TraceAdjust Overview

Norman Dunbar

20th March 2017

Contents

TraceAdjust

TraceAdjust is a new utility that will read an Oracle trace file and look for any lines containing the text "tim=". This indicates a number of microseconds offset from "some epoch" depending on the server/Operating System in use. On Windows Azure servers, it *appears* to be from the last time the server was rebooted. Hmm - useful or what?

The code looks for one other type of line, a timestamp, such as:

```
*** 2017-03-13 09:23:21.767
```

When it finds a line of this type, a couple of things happen:

- Internally, the base timestamp is adjusted to the new actual timestamp, including fractions of a second;
- The next "tim=" value is synchronised with the new timestamp;
- The following is then written to the output trace file (or stdout as appropriate):

```
*** 2017-03-13 09:23:21.767 *** TraceAdjust: Base Timestamp Adjusted to 'Mon Mar
13 09:23:21 2017'
```

I have noticed that some fractions of a second can lose a single microsecond when totalling. This is most likely down to the accuracy of a C++ `float` data type.

As TraceAdjust passes through the trace file, every line is read and written out. If a line has a "tim=" within it, it gets modified by adding the following extra trace fields:

- delta=nnnn
- dslt=nnnnnn
- local='yyyy Mon dd hh24:mi:ss.ffffff'

In addition to the above, the value of the "tim=" field itself is adjusted to show seconds and fractions of seconds, instead of simply a large unwieldy multi-digit number.

Where:

- delta is the difference, in microseconds, between the current line's "tim=" value, and the previously read "tim=" value. If we have just had a resynchronisation, this will be zero for the first "tim=" found after the timestamp line.
- dslt is the 'delta since last timestamp' and is a running total of micro seconds accumulated since the previous timestamp line. Again, if we have just had a resynchronisation, this will be the fractions of a second extracted from the just read timestamp line.
- local is a timestamp in a human readable format. It has microsecond precision.

The following is an example of an input line and an output line:

```
PARSE #4474286416:c=0,e=418,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1030574627220
```

Which becomes, all on one line:

```
PARSE #4474286416:c=0,e=418,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=0,tim=1030574.627220,
      delta=-1,dslt=768371,local='2017 Mar 13 09:23:21.768371'
```

Note: I've had to split the output line to get it to fit on a PDF page.

You may note that a delta can be negative! This usually occurs after the PARSING IN CURSOR line and affects the following PARSE line for that cursor. WAITs etc don't show these negatives. A few more examples are:

```
EXEC #4474286416:c=0,e=1031,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=2853959010,tim=1030574628420
FETCH #4474286416:c=0,e=13,p=0,cr=3,cu=0,mis=0,r=0,dep=1,og=4,plh=2853959010,tim=1030574628457
...
CLOSE #4474286416:c=0,e=2,dep=1,type=3,tim=1030574628514
```

Which becomes the following after processing:

```
EXEC #4474286416:c=0,e=1031,p=0,cr=0,cu=0,mis=1,r=0,dep=1,og=4,plh=2853959010,tim=1030574.628420,
     delta=1200,dslt=769571,local='2017 Mar 13 09:23:21.769571'
FETCH #4474286416:c=0,e=13,p=0,cr=3,cu=0,mis=0,r=0,dep=1,og=4,plh=2853959010,tim=1030574.628457,
      delta=37,dslt=769608,local='2017 Mar 13 09:23:21.769608'
...
CLOSE #4474286416:c=0,e=2,dep=1,type=3,tim=1030574.628514,
      delta=57,dslt=769665,local='2017 Mar 13 09:23:21.769665'
```

Hopefully, the additional fields make your trace file browsing experience a little more "human". They certainly help mine. While many tools exist to help analyse a trace file, sometimes you need to get down and dirty in the raw data.

Download the Source

Go to https://github.com/NormanDunbar/TraceAdjust and click the clone or download button. Choose the option to download a zip file.

Save it somewhere safe, I use my `SourceCode` folder, and extract it. This will create a new folder named `TraceAdjust-master` - you can remove the `-master` bit if you wish, however, the following instructions assume that you didn't.

Compile the Source

On Linux

```
cd SourceCode
cd TraceAdjust-master
mkdir bin
g++ -o bin/TraceAdjust -std=c++11 src/TraceAdjust.cpp
```

On Windows

Visual C++/Visual Studio

Just don't! You don't need the grief. Honestly, try it if you want to. I haven't.

Borland 10.1 compiler

```
cd SourceCode
cd TraceAdjust-master
mkdir bin
bcc32c -o bin\TraceAdjust.exe src\*.cpp
```

CodeBlocks IDE

There is a project file in the `SourceCode\TraceAdjust-master\` sub-folder, named `TraceAdjust.cbp`. Open that and select `Build->Build` or press CTRL-F9 to do the same. The executable will be found in `SourceCode\TraceAdjust-master\TraceAdjust\bin\Release` when it has completed.

Command Line Options

Executing TraceAdjust is easy. At a minimum it requires a single parameter, the name of a trace file. This can be a relative or absolute path as desired. Output will be to the screen in this case.

```
TraceAdjust trace_file
```

For best results, redirect `stdout` to a new trace file to make the output a tad more usable.

```
TraceAdjust trace_file > new_trace_file
```

TraceAdjust will create the new trace file which will be a complete copy of the input file, with a few extra lines thrown in - the indication that the base timestamp has been adjusted when timestamp lines are found, plus all the new trace fields mentioned above.

Other than inserting a decimal point into the existing "tim=" value, to make it a little more readable, none of the existing trace data are *changed*, the changes simply *add* additional data to the trace file, giving - hopefully - a more usable trace.

Need a Free C++ Compiler?

https://www.embarcadero.com/free-tools is the place to look for one. It's 32 bit and modern. It runs perfectly well on 64 bit Windows too.

You will need to sign up, but other than a few special offers, and a couple of training course advisory emails, you won't get too much hassle. And it's worth it for one of the finest Windows C/C++ compilers for free.

- Download it.
- Unzip it.
- Add the bin folder to your path.
- Use it and love it! ;-)

Have fun. Norm.