Wee_json: Génération de JSON

1. Historique

V1.00 : 22/02/2022 : Première version
V1.20 : 27/02/2022 : Synchro avec « specification_json.doc » 1.20
V1.21 : 27/02/2022 : changement de place de la clé « duration »
V1.22 : 07/03/2022 : correction des JSON exemples

2. TOC

3. Objectifs

Pouvoir récupérer des données sur une période en générant un fichier Json qui pourra être intégré dans la base Climato. Le fichier Json sera stocké dans un répertoire temporaire du RaspBerry, et devra être déplacé manuellement dans le répertoire de transfert une fois validé.

Cet utilitaire sera utilisé à la suite d'un incident de transfert, ou pour la reprise d'historique de données d'une station

4. Syntaxe de lancement

Outil de récupération de données et génération de fichiers JSON. Cette génération peut être demandée à différents niveaux :

- C -> Données élémentaires
- H -> Données horaires
- D -> Données journalières
- M -> Données mensuelles
- Y -> Données annuelles
- A -> Données globales

Dans un premier temps, seuls les niveaux C, D et M seront disponibles.

Il y aura six utilitaires différents, ou six fichiers de commande shell qui lanceront un utilitaire unique, suivant les contraintes de développement rencontrées.

Syntaxe : (il faut lancer l'outil à partir du répertoire /srv)

Parametres suivants le niveau d'information souhaité :
Format « date-heure» : YYYY-MM-DDTHH:mm :ss
Format « date » : YYYY-MM-DD
Format « date-mois » : YYYY-MM-01
Format « date-année » : YYYY-01-01

Niveau	--from	--to	Période par fichier	Commentaire
C	Date-heure de début, incluse	Date-heure de fin, exclue	1 fichier par jour	--from et --to doivent être sur le même jour
H	Date-heure ronde de début, incluse	Date-heure ronde de fin, exclue.	1 fichier par jour	--from et --to doivent être sur le même jour
D	Date de début, incluse	Date de fin, exclue	1 fichier par mois	--from et --to doivent être sur le même mois
M	Date-mois de début, incluse	Date-mois de fin, inclue	1 fichier par an	--from et --to doivent être sur la méme année
Y	Date-année de début, incluse	Date-année de fin, inclue	1 fichier pour la période	
A	n/a	n/a	1 seul fichier	

Autres paramètres communs à toutes les versions des utilitaires :
- **autoload**=Y/N Load dans repertoire de transfer, ou temporaire (défaut)
- **forcereplace**=Y/N Ajoute ou pas un ***« force_replace »: True*** dans le JSON
- **verbose** Affiche des informations sur l'avancement du process

5. Répertoires et noms de fichier

C'est le paramètre « autoload » qui contrôle le répertoire destination (défaut=N) :
- autoload=Y => va dans …
- autoload=N => va dans …

Le nom de fichier respectera le format :
- Niveau C => obs-regen. « meteor ».YYYY_MM_DDTHH_mm.json
- Niveau H => agg_hour. « meteor ».YYYY_MM_DDTHH_00.json
- Niveau D => agg_day. « meteor ».YYYY_MM_DDT00 :00.json
- Niveau M => agg_month. « meteor ».YYYY_MM_01T00 :00.json
- Niveau Y => agg_year. « meteor ».YYYY_01_01T00 :00.json
- Niveau A => agg_all. « meteor ».1900_01_01T00 :00.json
  1900/01/01 = convention utilisée dans le code du serveur)

6. Stratégie de transfert vers le serveur dina.

Il est possible de remplacer la date dans le nom du fichier par la date/heure au moment de la création du fichier JSON, permettant ainsi de contrôler les fichiers à transférer à partir du time-stamp. Sinon la date de création du fichier dans le filesystem peut aussi être utilisé. L'avantage d'utiliser le format proposé, et qu'il permet de retrouver un fichier JSON dans l'archive du serveur de façon plus simple et plus rapide.

7. Stratégie de reprise de l'historique

En regroupant les fichiers suivant le niveau demandé, le nombre de fichiers pour reprendre un historique reste raisonnable.

Suivant la fiabilité des données, et le niveau de détail désiré, une des deux stratégies suivantes devra être utilisées, et probablement combinée suivant les cas :

- Reprise totale de l'historique : Si les données de la base sont correctes, alors il est conseillé de récupérer les données observation en priorité. Critères de définition de « données correctes » :
  - Pas de trou dans l'archive pour la période
  - Pas de chevauchement
  - Données non récupérées de Wunderground…
- Reprise de l'historique à partir des agrégations WeeWX. Il faudra analyser les données pour regarder à quel niveau d'agrégation cela fait le plus de sens.

Il faudra dresser la liste des stations, et par période concernée, définir le moyen de récupération des données.

8. Exemple de fichier de récupération de données observation


```
[
  {
    "meteor": "BBF015",
    "info": {
      "version": 2,
      "json_type": "C"
    },
    "data": [
      {
        "stop_dat": "2022-02-18T00:05:00",
        "duration": 5,

        "valeurs": {
          "barometer_avg": 1008.3047501,
          "barometer_d": 5,
          "barometer_max_time": "2022-02-18T00:02:02",
          "barometer_max": 1008.36480205,
          "barometer_min_time": "2022-02-18T00:04:33",
          "barometer_min": 1008.22934652,
          "barometer": 1008.22934652
        }
      },
      {
        "stop_dat": "2022-02-18T00:10:00",
        "duration": 5,

        "valeurs": {
          "barometer_avg": 1008.3047501,
          "barometer_d": 5,
          "barometer_max_time": "2022-02-18T00:06:26",
```

```
            "barometer_max": 1008.36480205,
            "barometer_min_time": "2022-02-18T00:08:45",
            "barometer_min": 1008.22934652,
            "barometer": 1008.22934652
          }
        }
      ],
      "extremes":
      {
        "level": "D",
        "barometer_max_time": "2022-02-18T18:33:26",
        "barometer_max": 1008.36480205,
        "barometer_min_time": "2022-02-18T00:07:23",
        "barometer_min": 1007.22934652
      }
    }
]
```


9. Exemple de récupération de données agrégées heure


```
[
  {
    "meteor": "BBF015",
    "info":
    {
       "version": 2,
       "json_type": "H",
       "dat_day": "2022-02-18T00:00:00"
    },
    "data": [
       {
         "start_dat": "2022-02-18T00:05:00",

         "valeurs":
         {
           "barometer_s": 60498.285006,
           "barometer_d": 60
         }
       },
       {
         "start_dat": "2022-02-18T01:05:00",

         "valeurs":
         {
           "barometer_s": 60498.285006,
           "barometer_d": 60
         }
       }
      ],
```

```
     "extremes":
     {
       "level": "D",
       "barometer_max_time": "2022-02-18T18:33:26",
       "barometer_max": 1008.36480205,
       "barometer_min_time": "2022-02-18T00:07:23",
       "barometer_min": 1007.22934652
     }
   }
]
```


10. Exemple de récupération de données agrégée jour et supérieur


```
[
  {
    "meteor": "BBF015",
    "info":
    {
       "version": 2,
       "json_type": "D"
    },
    "data":
    [
        {
           "start_dat": "2021-01-01T00:05:00",
           "valeurs":
           {
              "barometer_s": 1461186.02319,
              "barometer_d": 1440,

              "barometer_max_time": "2021-01-01T00:02:03",
              "barometer_max": 1016.76304468,
              "barometer_min_time": "2021-01-01T22:54:03",
              "barometer_min": 1013.30892876,

              "barometer_omm_s": 1461217.0086432,
              "barometer_omm_d": 1440
           }
        },
        {
           "start_dat": "2021-01-02T00:05:00",
           "valeurs":
           {

              "barometer_s": 1462069.53186,
              "barometer_d": 1440,
              "barometer_max_time": "2021-01-02T00:05:00",
              "barometer_max": 1016.35667811,
              "barometer_min_time": "2021-01-02T15:49:03",
              "barometer_min": 1013.57983982,
```

```
            "barometer_omm_s": 1462048.0282944,
            "barometer_omm_d": 1440
        }
    }
  ]
 }
]
```