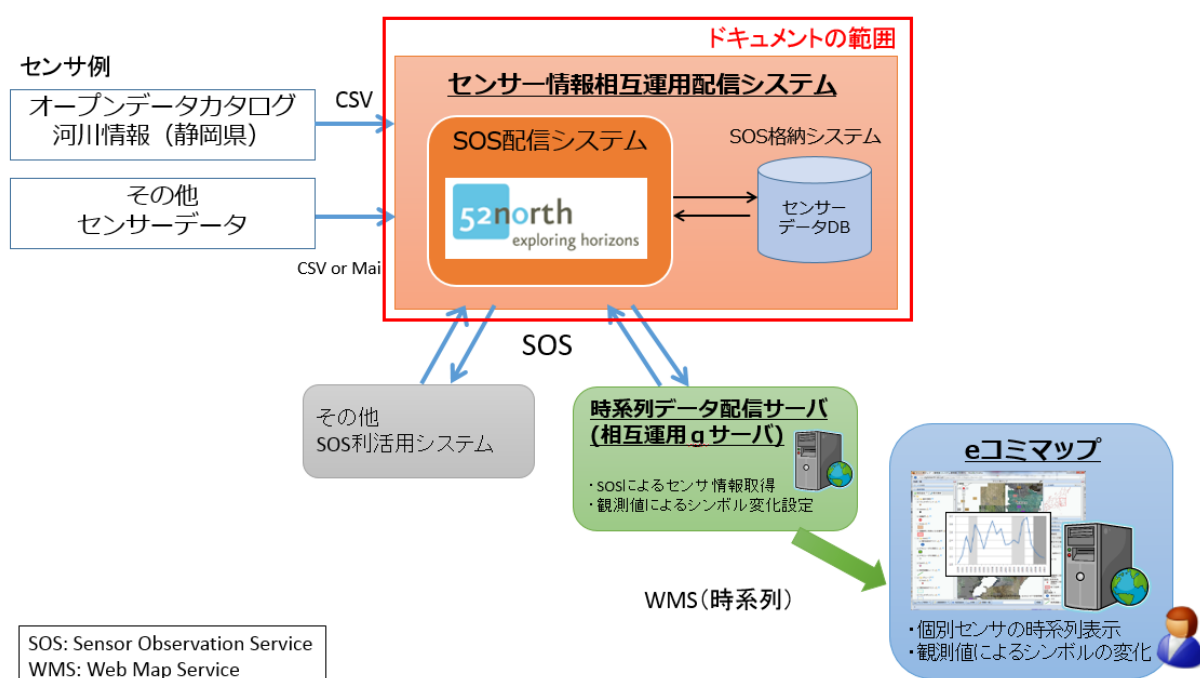
1 概要

本設定マニュアルでは防災科学技術研究所が開発したセンサー情報相互運用配信システムを利用する上での設定方法を記述する。

2 センサー情報相互運用配信システム概要

2.1 システム構成

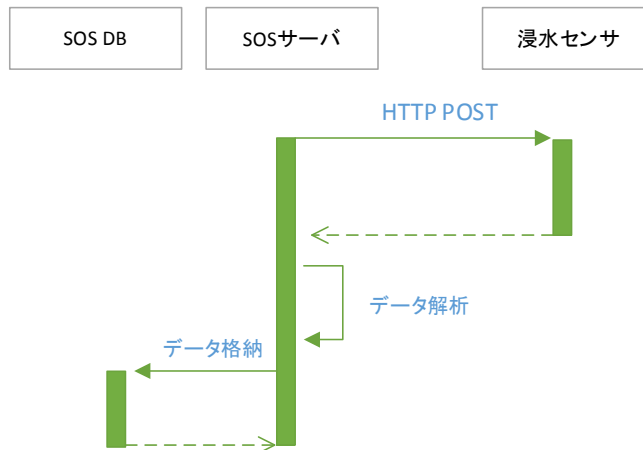
下図はセンサー情報相互運用配信システムの構成図である。本システムは、CSV フォーマットやメールで配信あるいは公開されているデータを取得する格納システムと格納されたセンサーデータを国際標準のセンサー情報の API 仕様（Sensor Observation Service）を満たした形で配信する配信システムで構成される。



2.2 センサーデータの処理概要

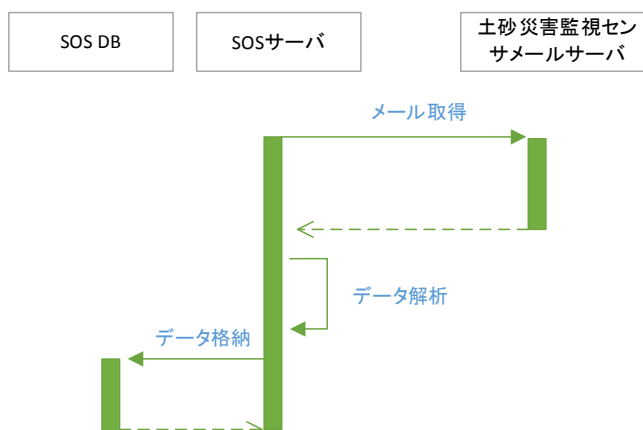
2.2.1 CSV ファイル形式のセンサーデータの処理概要

指定された URL にアクセスし CSV ファイルをダウンロードする。ダウンロードした CSV ファイルを解析し、観測時刻と観測値を取得し、定められたデータベーステーブルに格納する。



2.2.2 メール形式によるセンサーデータの処理概要

対象となるメール配信サーバーに本システム用のメールアドレスを登録する。本システムは登録されたメールアドレスにアクセスし、メールを取得する。取得されたメールを解析し定められたデータベーステーブルに格納する。



3 SOS 配信設定

3.1 設定フロー

設定フローの概要は以下のとおり

- 1) プロパティファイルの作成 (手動)
- 2) 取得するセンサーのデータベーステーブル作成 (SQL 文を作成しデータベーステーブルに流し込む)

3.1.1 プロパティファイルの作成

各センサーの設定ファイル (プロパティファイル) では、データ取得に必要な情報 (メール形式であればメールサーバの接続情報等、ウェブアクセスであれば URL パターンなど) や取得したデータの解析に必要な情報を設定する。

ここではメール形式の場合の設定ファイルを記載する。その他については「3.3.2 プロパティファイルの作成」を参照のこと。

設定ファイル：パッケージ NIED_SOS¥SOSDataService¥conf¥xxxsensor.properties

ファイル名	xxxsensor.properties
ファイル説明	メール配信データの設定ファイル
場所	conf/
連動するテーブル名	xxx_sensor

項目	説明	値
DATA_CRAWL_INTERVAL	データ得る間隔 (秒)	300
CSV_DATA_ROW_INDEX	CSV 無視する行数	0
CSV_DATETIME_FORMAT	CSV 日付フォーマット	yy/MM/dd HH:mm:ss
CSV_DATETIME_TIMEZONE	日付時間帯	Asia/Tokyo
PROCEDURE_ID_PATTERN	Procedure Id のパターン \${xxx_sensor_id} はデータベースのテーブル Id により自動作成する	urn:ogc:object:feature:Sensor:NIED:flood-\${xxx_sensor_id}
FOI_ID_PATTERN	Feature of Interest Id のパターン \${xxx_sensor_id} はデータベースのテーブル Id により自動作成する	foi_xxx_sensor_\${xxx_sensor_id}
POP3_HOST	メールサーバのホスト名、IP	
POP3_PORT	メールサーバのポート	
POP3_SSL	SSL 有無	true
POP3_USER	ユーザー名	
POP3_PASS	パスワード	

メールサーバの情報を記載

3.1.2 取得するセンサーのデータベーステーブル作成

センサー位置（緯度、経度）情報のデータ作成する。詳細は、「3.3.3 取得するセンサーのデータベーステーブル作成」を参照のこと。

3.2 対応するデータ形式詳細

本システムが対応するデータ形式は以下2種類である。

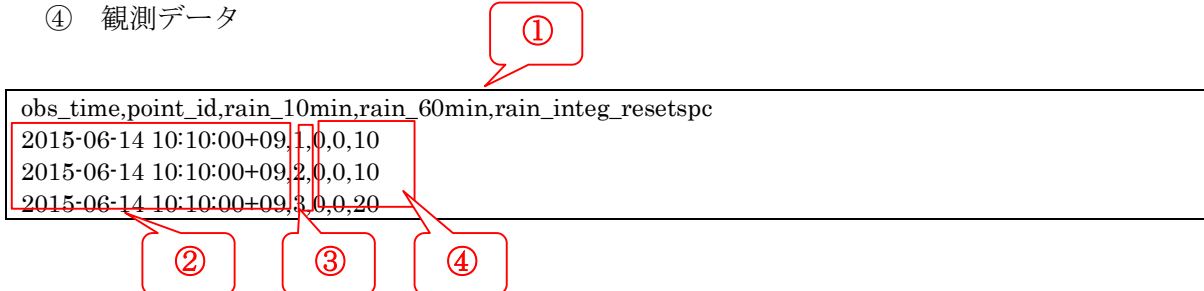
- 1) CSV ファイル形式
- 2) メール形式

3.2.1 CSV ファイル形式

指定された URL にアクセスし CSV ファイルをダウンロードする。ダウンロードした CSV ファイルを解析し、観測時刻と観測値を取得し、定められたデータベーステーブルに格納する。

対応する CSV ファイルには、以下の①～④が含まれていると想定する。

- ① ヘッダー部
- ② 観測時刻
- ③ 観測地（観測データの地物番号等）
- ④ 観測データ



```
obs_time,point_id,rain_10min,rain_60min,rain_integ_resetspc
2015-06-14 10:10:00+09,1,0,0,10
2015-06-14 10:10:00+09,2,0,0,10
2015-06-14 10:10:00+09,3,0,0,20
```

ヘッダー部の記述は CSV ファイル解析処理により無視される。プロパティファイルでは、この無視するヘッダー行数を指定する。（後述 CSV_DATA_ROW_INDEX）

②～④はデータ部分であり、カンマで区切られている。②は日付フォーマットを表しておりプロパティファイルで日付フォーマットを指定する。この例では、yyyy-MM-dd HH:mm:ssZ の日付フォーマットを示す。（後述 CSV_DATETIME_FORMAT）

③は、センサーが配置されている地物の番号であり、その後に観測データが並ぶ。

3.2.2 メール形式

対象となるメール配信サーバーに本システム用のメールアドレスを登録する。本システムは登録されたメールアドレスにアクセスし、メールを取得する。取得されたメールを解析し定められたデータベーステーブルに格納する。

3.3 具体的な設定要領

以下では静岡県公開データを SOS 格納配信システムで配信する場合の設定手順を記述する。

3.3.1 静岡県のファイルフォーマット詳細¹

CSV ファイルのヘッダー部
(実際は 1 行)

obs_time 観測時刻	point_id 雨量観 測局番 号	rain_10min 10分雨量 (10倍値 mm)	rain_60min 60分雨量 (10倍値 mm)	rain_integ_resetspc 24時間連続雨量 (10倍値 mm)
2014-06-25 15:50:00+09	15	0	0	0
2014-06-25 15:50:00+09	19	0	0	0
2014-06-25 15:50:00+09	20	0	0	0
2014-06-25 15:50:00+09	23	0	0	0
2014-06-25 15:50:00+09	37	0	0	0
2014-06-25 15:50:00+09	101	70	140	190
2014-06-25 15:50:00+09	102	90	200	250
2014-06-25 15:50:00+09	103	0	20	320
2014-06-25 15:50:00+09	104	100	140	170
2014-06-25 15:50:00+09	105	0	0	20
2014-06-25 15:50:00+09	106	0	10	50
2014-06-25 15:50:00+09	201	0	40	70

3.3.2 プロパティファイルの作成

URL パターン、データ取得間隔、CSV ファイルフォーマットを設定するテキストファイルである。

shizuoka_rainsensor.properties ファイル情報一覧

ファイル名	shizuoka_rainsensor.properties
ファイル説明	雨量センサの設定ファイル
場所	conf/
連動するテーブル名	shizuoka_rain_sensor

項目	説明	値
DATA_CRAWL_INTERVAL	データ得る間隔 (秒)	60
DATA_CRAWL_TIME_OFFSET	URL で使われる日付との差 (秒)	-300

¹ ふじのくにオープンデータカタログ

http://open-data.pref.shizuoka.jp/mu0b05bub-139/#_139

雨量情報 <http://sipos.shizuoka2.jp/rain-river/DownLoad.html?type=rain&mode=list>

雨量観測局情報

http://sipos.shizuoka2.jp/rain-river/master/Shizuoka_Rain_ObservationPoint.csv

URL_PATTERN	URL パターン \${formatted_datetime} は日付パターン自動作成箇所	http://sipos.shizuoka2.jp/rain-river/Rain/Shizuoka_Rain_\${formatted_datetime}.csv
FORMATTED_DATETIME_EXPR	URL 日付パターン	yyyyMMddHHmm
FORMATTED_DATETIME_STEP	URL 日付のステップ (分)	5
CSV_DATA_ROW_INDEX	CSV 無視する行数	1
CSV_DATETIME_FORMAT	CSV 日付フォーマット	yyyy-MM-dd HH:mm:ssZ
CSV_DATETIME_TIMEZONE	CSV 日付時間帯	Asia/Tokyo
PROCEDURE_ID_PATTERN	Procedure Id のパターン \${rain_sensor_id} はデータベースのテーブル Id により自動作成する	urn:ogc:object:feature:Sensor:NIED:rain-\${rain_sensor_id}
FOI_ID_PATTERN	Feature of Interest Id のパターン \${rain_sensor_id} はデータベースのテーブル Id により自動作成する	foi_rain_sensor_\${rain_sensor_id}

shizuoka_rainsensor.properties ファイル内容

```
#Crawling Interval (second)
DATA_CRAWL_INTERVAL=60
#Crawling Time Offset (second)
DATA_CRAWL_TIME_OFFSET=-300

#Date time formatting expression used in url_pattern | sample:
http://sipos.shizuoka2.jp/rain-river/Rain/Shizuoka_Rain_201410151625.csv
URL_PATTERN=http://sipos.shizuoka2.jp/rain-river/Rain/Shizuoka_Rain_${formatted_datetime}.csv

#Date time formatting expression used in url_pattern
FORMATTED_DATETIME_EXPR=yyyyMMddHHmm

#File formatted datetime stepping (minute)
FORMATTED_DATETIME_STEP=5

#CSV Data row index
CSV_DATA_ROW_INDEX=1

#CSV Date Time Format // 2014-10-10 17:00:00+09
CSV_DATETIME_FORMAT=yyyy-MM-dd HH:mm:ssZ
CSV_DATETIME_TIMEZONE=Asia/Tokyo

#OGC
PROCEDURE_ID_PATTERN=urn:ogc:object:feature:Sensor:NIED:rain-${rain_sensor_id}
FOI_ID_PATTERN=foi_rain_sensor_${rain_sensor_id}
```

3.3.3 取得するセンサーのデータベーステーブル作成

センサーの位置情報（観測局情報）を事前にデータベースにテーブル定義し、格納する必要がある。

shizuoka_rain_sensor

テーブル名	shizuoka_rain_sensor
説明	雨量センサ設定テーブル

フィールド名	データ型	説明
rain_sensor_id[PK]	int	雨量センサ ID
name	varchar(50)	センサ名
office_id	int	雨量センサ事務所 Id
address	varchar(250)	住所（参考のために保留）
latitude	double	センサの経度
longitude	double	センサの緯度
feature_of_interest_id	varchar(100)	SOS 配信システムから戻した FOI の ID （RegisterSensor を作成する際、設定ファイルに指定した FOI_ID_PATTERN を基づき生成する）
last_timestamp	timestamp	最後のデータ取得した時間
procedure_id	varchar(100)	SOS 配信システムから戻した Procedure の ID （RegisterSensor を作成する際、設定ファイルに指定した PROCEDURE_ID_PATTERN を基づき生成する）
date_column_index	int	データを取込際に日付データのコラム位置

テーブル定義作成

```
CREATE TABLE shizuoka_rain_sensor (
  rain_sensor_id integer NOT NULL,
  name character varying(50),
  office_id integer,
  address character varying(250),
  latitude double precision,
  longitude double precision,
  feature_of_interest_id character varying(100),
  last_timestamp timestamp with time zone,
  procedure_id character varying(100),
  date_column_index integer
);

ALTER TABLE public.shizuoka_rain_sensor OWNER TO postgres;
```

レコード挿入

```
INSERT INTO shizuoka_rain_sensor (rain_sensor_id, name, office_id, address, latitude, longitude, feature_of_interest_id, last_timestamp, procedure_id, date_column_index) VALUES (419, '俵沢', 20, '静岡市葵区俵沢', 35.104100000000003, 138.3683, NULL, '2000-01-01 09:00:00+09', NULL, 0);
```

```

INSERT INTO sensor_observation_setting (id, table_name, ref_sensor_id, csv_column_index,
calculate_expr, phenomenon_id, offering_id, uom) VALUES (1, 'shizuoka_rain_sensor', 1, 2, 'obsVal/10',
'urn:ogc:def:phenomenon:OGC:1.0.30:precipitation', 'PRECIPITATION_10M', 'mm');
INSERT INTO sensor_observation_setting (id, table_name, ref_sensor_id, csv_column_index,
calculate_expr, phenomenon_id, offering_id, uom) VALUES (2, 'shizuoka_rain_sensor', 1, 3, 'obsVal/10',
'urn:ogc:def:phenomenon:OGC:1.0.30:precipitation', 'PRECIPITATION_60M', 'mm');
INSERT INTO sensor_observation_setting (id, table_name, ref_sensor_id, csv_column_index,
calculate_expr, phenomenon_id, offering_id, uom) VALUES (3, 'shizuoka_rain_sensor', 1, 4, 'obsVal/10',
'urn:ogc:def:phenomenon:OGC:1.0.30:precipitation', 'PRECIPITATION_24H', 'mm');

```

Init_sos36_data.sql

```

-- phenomenon
INSERT INTO phenomenon VALUES ('urn:ogc:def:phenomenon:OGC:1.0.30:precipitation', '雨量',
'mm','numericType');
--offering
INSERT INTO offering VALUES ('PRECIPITATION','Precipitation at the present. ');
INSERT INTO offering VALUES ('PRECIPITATION_10M','Precipitation within last 10 minutes. ');
INSERT INTO offering VALUES ('PRECIPITATION_60M','Precipitation within last 60 minutes. ');
INSERT INTO offering VALUES ('PRECIPITATION_24H','Precipitation within last 24 hours. ');
--phen_off
INSERT INTO phen_off VALUES ('urn:ogc:def:phenomenon:OGC:1.0.30:precipitation','PRECIPITATION');
INSERT INTO phen_off VALUES ('urn:ogc:def:phenomenon:OGC:1.0.30:precipitation','PRECIPITATION_10M');
INSERT INTO phen_off VALUES ('urn:ogc:def:phenomenon:OGC:1.0.30:precipitation','PRECIPITATION_60M');
INSERT INTO phen_off VALUES ('urn:ogc:def:phenomenon:OGC:1.0.30:precipitation','PRECIPITATION_24H');

```

以上