On 20140210

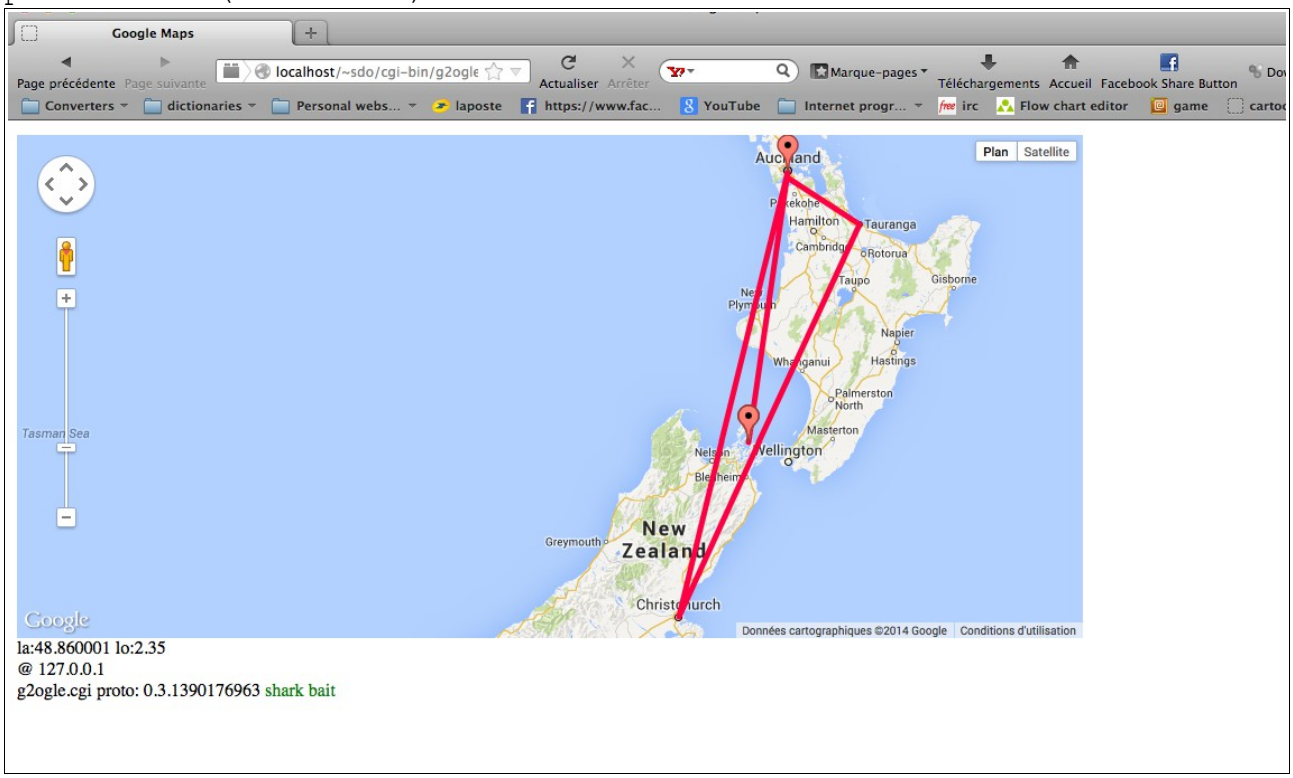
Time change scripts changes bugs persist

Introduction

During tests period with G2 (that's the second version of the map script with google map) path was printed. We had several point on the path that indicated where the user watched the album. A ine between each mark was drawn. Alas the line was drawn but not the mark.

The bug

During tests period with G2 a bug was noticed see Introduction in the page. We passed from G2 to G3 (third version) and bug was persistant (see below).



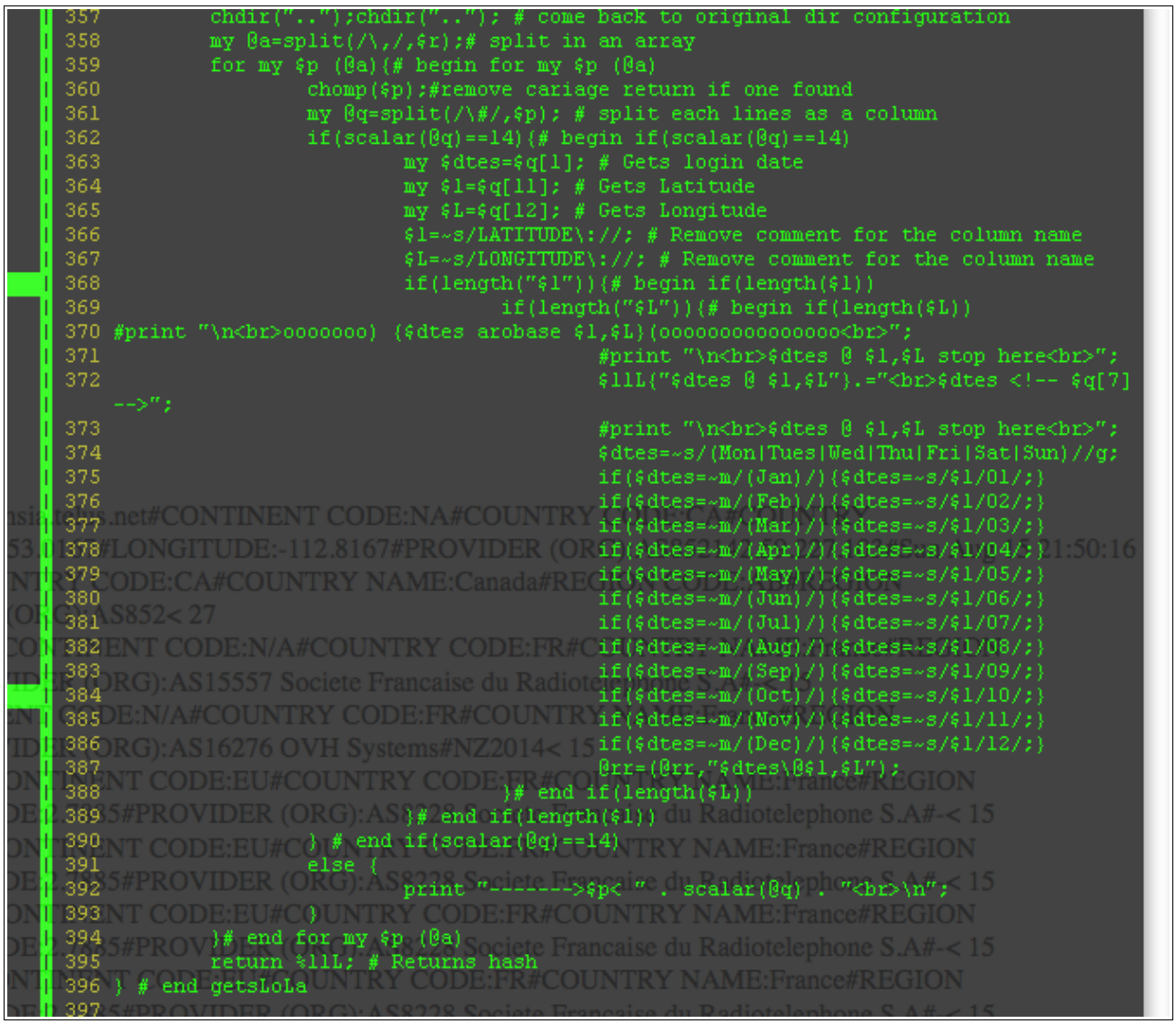
Version 2

Version 3

If we compare we have the same number of markers for New Zealand. In the second version we have lines (not implemented yet in the third version. In the second version we have several angles. At each angles should be a Mark in New Zealand. That's the bug.

The bug test to debug

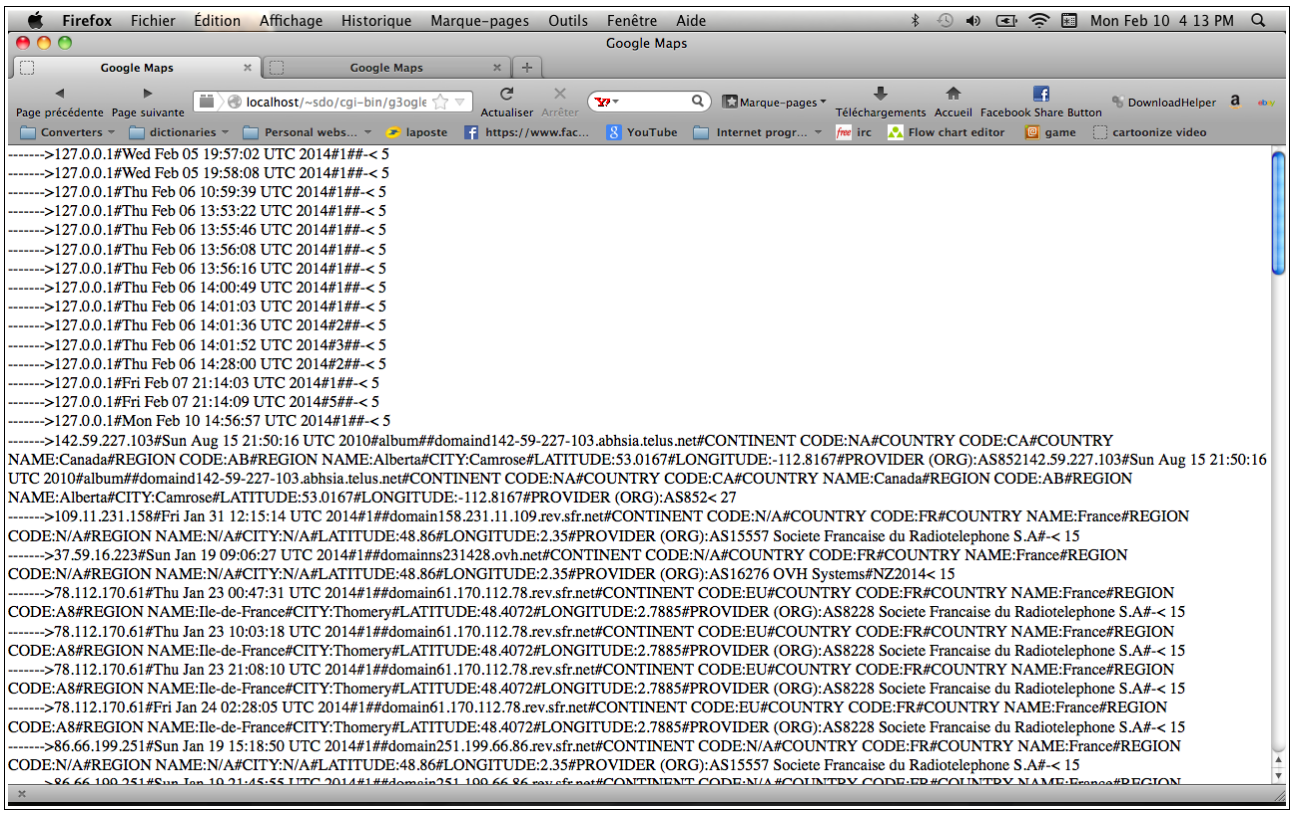
To build the mat it is needed to collect data recorded.
See picture below:



```
357         chdir("..");chdir(".."); # come back to original dir configuration
358         my @a=split(/\,/,$r);# split in an array
359         for my $p (@a){# begin for my $p (@a)
360                 chomp($p);#remove cariage return if one found
361                 my @q=split(/\#/,$p); # split each lines as a column
362                 if(scalar(@q)==14){# begin if(scalar(@q)==14)
363                         my $dtes=$q[1]; # Gets login date
364                         my $l=$q[11]; # Gets Latitude
365                         my $L=$q[12]; # Gets Longitude
366                         $l=~s/LATITUDE\://; # Remove comment for the column name
367                         $L=~s/LONGITUDE\://; # Remove comment for the column name
368                         if(length("$l")){# begin if(length($l))
369                                 if(length("$L")){# begin if(length($L))
370 #print "\n<br>ooooooo) {$dtes arobase $l,$L}(ooooooooooooooo<br>";
371                                 #print "\n<br>$dtes @ $l,$L stop here<br>";
372                                 $llL{"$dtes @ $l,$L"}.="<br>$dtes <!-- $q[7]
    -->";
373                                 #print "\n<br>$dtes @ $l,$L stop here<br>";
374                                 $dtes=~s/(Mon|Tues|Wed|Thu|Fri|Sat|Sun)//g;
375                                 if($dtes=~m/(Jan)/){$dtes=~s/$l/01/;}
376                                 if($dtes=~m/(Feb)/){$dtes=~s/$l/02/;}
377                                 if($dtes=~m/(Mar)/){$dtes=~s/$l/03/;}
378                                 if($dtes=~m/(Apr)/){$dtes=~s/$l/04/;}
379                                 if($dtes=~m/(May)/){$dtes=~s/$l/05/;}
380                                 if($dtes=~m/(Jun)/){$dtes=~s/$l/06/;}
381                                 if($dtes=~m/(Jul)/){$dtes=~s/$l/07/;}
382                                 if($dtes=~m/(Aug)/){$dtes=~s/$l/08/;}
383                                 if($dtes=~m/(Sep)/){$dtes=~s/$l/09/;}
384                                 if($dtes=~m/(Oct)/){$dtes=~s/$l/10/;}
385                                 if($dtes=~m/(Nov)/){$dtes=~s/$l/11/;}
386                                 if($dtes=~m/(Dec)/){$dtes=~s/$l/12/;}
387                                 @rr=(@rr,"$dtes\@$l,$L");
388                         }# end if(length($L))
389                 }# end if(length($l))
390         } # end if(scalar(@q)==14)
391         else {
392                 print "-------->$p< " . scalar(@q) . "<br>\n";
393         }
394     }# end for my $p (@a)
395     return %llL; # Returns hash
396 } # end getsLoLa
397
```

Printing of a piece of code of g3ogle.cgi

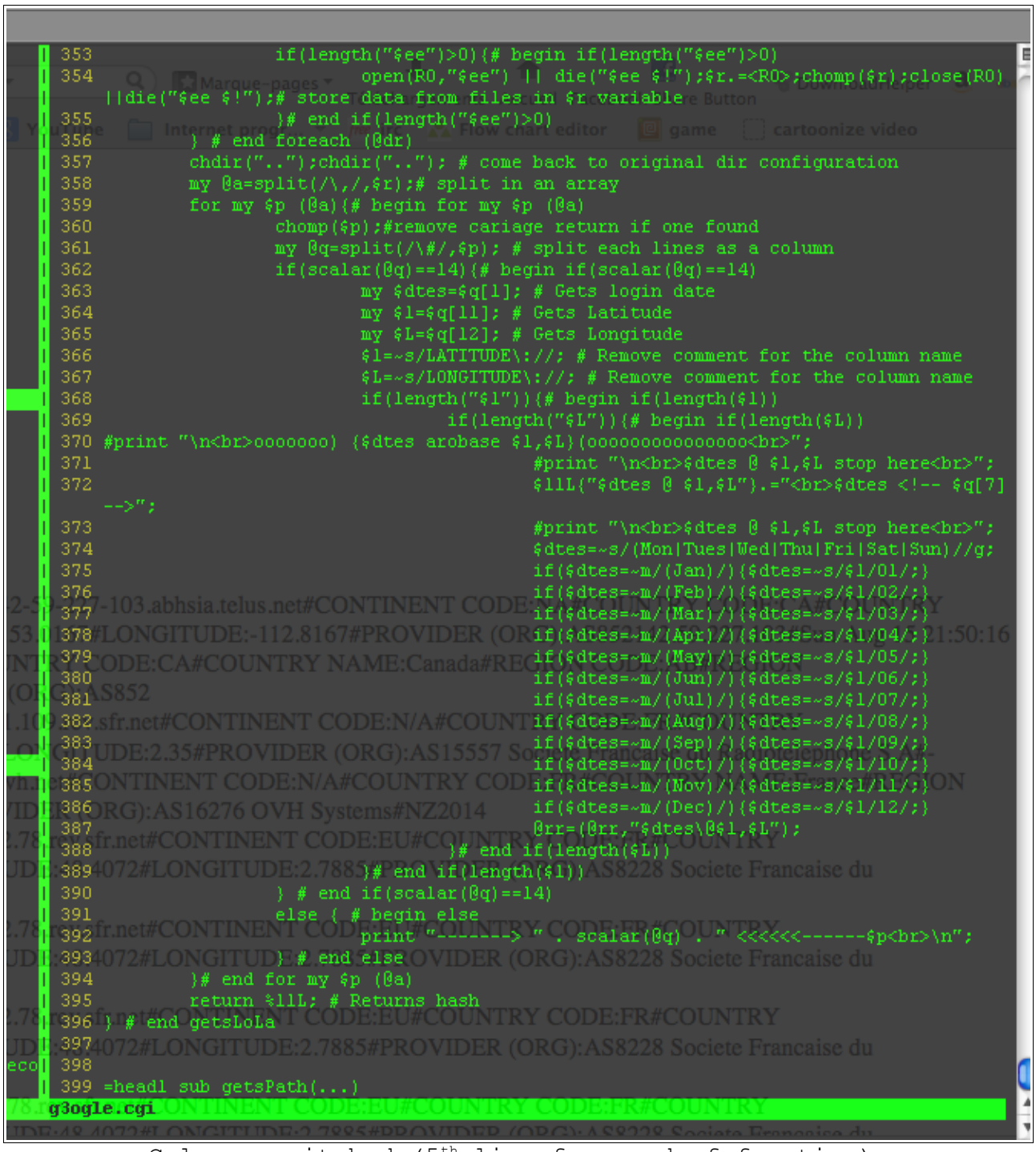
An alternative was added to print what is not printed (5th line from end of function).
Now a printing of the execution:



Printing of the execution

To summup: quite a lot of data are removed.

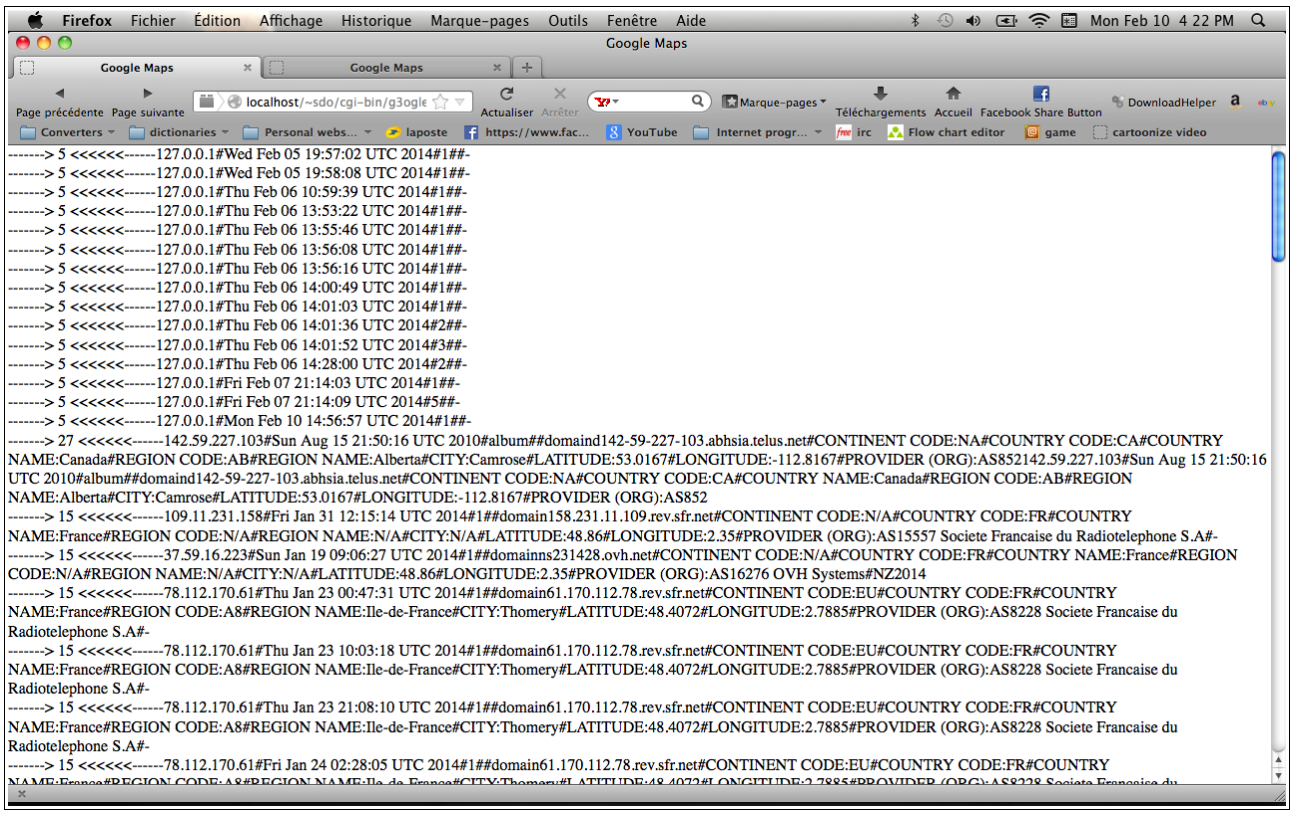
We switched columns:



```
353                 if(length("$ee")>0){# begin if(length("$ee")>0)
354                         open(RO,"$ee") || die("$ee $!");$r.=<RO>;chomp($r);close(RO)
    ||die("$ee $!");# store data from files in $r variable
355                 }# end if(length("$ee")>0)
356         } # end foreach (@dr)
357         chdir("..");chdir(".."); # come back to original dir configuration
358         my @a=split(/\,/,$r);# split in an array
359         for my $p (@a){# begin for my $p (@a)
360                 chomp($p);#remove cariage return if one found
361                 my @q=split(/\#/,$p); # split each lines as a column
362                 if(scalar(@q)==14){# begin if(scalar(@q)==14)
363                         my $dtes=$q[1]; # Gets login date
364                         my $l=$q[11]; # Gets Latitude
365                         my $L=$q[12]; # Gets Longitude
366                         $l=~s/LATITUDE\://; # Remove comment for the column name
367                         $L=~s/LONGITUDE\://; # Remove comment for the column name
368                         if(length("$l")){# begin if(length($l))
369                                 if(length("$L")){# begin if(length($L))
370 #print "\n<br>ooooooo) ($dtes arobase $l,$L)(ooooooooooooooo<br>";
371                                         #print "\n<br>$dtes @ $l,$L stop here<br>";
372                                         $llL{"$dtes @ $l,$L"}.="<br>$dtes <!-- $q[7]
    -->";
373                                         #print "\n<br>$dtes @ $l,$L stop here<br>";
374                                         $dtes=~s/(Mon|Tues|Wed|Thu|Fri|Sat|Sun)//g;
375                                         if($dtes=~m/(Jan)/){$dtes=~s/$1/01/;}
376                                         if($dtes=~m/(Feb)/){$dtes=~s/$1/02/;}
377                                         if($dtes=~m/(Mar)/){$dtes=~s/$1/03/;}
378                                         if($dtes=~m/(Apr)/){$dtes=~s/$1/04/;}
379                                         if($dtes=~m/(May)/){$dtes=~s/$1/05/;}
380                                         if($dtes=~m/(Jun)/){$dtes=~s/$1/06/;}
381                                         if($dtes=~m/(Jul)/){$dtes=~s/$1/07/;}
382                                         if($dtes=~m/(Aug)/){$dtes=~s/$1/08/;}
383                                         if($dtes=~m/(Sep)/){$dtes=~s/$1/09/;}
384                                         if($dtes=~m/(Oct)/){$dtes=~s/$1/10/;}
385                                         if($dtes=~m/(Nov)/){$dtes=~s/$1/11/;}
386                                         if($dtes=~m/(Dec)/){$dtes=~s/$1/12/;}
387                                         @rr=(@rr,"$dtes\@$l,$L");
388                                 }# end if(length($L))
389                         }# end if(length($l))
390                 } # end if(scalar(@q)==14)
391                 else { # begin else
392                         print "-------> " . scalar(@q) . " <<<<<<------$p<br>\n";
393                 }# end else
394         }# end for my $p (@a)
395         return %llL; # Returns hash
396 } # end getsLoLa
397
398
399 =headl sub getsPath(...)
g3ogle.cgi
```

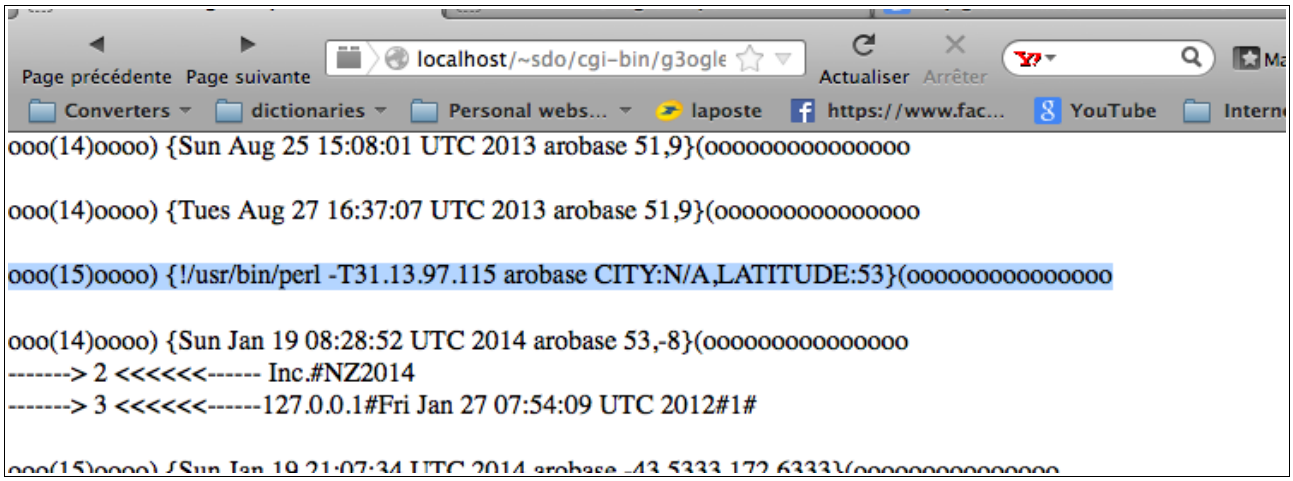
Columns switched (5th line from end of function).

The printing will give:



Printing for debuging

After several modifications in the script that prints debug information this was obtained:



The blue revealed the bug (we have some more in other parts)

this prints the line field values.
Another print out but with more data:

ooo(14)oooo) {Sun Aug 25 15:08:01 UTC 2013 arobase 51,9}(oooooooooooooo>

ooo(14)oooo) {Tues Aug 27 16:37:07 UTC 2013 arobase 51,9}(oooooooooooooo>

ooo(15)oooo) {!/usr/bin/perl -T31.13.97.115 arobase CITY:N/A,LATITUDE:53}(oooooooooooooo>---->#!/usr/bin/perl -T31.13.97.115#Mon Aug 12 04:00:23 UTC 2013#1##domain31.13.97.115#CONTINENT CODE:N/A#COUNTRY CODE:IE#COUNTRY NAME:Ireland#REGION CODE:N/A#REGION NAME:N/A#CITY:N/A#LATITUDE:53#LONGITUDE:-8#PROVIDER (ORG):N/A

ooo(14)oooo) {Sun Jan 19 08:28:52 UTC 2014 arobase 53,-8}(oooooooooooooo>
-------> 2 <<<<<<------ Inc.#NZ2014
-------> 3 <<<<<<------127.0.0.1#Fri Jan 27 07:54:09 UTC 2012#1#

extra print out with more field values

There is a path with Perl call that must not be there. Best guess: from this file all data are shifted. Now lets see why.

The bug solved

To print all marks data are stored somewhere. Due to many transforms in the time some perl script were written. Alas the program that parsed the directory took care of theses files. As it contained data that are used as separator in the script it jammed printings.

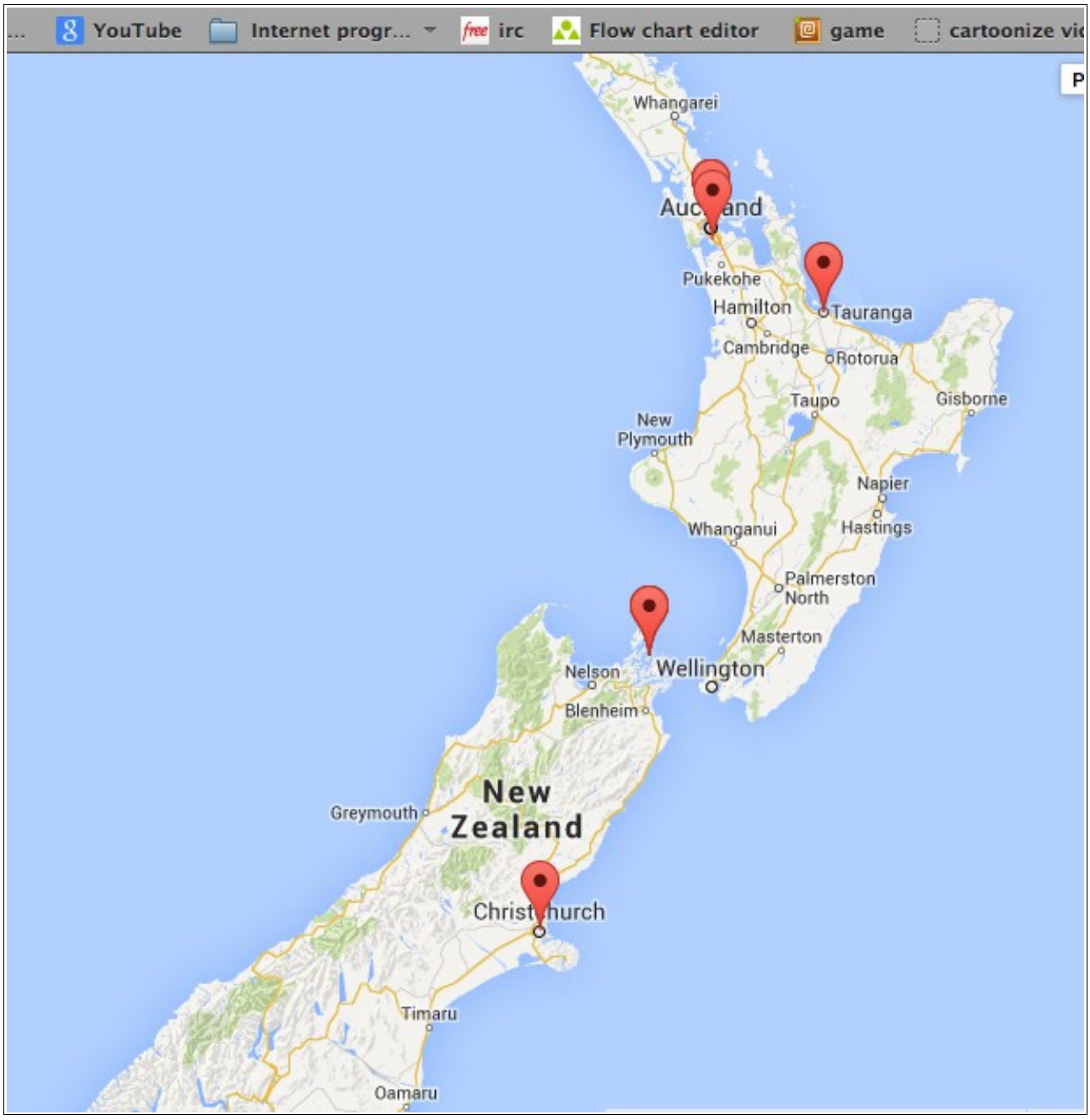
The perl script modified:

```
346         my $r=();# store content of each file
347         opendir(ARD,".") || die(". $!");# open current directory
348         my @dr= grep { $_ ne '.' and $_ ne '..' and $_ !~ m/pl$/i and $_ !~ m/cgi$/i
    } readdir(ARD);# parse current directory
349         closedir(ARD) || die(". $!");# close directory
350
```

Lines modified

The blue lines are where the transforms occured.

Now we have:



All data were in a directory. We parsed the directory but script file were in it due to make modifications. They contained separators, they jammed the printings on the screen (see below).

ooo(14)oooo) {Sun Aug 25 15:08:01 UTC 2013 arobase 51,9}(oooooooooooooo>

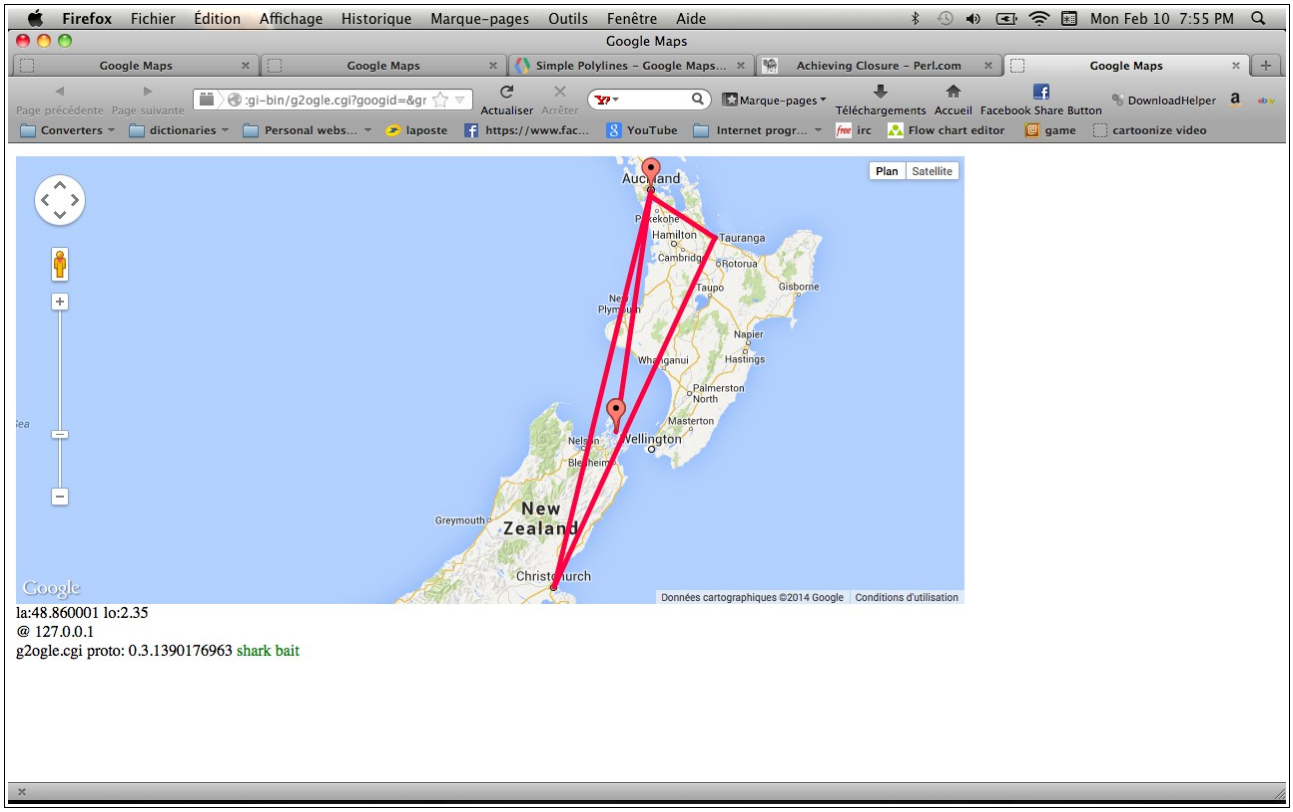
ooo(14)oooo) {Tues Aug 27 16:37:07 UTC 2013 arobase 51,9}(oooooooooooooo>

ooo(15)oooo) {!/usr/bin/perl -T31.13.97.115 arobase CITY:N/A,LATITUDE:53}(oooooooooooooo>---->#!/usr/bin/perl -T31.13.97.115#Mon Aug 12 04:00:23 UTC 2013#1##domain31.13.97.115#CONTINENT CODE:N/A#COUNTRY CODE:IE#COUNTRY NAME:Ireland#REGION CODE:N/A#REGION NAME:N/A#CITY:N/A#LATITUDE:53#LONGITUDE:-8#PROVIDER (ORG):N/A

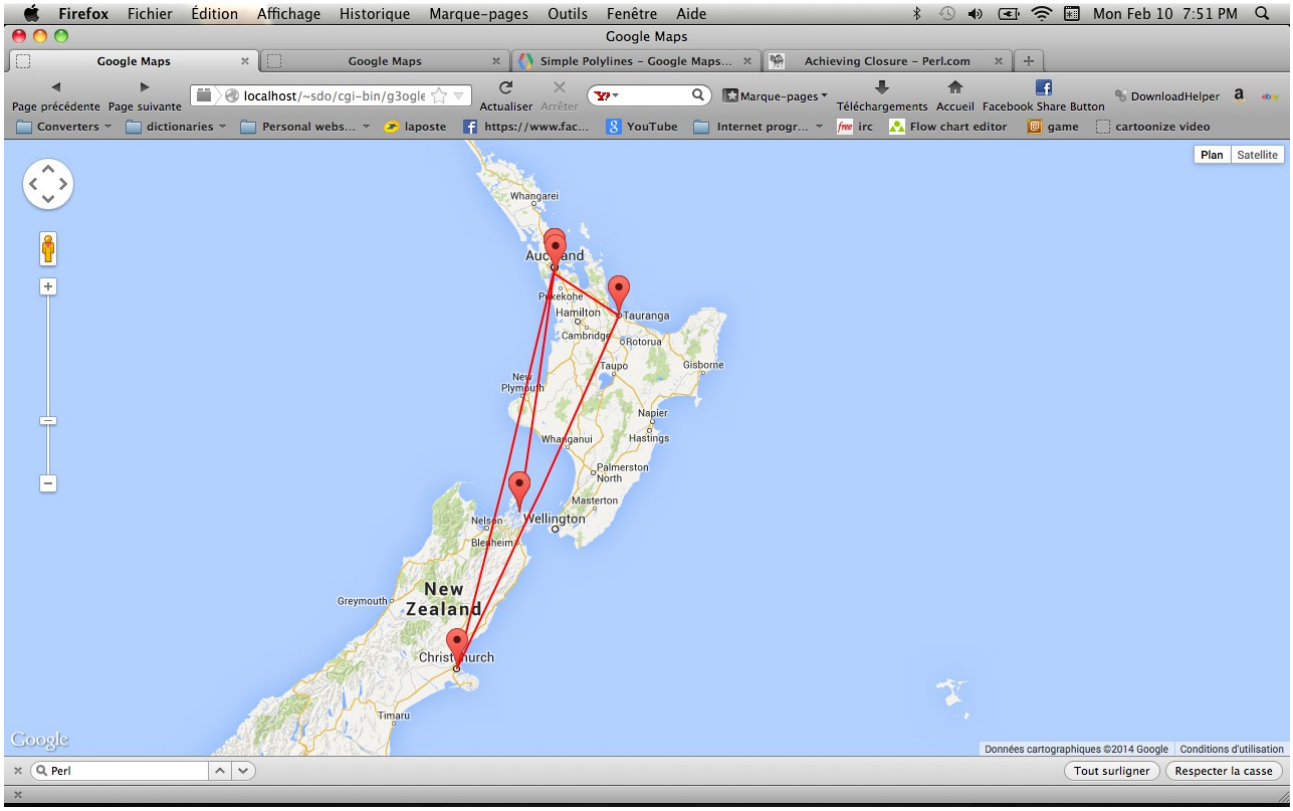
ooo(14)oooo) {Sun Jan 19 08:28:52 UTC 2014 arobase 53,-8}(oooooooooooooo>
-------> 2 <<<<<<------ Inc.#NZ2014
-------> 3 <<<<<<------127.0.0.1#Fri Jan 27 07:54:09 UTC 2012#1#

Fields jammed

After solving the problem we have gotten:



Old printout G2



New printout G3