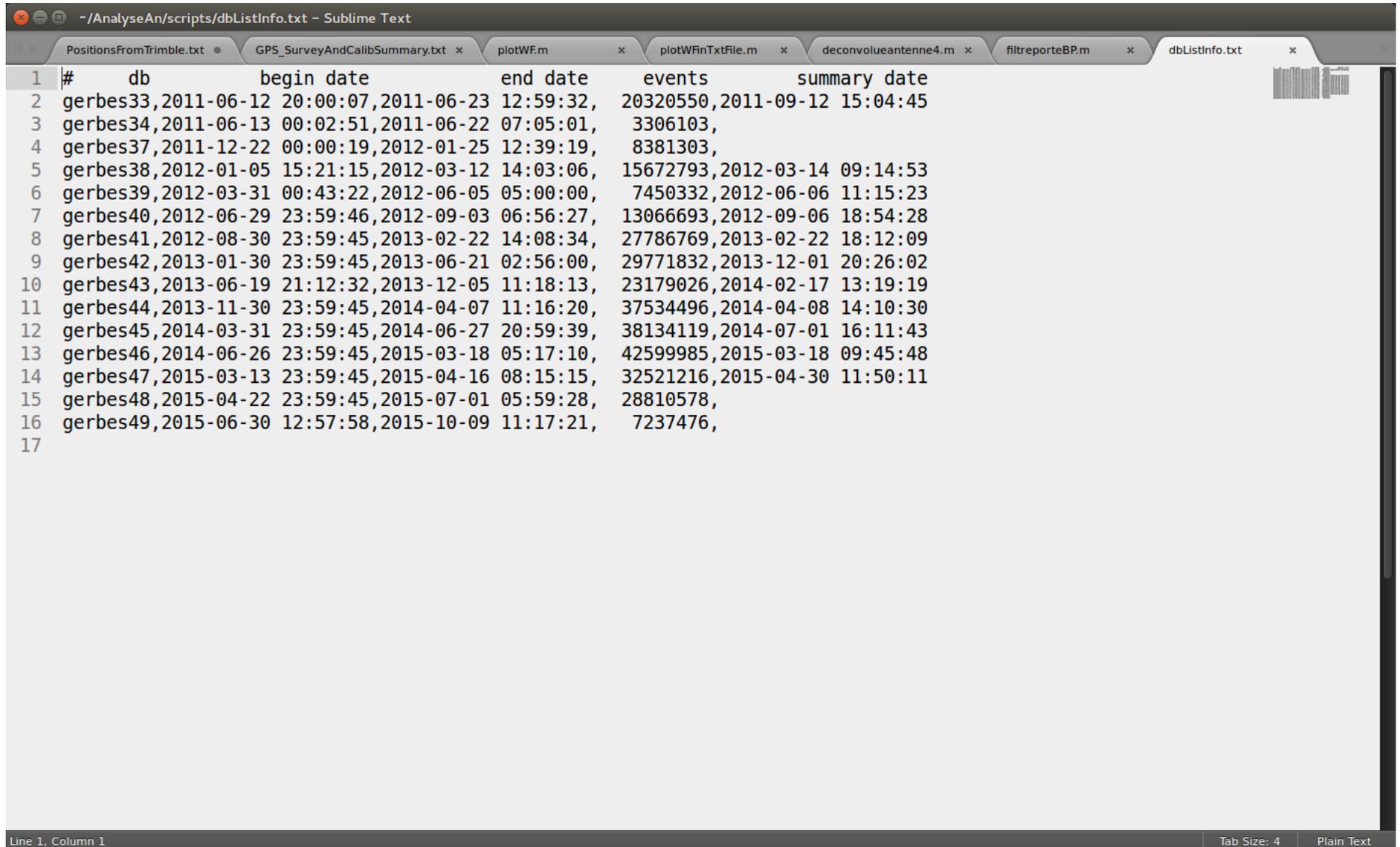


Données

- Les données sont écrites en continue sur un disque réseau.
- Pour les instruments principaux, elles sont versées dans une base de données de type relationnel.
- Beaucoup de données produites (stations autonomes) et donc morcellement par périodes (de 1 mois a quelques mois).
- Les informations sont réparties dans des tables qui facilitent des recherches sélectives

Les bases de données actuelles



```
~/AnalyseAn/scripts/dbListInfo.txt - Sublime Text
PositionsFromTrimble.txt x GPS_SurveyAndCalibSummary.txt x plotWF.m x plotWFinTxtFile.m x deconvolueantenne4.m x filtreporteBP.m x dbListInfo.txt x
1 # db begin date end date events summary date
2 gerbes33,2011-06-12 20:00:07,2011-06-23 12:59:32, 20320550,2011-09-12 15:04:45
3 gerbes34,2011-06-13 00:02:51,2011-06-22 07:05:01, 3306103,
4 gerbes37,2011-12-22 00:00:19,2012-01-25 12:39:19, 8381303,
5 gerbes38,2012-01-05 15:21:15,2012-03-12 14:03:06, 15672793,2012-03-14 09:14:53
6 gerbes39,2012-03-31 00:43:22,2012-06-05 05:00:00, 7450332,2012-06-06 11:15:23
7 gerbes40,2012-06-29 23:59:46,2012-09-03 06:56:27, 13066693,2012-09-06 18:54:28
8 gerbes41,2012-08-30 23:59:45,2013-02-22 14:08:34, 27786769,2013-02-22 18:12:09
9 gerbes42,2013-01-30 23:59:45,2013-06-21 02:56:00, 29771832,2013-12-01 20:26:02
10 gerbes43,2013-06-19 21:12:32,2013-12-05 11:18:13, 23179026,2014-02-17 13:19:19
11 gerbes44,2013-11-30 23:59:45,2014-04-07 11:16:20, 37534496,2014-04-08 14:10:30
12 gerbes45,2014-03-31 23:59:45,2014-06-27 20:59:39, 38134119,2014-07-01 16:11:43
13 gerbes46,2014-06-26 23:59:45,2015-03-18 05:17:10, 42599985,2015-03-18 09:45:48
14 gerbes47,2015-03-13 23:59:45,2015-04-16 08:15:15, 32521216,2015-04-30 11:50:11
15 gerbes48,2015-04-22 23:59:45,2015-07-01 05:59:28, 28810578,
16 gerbes49,2015-06-30 12:57:58,2015-10-09 11:17:21, 7237476,
17
```

Line 1, Column 1 Tab Size: 4 Plain Text

Les tables contenant les données

The image shows two windows from a database administration tool. The left window, titled 'FlameRobin Database Admin', displays a tree view of databases. The 'gerbes48' database is expanded, showing various objects including 'Tables (14)'. The 'EVENTS' table is highlighted. The right window, titled 'gerbes48 - EVENTS', shows the table's structure with columns, data types, and nullability.

FlameRobin Database Admin

- gerbes44
- gerbes45
- gerbes43
- gerbes47
- gerbes34
- gerbes48
 - Domains (3)
 - Exceptions
 - Functions (8)
 - Generators (8)
 - Procedures (19)
 - Roles (1)
 - System tables (42)
 - Tables (14)
 - AUTOS_BY_HOUR
 - DEVICES
 - EVENTS**
 - EVENTS_BY_HOUR
 - EVENTS_BY_MINUTE
 - EVENTS_BY_SECOND
 - MULTIPLES
 - RAWFILES
 - SA_POSITIONS
 - SC_DATA
 - SINGLES
 - SOURCES
 - SUMMARY
 - WAVEFORMS
 - Triggers (3)
 - Views (7)
- gerbes49

gerbes48 - EVENTS

EVENTS: Table

Summary | [Constraints](#) | [Triggers](#) | [Indices](#) | [Privileges](#) | [Dependencies](#) | [DDL](#)

EVENTS
Owner: GERBES
No description [\[edit\]](#)

Field	Type	Not Null
EVT_ID	bigint (INDEXS)	✓
FILE_ID	bigint (INDEXS)	✓
EVT_TIME	timestamp	
GPS_TIME	timestamp	
GPS_NANO	integer	
TRIGGER_TYPE	char(4)	
CONTROLLER_STATUS	smallint	
TRIGGER_STATUS	smallint	
GPS_STATUS	smallint	
SA_IDENT	smallint	
EVENT_COUNTER	integer	
CONTROLLER_MODE	smallint	
MJD	double precision	
SOURCE	char(3)	
LON	double precision	
LAT	double precision	
ALT	double precision	
TRIGGER_POSITION	smallint	
CRATE_POWER	smallint	
CRATE_TEMPERATURE	smallint	
TRIGNUM	smallint	
TBS	char(1)	
PEDESTALS	char(1)	

[Add field](#) | [Reorder fields](#) | [Drop fields](#) | [Generate rebuild script](#)

codalema@coda4.obs-nancay.fr:gerbes48 (NONE)

L'accès aux bases de données

- Les bases sont stockées et accessibles depuis a travers une machine dédiée (coda4.obs-nancay.fr).
- La communication avec cette machine est autorisée depuis des adresses IP déclarées.
- Il faut un outil pour effectuer des requêtes. Plusieurs solutions possibles :
 - Outils de diagnostic des bases (plutôt pour faire de la gestion que des requêtes)
 - Macros/commandes écrites pour faire des requêtes dans des logiciels de traitement de données (IDL, Matlab, Python)
 - Codes spécifiques pour extraire et sauvegarder les données dans un format particulier(Python, Shell, code compilé)

Un exemple de requête via Python

```
codalema@codadaqa: /data/An/branches/v2.1/Mn/scripts
'KB020', 'NK022', 'QH023', 'FI025', 'FG026', 'DF027', 'ED028', 'BB029', \
'DB030', 'EB031', 'PB033', 'NI034', 'HL035', 'OC036', 'GB037', 'MC250', \
]
treq=0
tplo=0
tfft=0
tsta=time.clock()
for iSA in range(0, len(idSA)):
#for iSA in range(0, 2):
    tic=time.clock()
    # print 'tic = %f \n' %(tic,)
    SA=idSA[iSA]
    fileout=path+"SA"+SA+"_tempo.bin"
    fileouth=path+"SA"+SA+"_tempo.txt"
    sel_evt_id="select w.volts, e.evt_time, e.gps_nano, e.trigger_type, w.threshold, w.dev_id from events e left join waveforms w on (w.evt_id=e.evt_id) where (e.sa_ident="+SA+" AND e.evt_time
= (select max(evt_time) from EVENTS where trigger_type<>'AUTO' and sa_ident="+SA+")) order by w.dev_id asc"

    con = kib.connect(
        dsn='coda4.obs-nancay.fr:gerbes43',
        user='codalema', password='piginthecorn', role='SIMPLEUSER')
    cur = con.cursor();
    cur.set_type_trans_out({'BLOB': {'mode': 'stream'}})
    cur.execute(sel_evt_id);
    rows=cur.fetchmany(2)
    fout = open(fileout, 'w')
    fouth = open(fileouth, 'w')
    for row in rows:
        readerB=row[0]
        egpst=row[1]
        gpsnano=row[2]
        trig=row[3]
        thres=row[4]
        devid=row[5]
        ww=readerB.read()
        fout.write(ww)
        gpsmicro=gpsnano/1000
        fouth.write('%s %03d %04d %s\n' %(egpst.strftime('%Y %m %d %H %M %S'), devid, thres, trig,))
    fout.close()
    fouth.close()
    cur.close();
    tac=time.clock()
    treq=treq+tac-tic
    num_lon = 2560
    num_lat = 2
    if 'CHN2' in trig:
        trigChain='NS'
    if 'CHN1' in trig:
        trigChain='EW'
    if 'AUTO' in trig:
        trigChain='1W'
    if 'EXT' in trig:
        trigChain='Ext.'
    print '%2d %s %s %s' %(iSA, stations[iSA], trigChain, egpst.strftime('%Y-%m-%d %H:%M:%S'),)
    if os.path.getsize(fileout)==20480:
        tic=time.clock()
        data=N.fromfile(fileout, 'float32')
        data = N.reshape(data, (num_lat, num_lon))
        ft=N.fft.fft(N.transpose(data), None, 0)
```