

Appendix G  
Test Report for 100 Test Runs  
for the Hazelcast Prototype

Appendix for Master Thesis:  
*Different Paths to High Availability  
by Introducing Redundancy  
in a Distributed SCADA System*

Morten Andersen, Årskort 19941458  
Master of Information Technology, Software Development  
Department of Computer Science, University of Aarhus  
8200 Århus N, Denmark

June 14, 2011

**Contents**

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Comments to the Graphs</b>                       | <b>2</b> |
| <b>2</b> | <b>Network Measurements</b>                         | <b>3</b> |
| <b>3</b> | <b>Measurements for the 100 Prototype Test Runs</b> | <b>5</b> |
| 3.1      | Crash After Test Run 17 . . . . .                   | 6        |

## 1 Comments to the Graphs

This appendix contains the detailed test graphs for all 100 test runs for the Hazelcast based prototype. The unprocessed data that these graphs are created from, are included in the source archive for the thesis in the `test-runs/hazelcast-timings` folder.

Figure 1 below is a summary of all 100 tests in one graph, showing the *maximum* observed client data age for each of the 100 test runs as the solid line, and the *mean value over the max values* as the dashed line.



Figure 1: Aggregated max and mean of the max values for all 100 test runs.

Figure 2 shows the same data as figure 1, but is not capped at 6000 ms for the y-axis, showing in full extend how badly the prototype performed at some of the failover situations.

The two figures shows that only 20% of the test runs was successful in that the maximum observed value for the test run was below the 5 seconds deadline (QAS1). For the remaining 80% percent of the tests there were at least one failover situation that failed in fulfilling QAS1.

Based on these aggregated graphs it is relevant to study the graphs for the individual test runs in chapter 3 to search for a pattern in the failing tests.

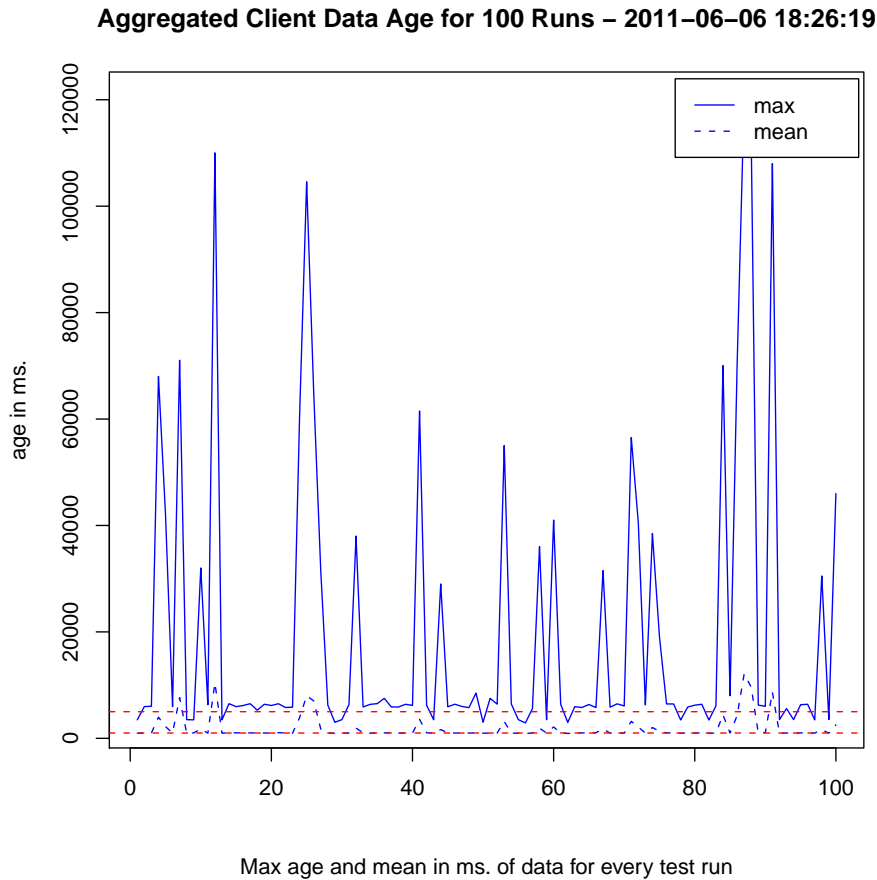


Figure 2: Aggregated max and mean of the max values for all 100 test runs.

## 2 Network Measurements

The two figures on the following page shows the network measurements of the nodes in the test system before and after the 100 test runs.

It is seen that the network layout as reported by the hop length between the nodes are stable during the test. This verifies that no live migration of running systems happens in the Amazon EC2 data center.

With respect to the network round trip times (rtt) 4 notable outliers are seen in the test after the tests was performed. These are the 4 white spots in figure 4, with rtt crossing 100 ms. This is somewhat surprising and shows that one must expect a certain degree of variation in network latency in the test system. This would have been a problem if the deadlines in the wind farm SCADA system has been in the millisecond range, instead of in the seconds range, and might be a factor in explaining the large deviations in the tests. But as this variation does not seem to have a negative influence on the end-to-end based prototype it is not likely that these network fluctuations can be attributed as the cause.

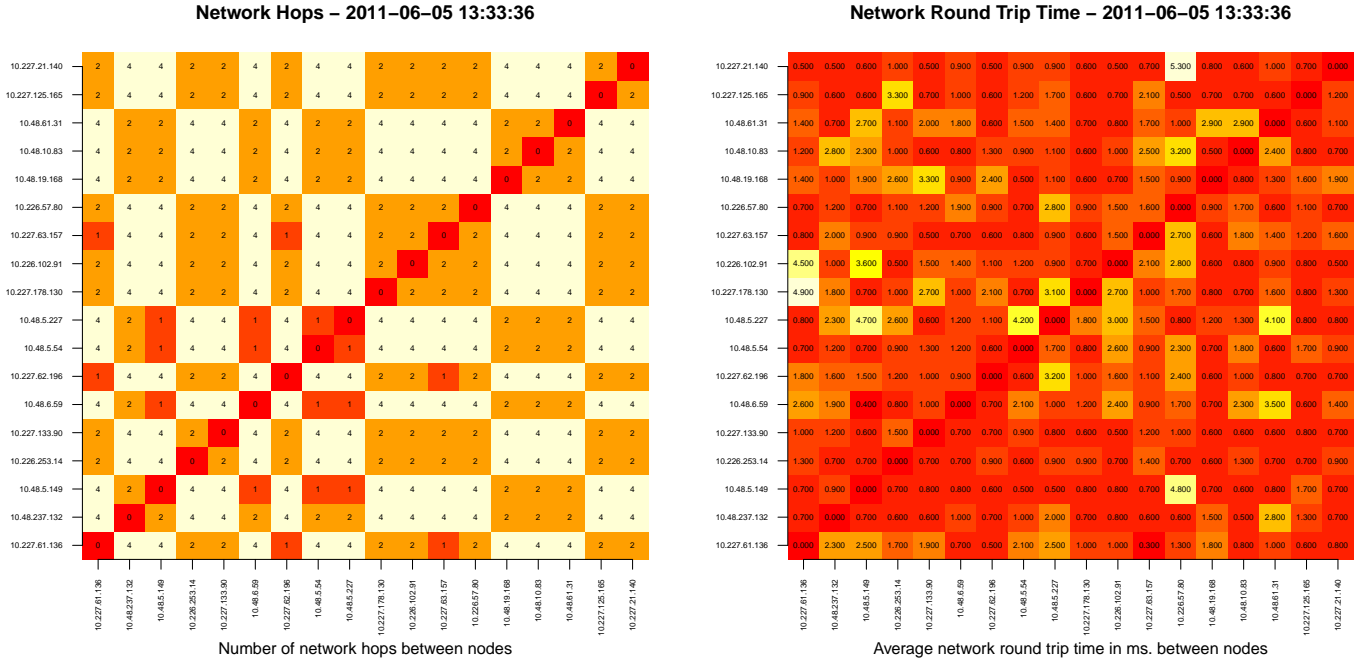
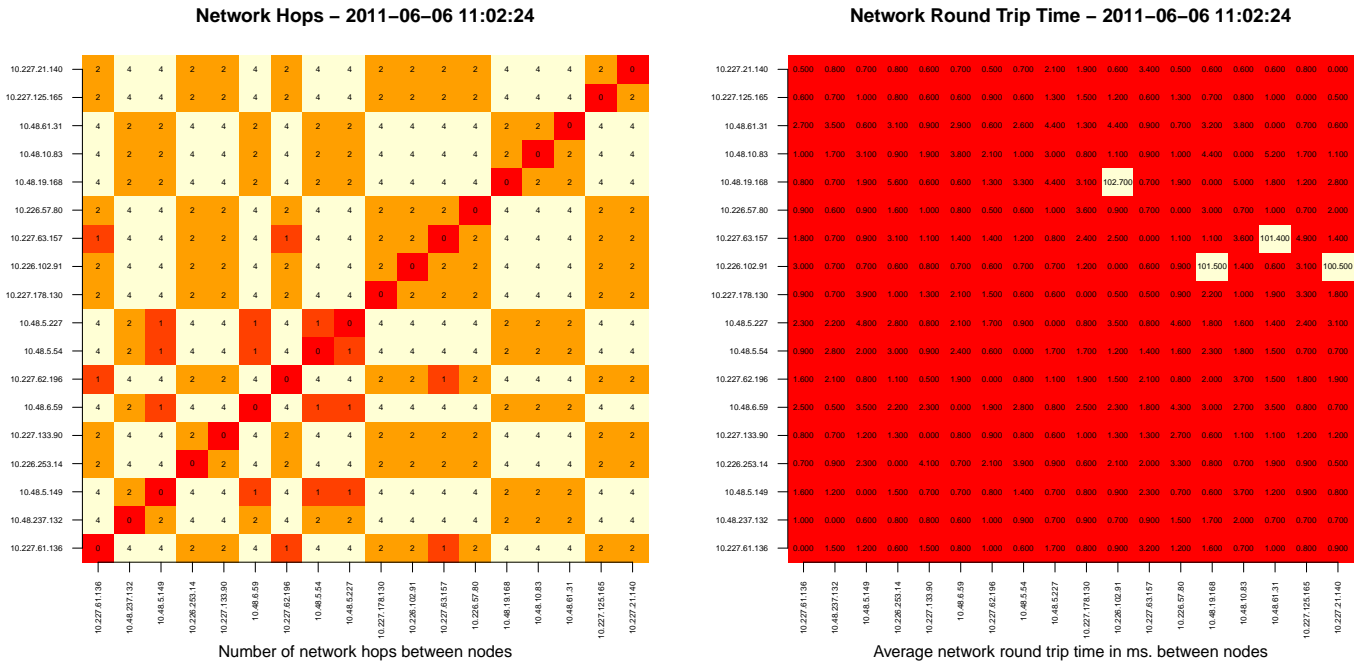


Figure 3: Network measurements *before* the test runs.



### 3 Measurements for the 100 Prototype Test Runs

The following 100 images each depict the measured timings during *one* prototype test run. It is seen that some of the test runs, such as test number 1, nicely fulfills the QAS deadlines. Then there are tests like test number 2 where there is a single failover situation where the 5 seconds deadline (QAS1) is slightly exceeded.

But then there are also some of the test runs where the prototype almost breaks down, like test number 4, 5 and 7 where no data seems to reach the clients in the first couple of periods. The most extreme of these failures are test number 12, 25, 26, 87, 88 and 91 where the data age reaches almost 110 seconds before falling back to 1 second.

The actual data for one of the problematic periods in test number 12 are depicted below in table 1 for one of the clients. From the min column, showing the minimum age of the 100 sensors that the client subscribes to, it is clear that some sensor values are steadily coming through to the client. Based on the mean column values of 25 seconds and the max values of 100 seconds it can be estimated that if the age for the 100 sensor values falls in just two groups, one group with ages around 1 second, and one group with values around 100 seconds, approximately 25 of the sensor values are not getting updated. This could be interpreted as data from 2 or 3 turbine nodes are not getting through to the client.

```
"min";"mean";"max"
...;.....;.....
544;24173;100017
544;24472;101017
544;24771;102017
544;25070;103017
547;25372;104020
543;25666;105016
548;25970;106021
543;26264;107016
551;26571;108024
543;26862;109016
526;12018;110021
522;733;977
525;736;981
```

Table 1: Data from one of the clients from test run number 12  
(source archive file: test-runs/hazelcast-timings/  
20110605\_160332\_timings/46.137.54.3.csv).

But this analysis of what is actually going on is only a hypothesis, that will need to be verified by collecting more fine grained data from the nodes. As the problem only seems to be present at the beginning of the test runs, it could to be attributed to some of the turbine nodes not joining the Hazelcast cluster correctly. A hypothesis that would also need further debugging to verify.

So summarizing there are at least two problems with the prototype:

1. Occasionally small exceedings of the 5 second deadline. This problem

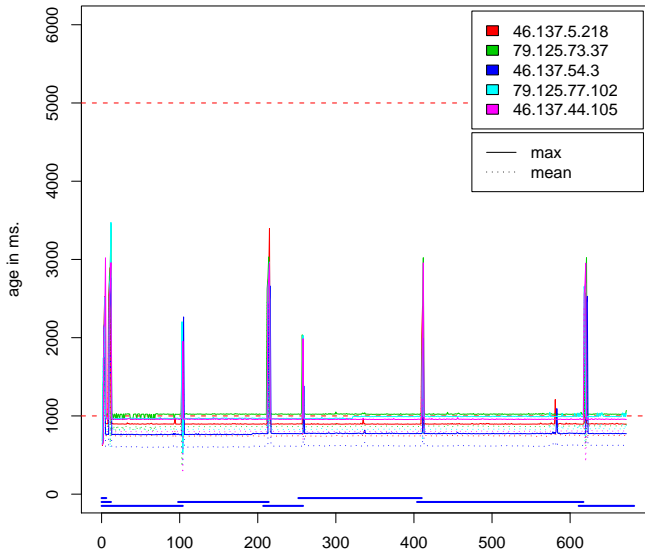
might be hard to solve, as it is not expected to be a direct bug in the source code, but more a performance problem with the Hazelcast multicast algorithm.

2. Problems at startup for some tests, leading to significantly exceeding the 5 second deadline. This problem is probably attributable to a bug in the source code. With some effort it is assumed that it would be possible to find and fix this.

### 3.1 Crash After Test Run 17

During the test run the management console running on the local workstation crashed after test run number 17. Therefore the test was restarted at that point for the remaining 83 tests. When the test runs was later analysed it was noticed that this restart poses a statistical problem as the seed for the restarted sequence was also reset, wherefore test number 1 and test number 18 is actually a test of the same crash sequence (and so on up until test number 17 and 34). Seen from a straight statistical view the Hazelcast test should be rerun to be completely comparable to the other test run. But from a pragmatic view it is not assumed that this would lead to any change in the conclusion about the lacking performance of the Hazelcast prototype.

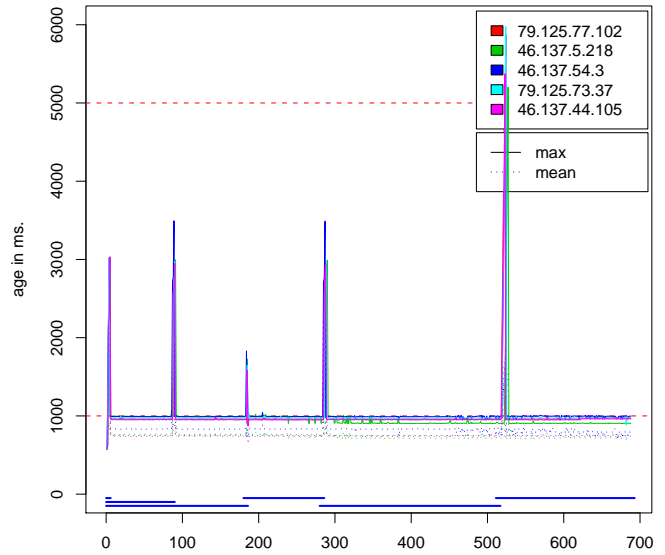
Client Data Age – 2011-06-05 13:46:01



Age in ms. of data on one client – sampling every second

(Test run 1)

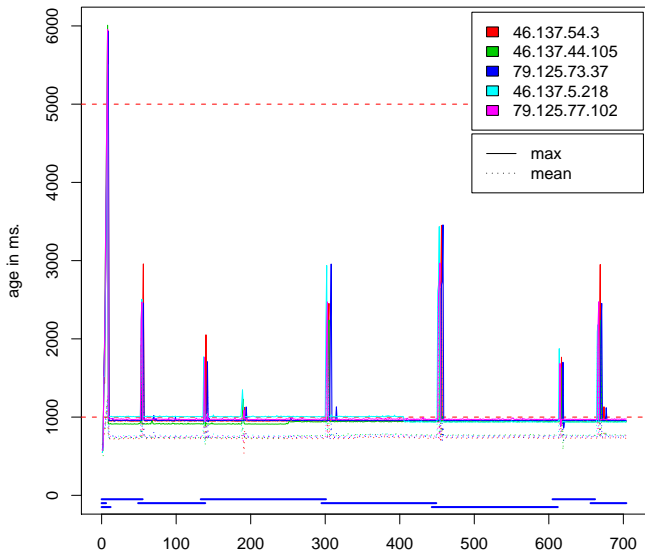
Client Data Age – 2011-06-05 13:58:19



Age in ms. of data on one client – sampling every second

(Test run 2)

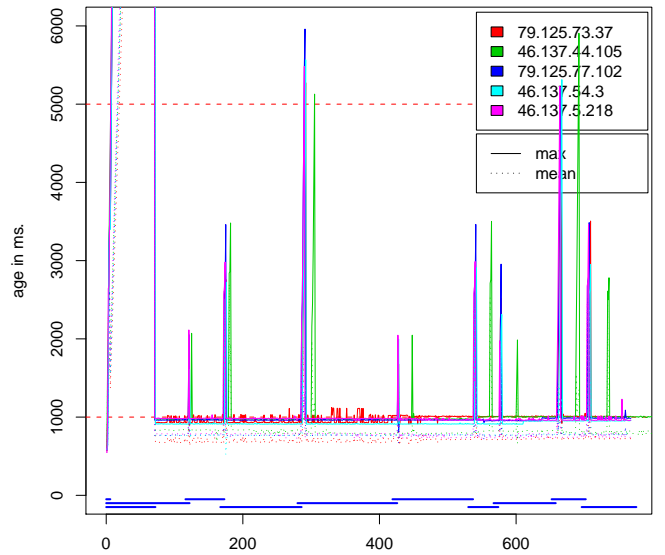
Client Data Age – 2011-06-05 14:10:50



Age in ms. of data on one client – sampling every second

(Test run 3)

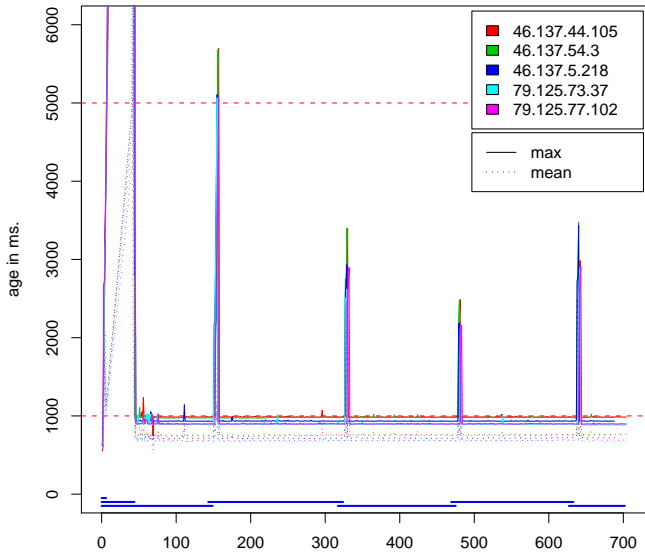
Client Data Age – 2011-06-05 14:24:35



Age in ms. of data on one client – sampling every second

(Test run 4)

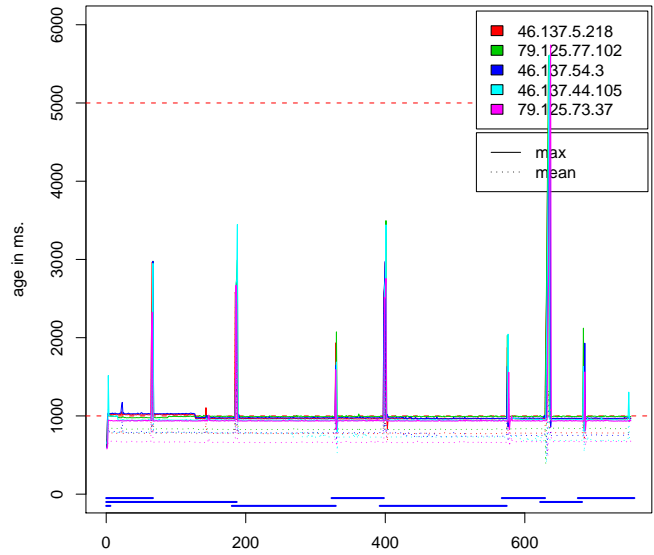
Client Data Age – 2011-06-05 14:37:03



Age in ms. of data on one client – sampling every second

(Test run 5)

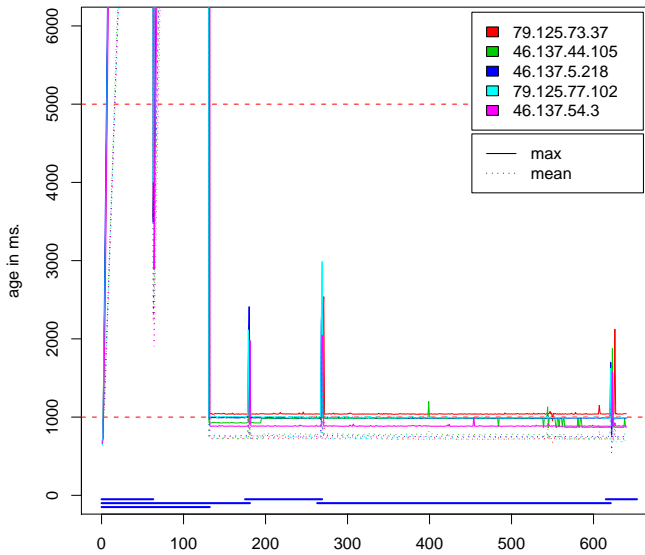
Client Data Age – 2011-06-05 14:50:25



Age in ms. of data on one client – sampling every second

(Test run 6)

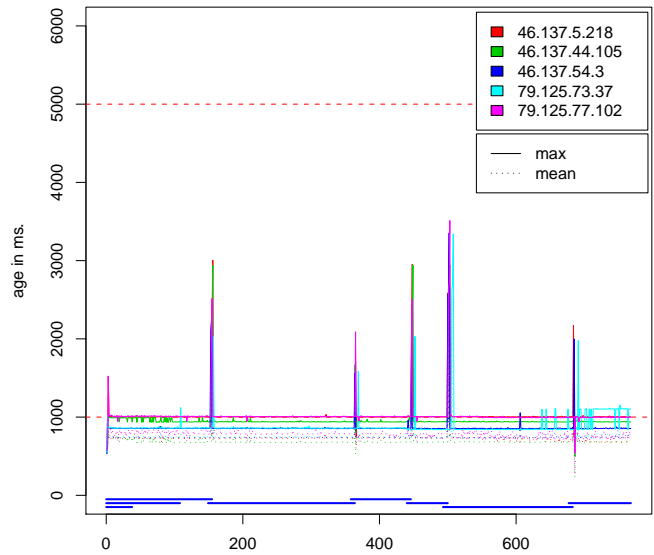
Client Data Age – 2011-06-05 15:02:06



Age in ms. of data on one client – sampling every second

(Test run 7)

Client Data Age – 2011-06-05 15:15:41

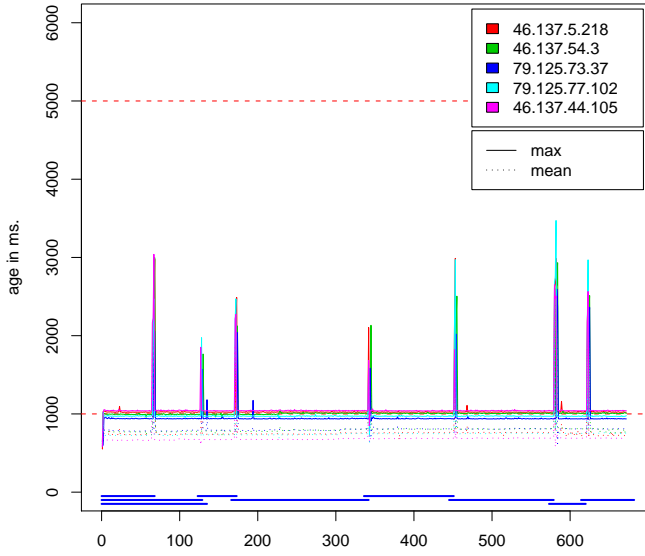


Age in ms. of data on one client – sampling every second

(Test run 8)



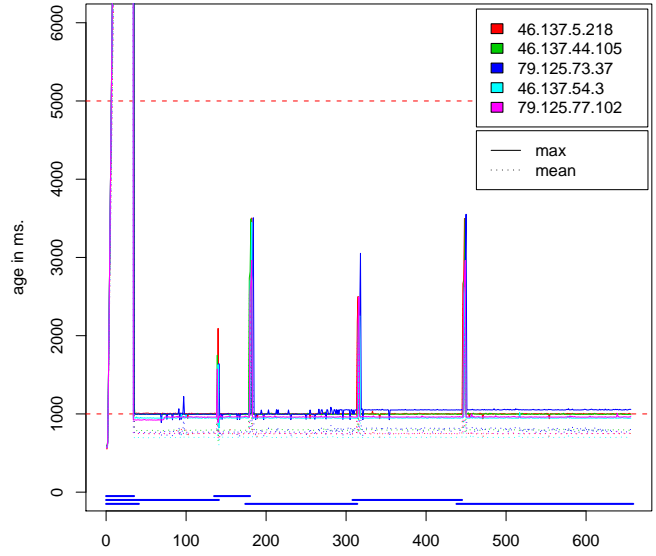
Client Data Age – 2011-06-05 15:27:53



Age in ms. of data on one client – sampling every second

(Test run 9)

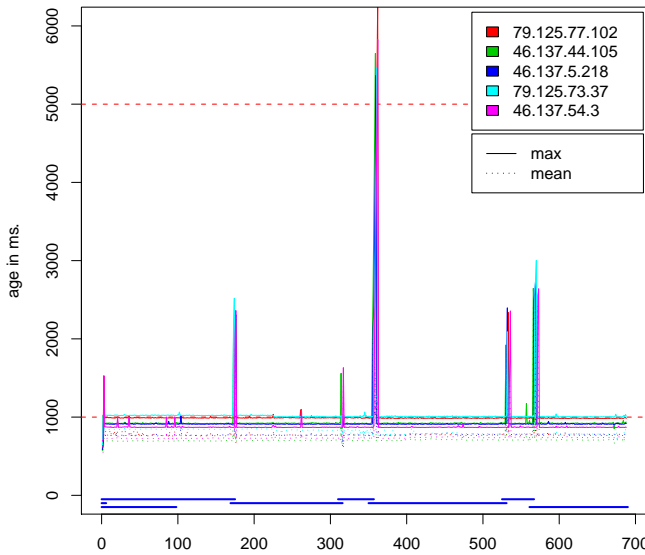
Client Data Age – 2011-06-05 15:39:40



Age in ms. of data on one client – sampling every second

(Test run 10)

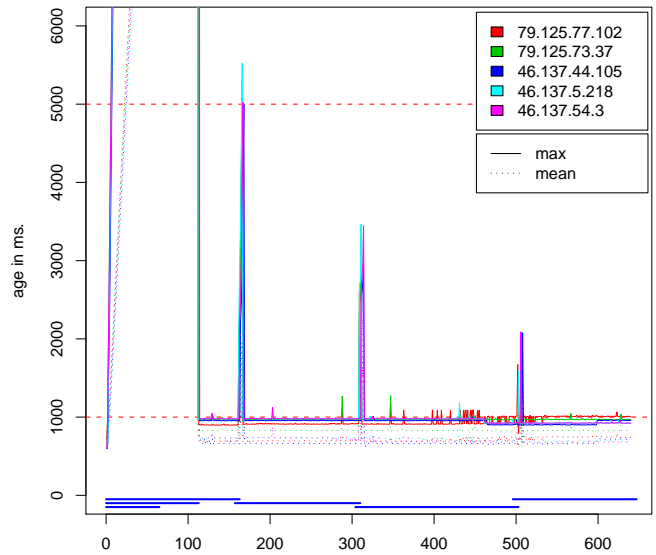
Client Data Age – 2011-06-05 15:52:02



Age in ms. of data on one client – sampling every second

(Test run 11)

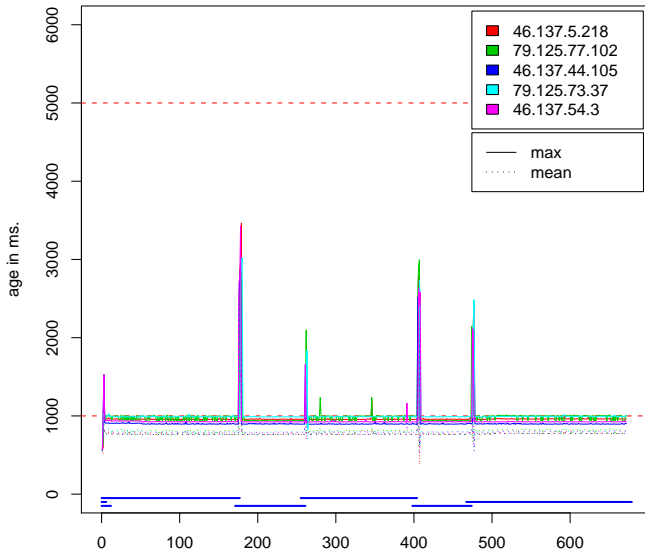
Client Data Age – 2011-06-05 16:03:39



Age in ms. of data on one client – sampling every second

(Test run 12)

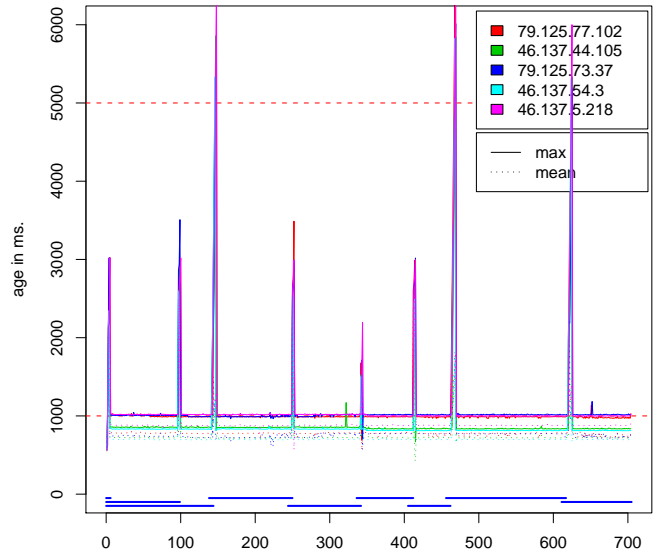
Client Data Age – 2011-06-05 16:15:48



Age in ms. of data on one client – sampling every second

(Test run 13)

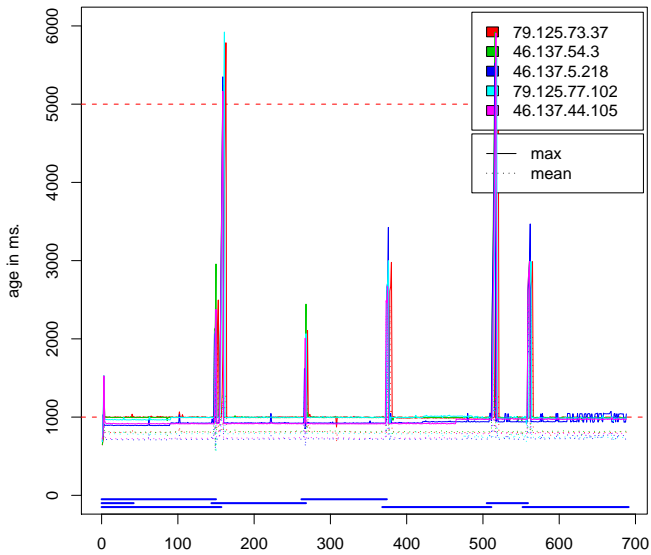
Client Data Age – 2011-06-05 16:28:23



Age in ms. of data on one client – sampling every second

(Test run 14)

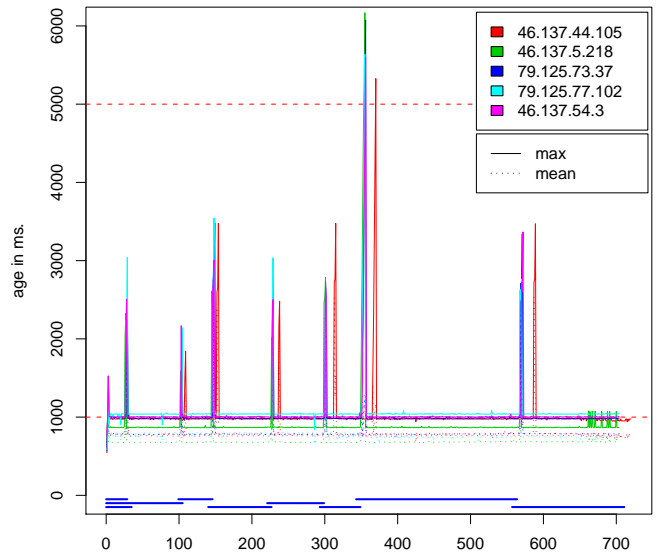
Client Data Age – 2011-06-05 16:40:43



Age in ms. of data on one client – sampling every second

(Test run 15)

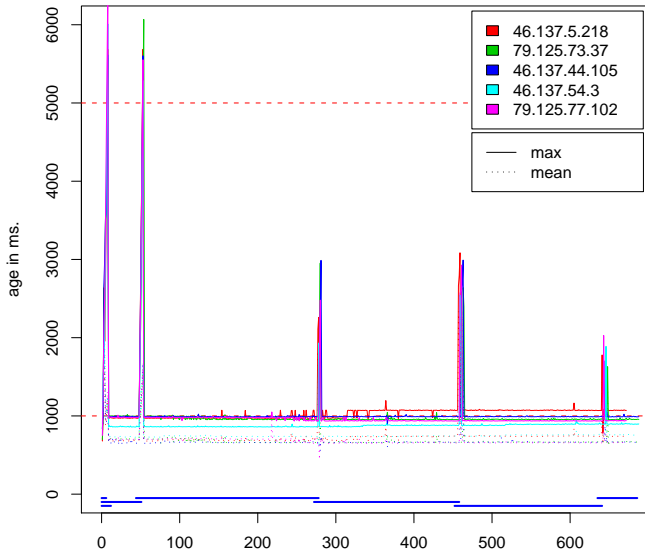
Client Data Age – 2011-06-05 16:53:19



Age in ms. of data on one client – sampling every second

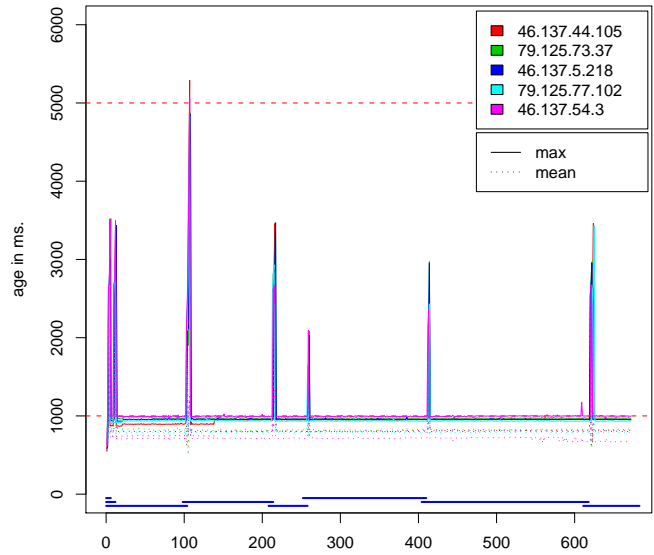
(Test run 16)

Client Data Age – 2011-06-05 17:05:37



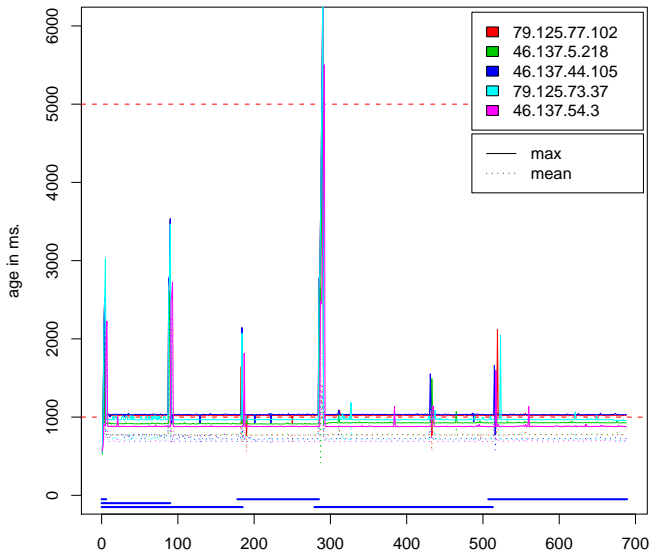
Age in ms. of data on one client – sampling every second  
(Test run 17) Last test before crash.

Client Data Age – 2011-06-05 17:54:14



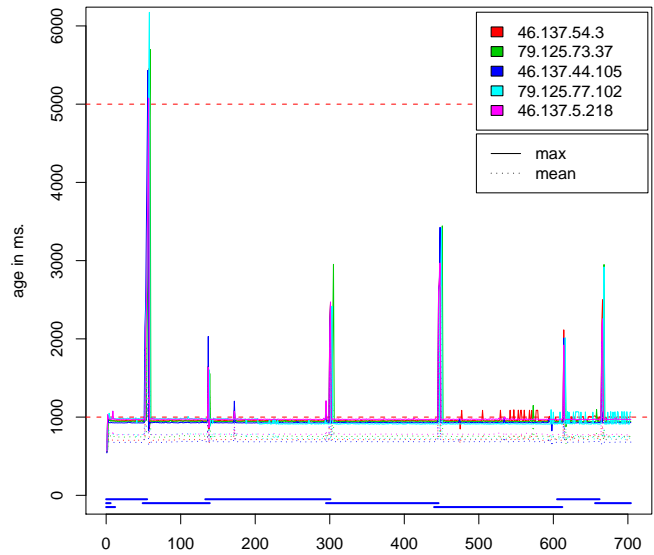
Age in ms. of data on one client – sampling every second  
(Test run 18) First test after crash.

Client Data Age – 2011-06-05 18:06:34



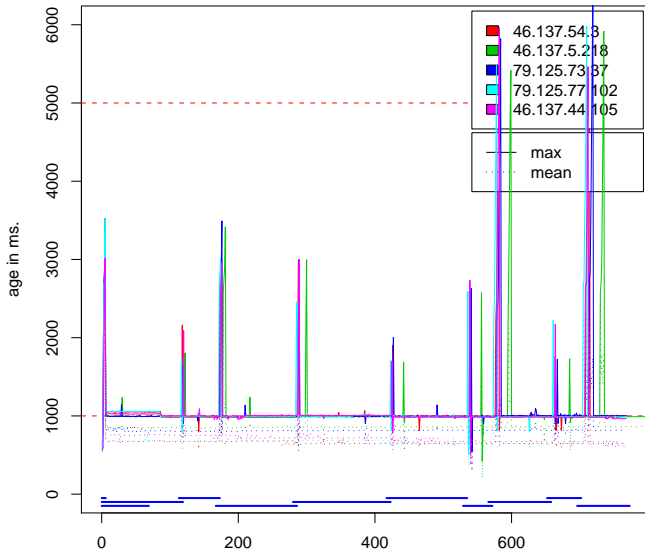
Age in ms. of data on one client – sampling every second  
(Test run 19)

Client Data Age – 2011-06-05 18:19:06



Age in ms. of data on one client – sampling every second  
(Test run 20)

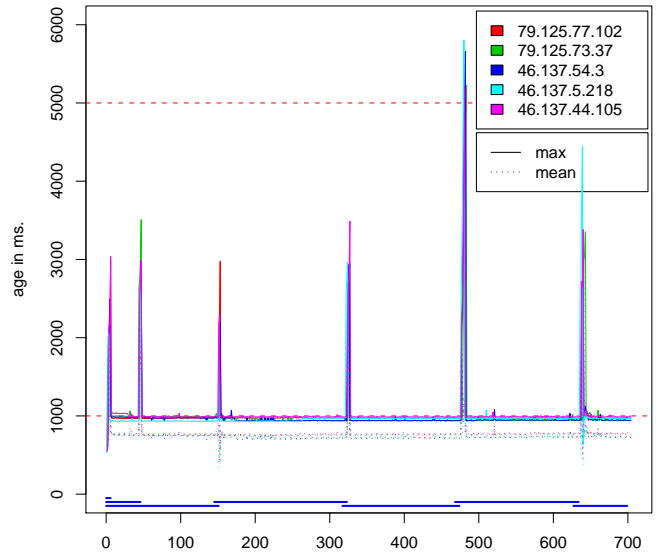
Client Data Age – 2011-06-05 18:32:49



Age in ms. of data on one client – sampling every second

(Test run 21)

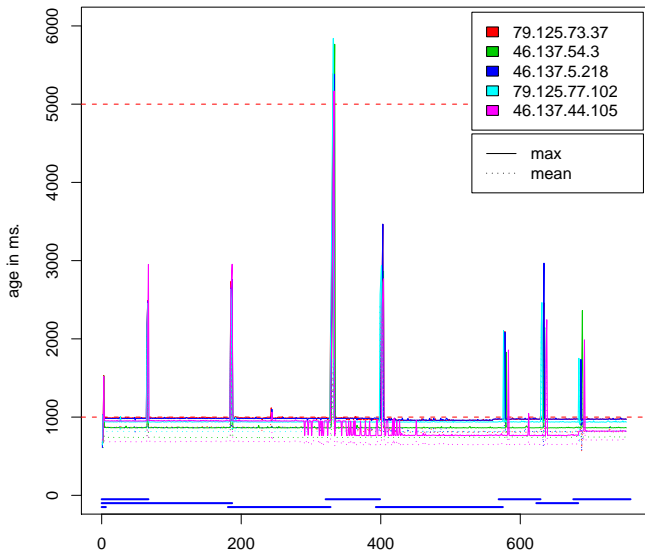
Client Data Age – 2011-06-05 18:45:24



Age in ms. of data on one client – sampling every second

(Test run 22)

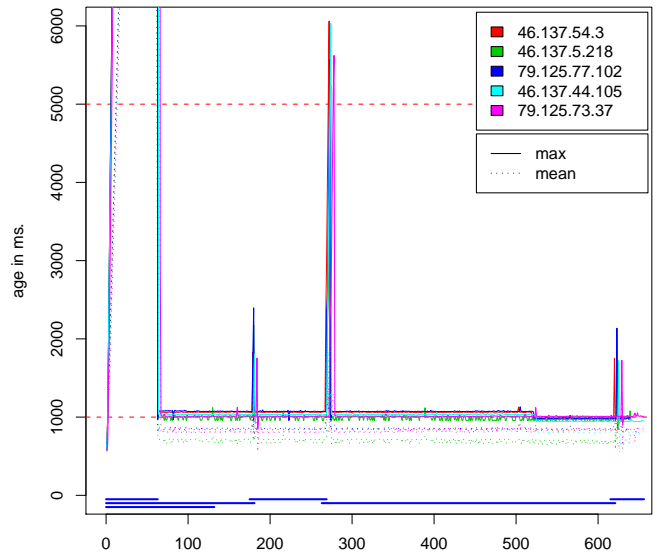
Client Data Age – 2011-06-05 18:58:49



Age in ms. of data on one client – sampling every second

(Test run 23)

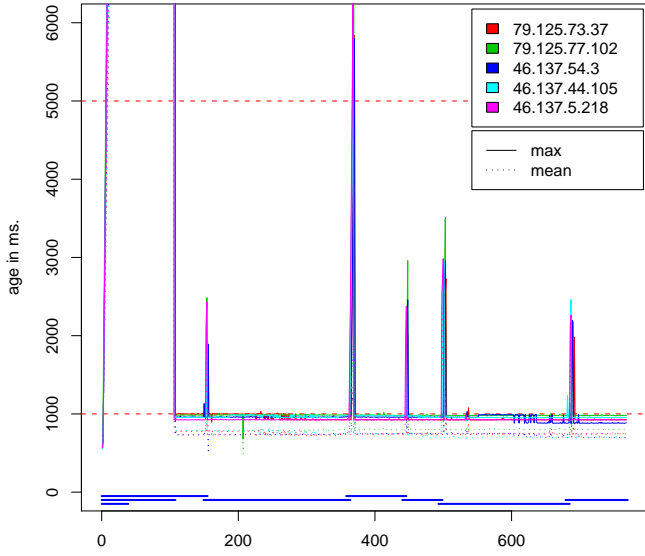
Client Data Age – 2011-06-05 19:10:33



Age in ms. of data on one client – sampling every second

(Test run 24)

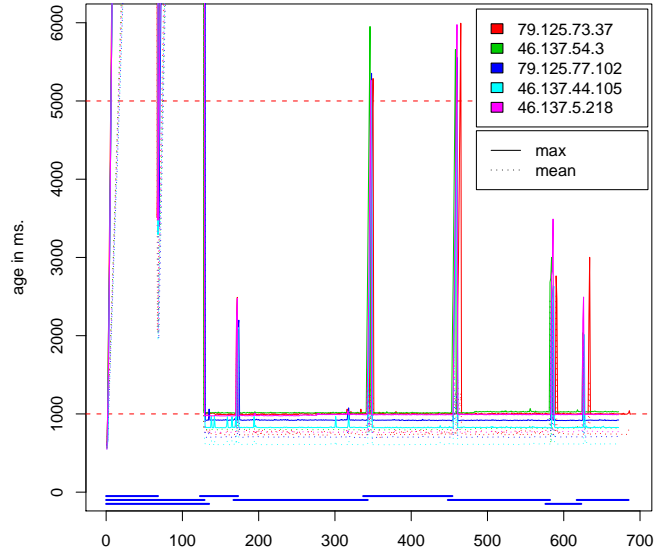
Client Data Age – 2011-06-05 19:24:11



Age in ms. of data on one client – sampling every second

(Test run 25)

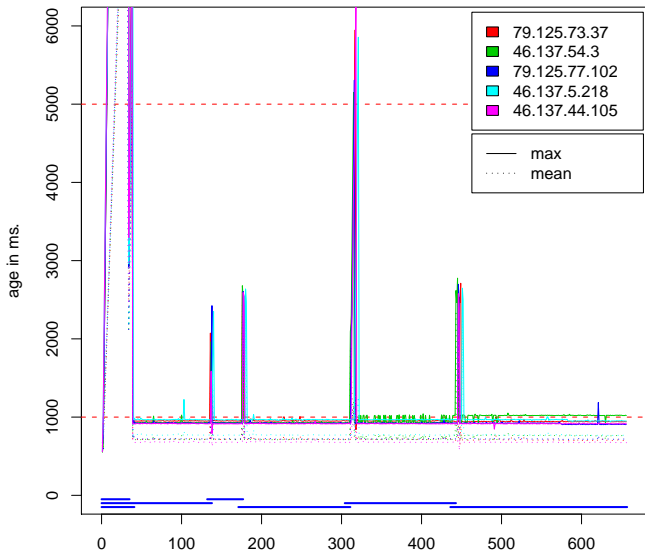
Client Data Age – 2011-06-05 19:36:22



Age in ms. of data on one client – sampling every second

(Test run 26)

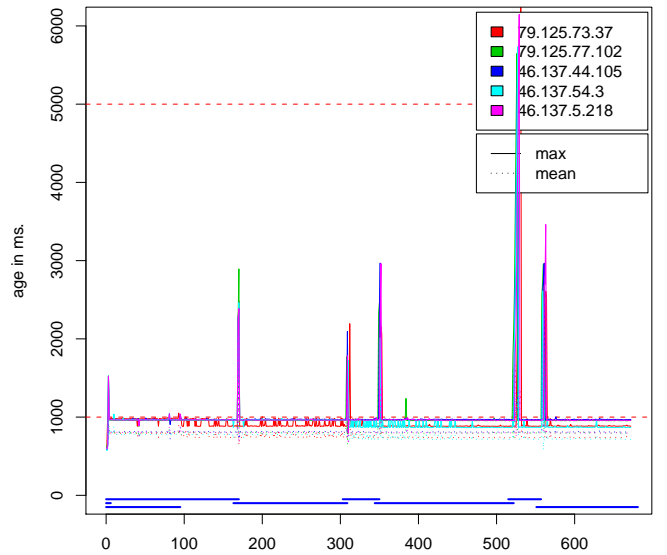
Client Data Age – 2011-06-05 19:48:05



Age in ms. of data on one client – sampling every second

(Test run 27)

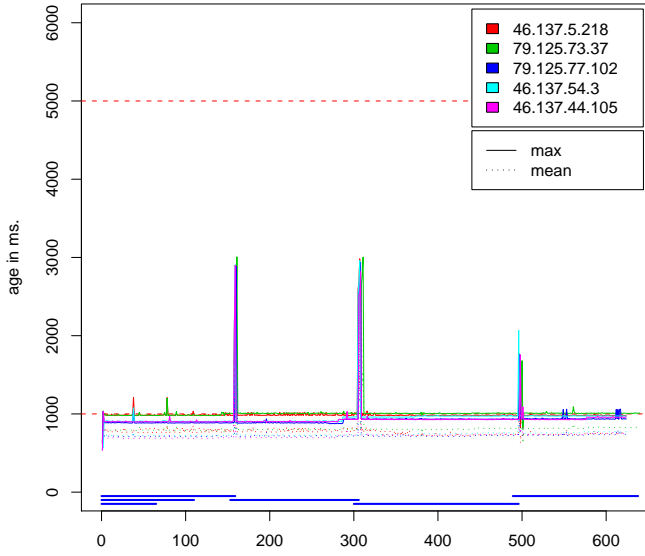
Client Data Age – 2011-06-05 20:00:12



Age in ms. of data on one client – sampling every second

(Test run 28)

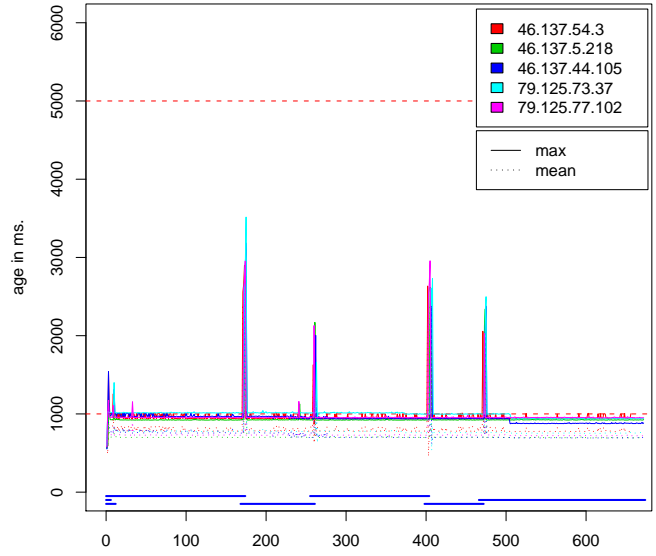
Client Data Age – 2011-06-05 20:11:36



Age in ms. of data on one client – sampling every second

(Test run 29)

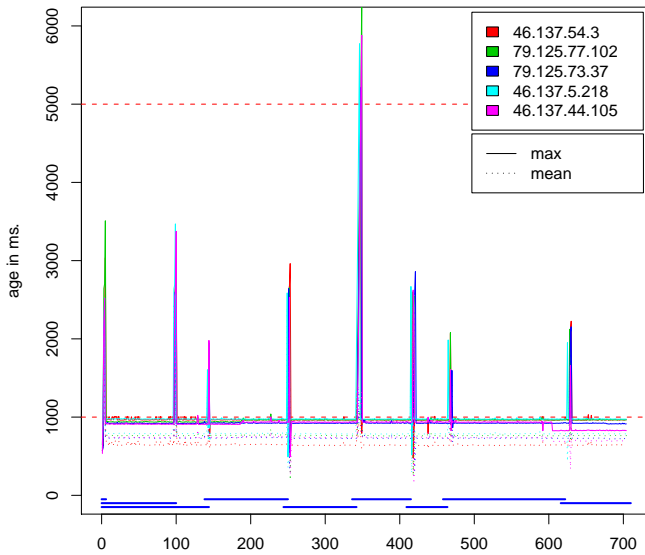
Client Data Age – 2011-06-05 20:23:42



Age in ms. of data on one client – sampling every second

(Test run 30)

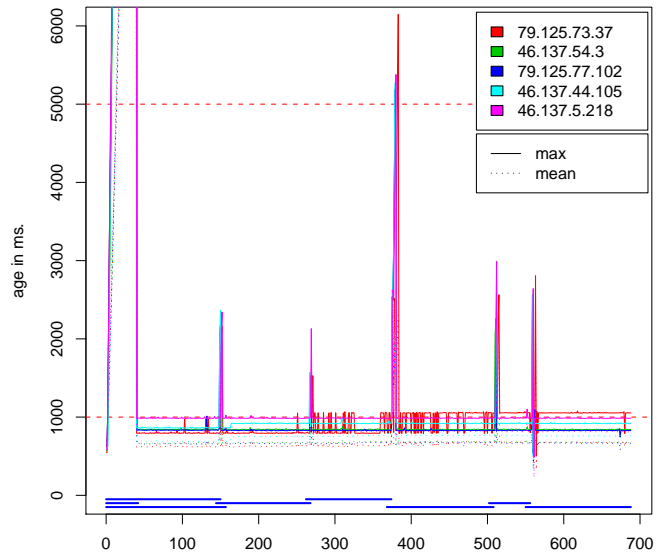
Client Data Age – 2011-06-05 20:36:18



Age in ms. of data on one client – sampling every second

(Test run 31)

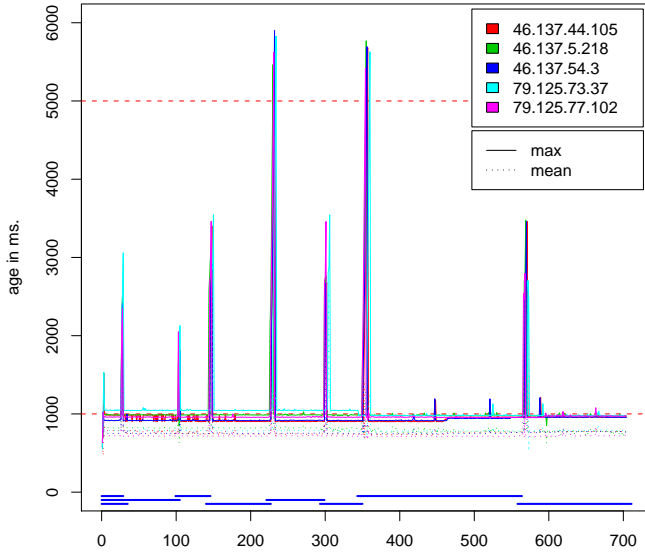
Client Data Age – 2011-06-05 20:48:32



Age in ms. of data on one client – sampling every second

(Test run 32)

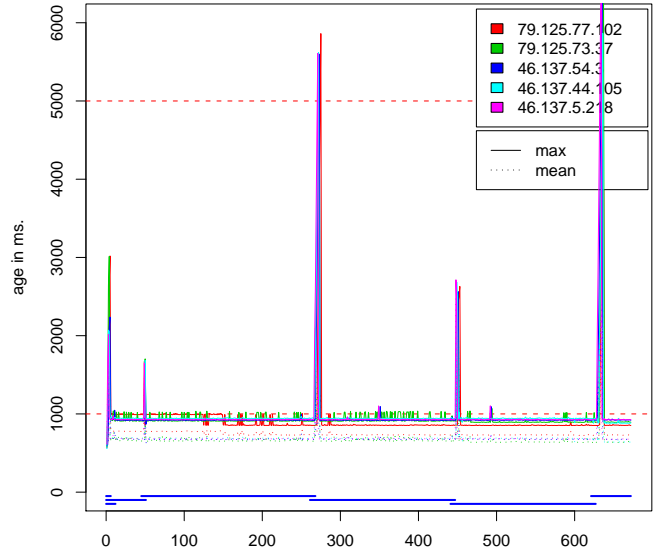
Client Data Age – 2011-06-05 21:01:19



Age in ms. of data on one client – sampling every second

(Test run 33)

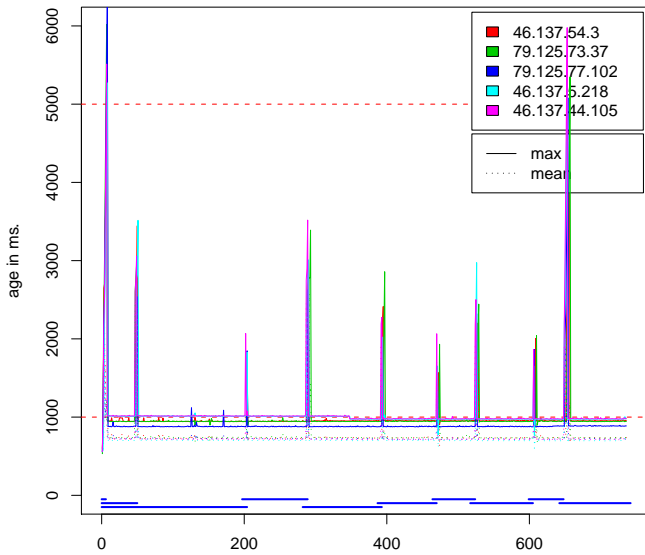
Client Data Age – 2011-06-05 21:13:17



Age in ms. of data on one client – sampling every second

(Test run 34)

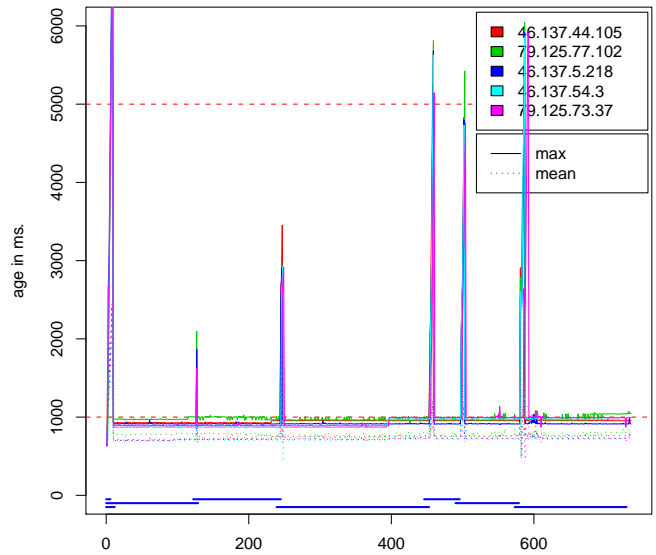
Client Data Age – 2011-06-05 21:26:24



Age in ms. of data on one client – sampling every second

(Test run 35)

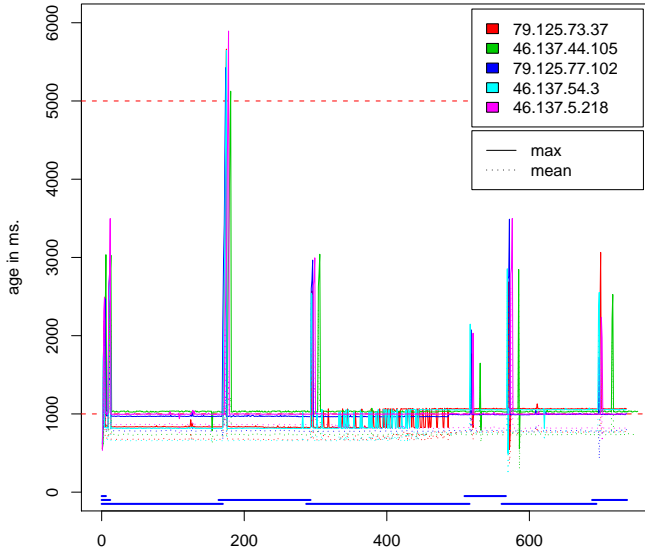
Client Data Age – 2011-06-05 21:39:22



Age in ms. of data on one client – sampling every second

(Test run 36)

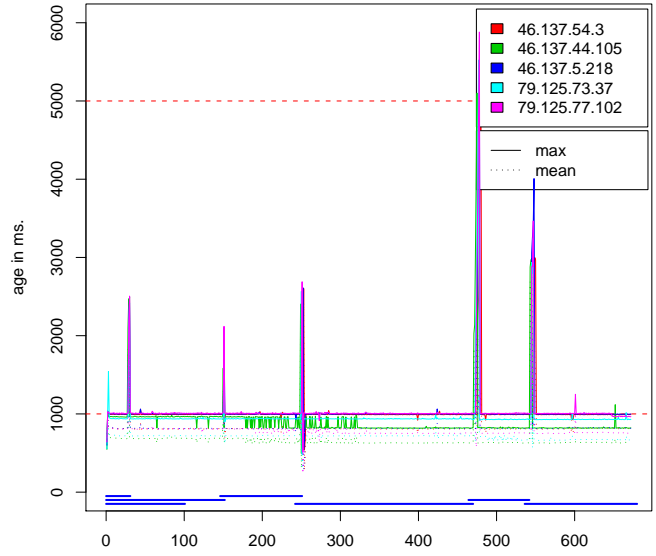
Client Data Age – 2011-06-05 21:52:28



Age in ms. of data on one client – sampling every second

(Test run 37)

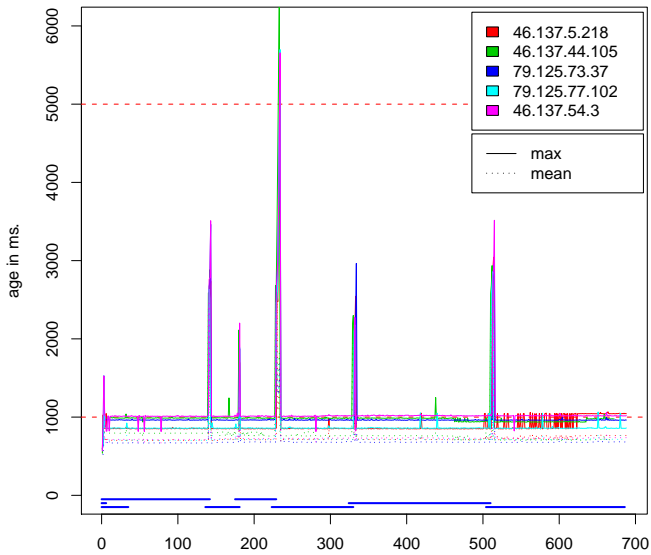
Client Data Age – 2011-06-05 22:04:32



Age in ms. of data on one client – sampling every second

(Test run 38)

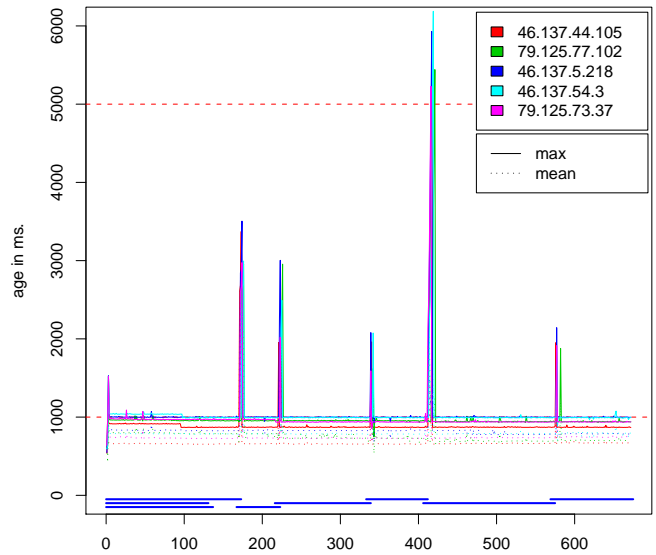
Client Data Age – 2011-06-05 22:16:40



Age in ms. of data on one client – sampling every second

(Test run 39)

Client Data Age – 2011-06-05 22:28:42

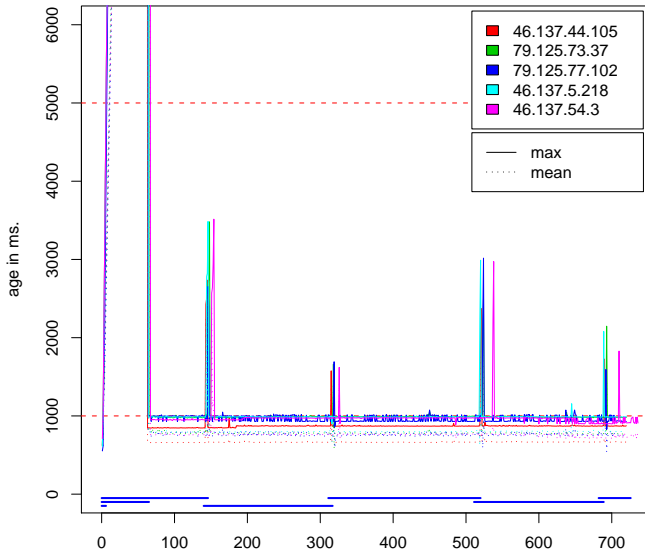


Age in ms. of data on one client – sampling every second

(Test run 40)



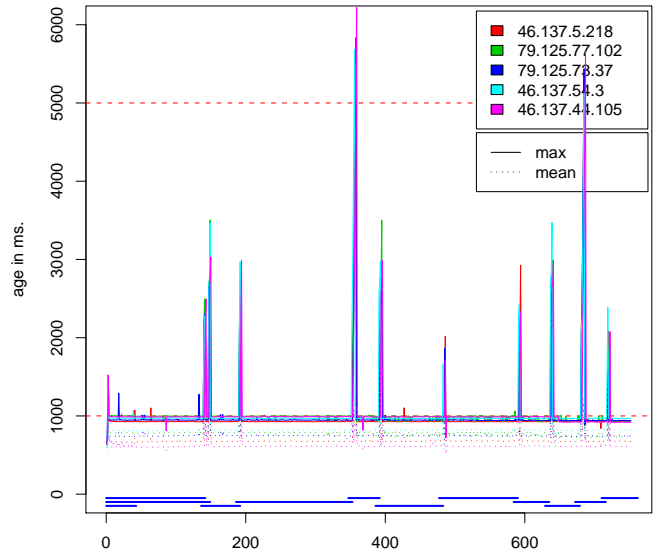
Client Data Age – 2011-06-05 22:41:37



Age in ms. of data on one client – sampling every second

(Test run 41)

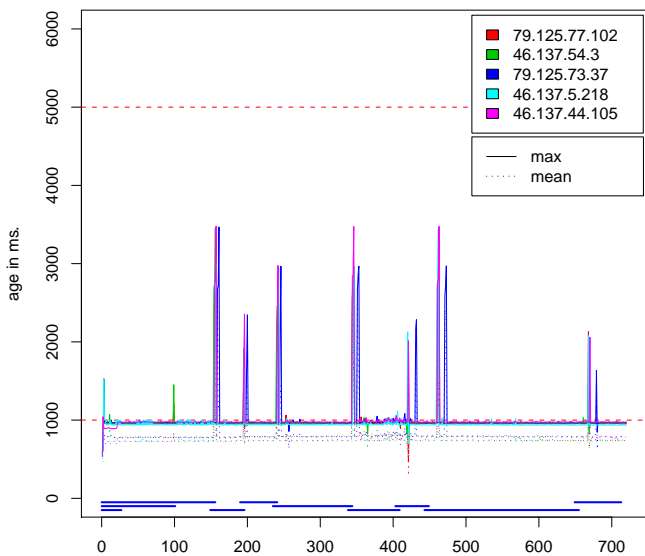
Client Data Age – 2011-06-05 22:55:15



Age in ms. of data on one client – sampling every second

(Test run 42)

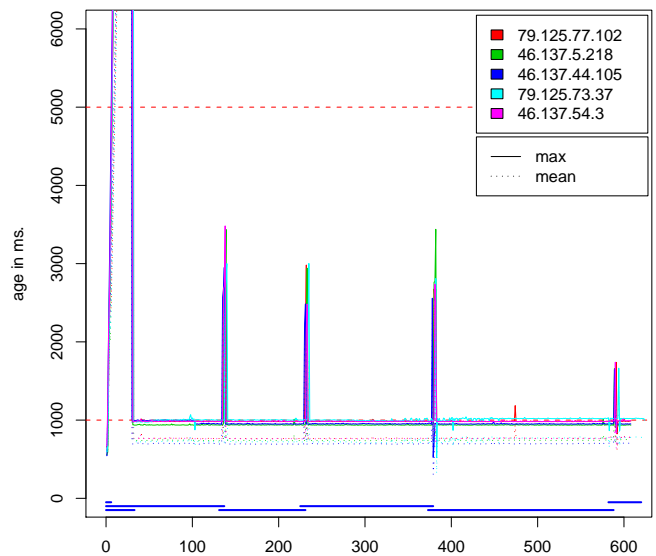
Client Data Age – 2011-06-05 23:07:54



Age in ms. of data on one client – sampling every second

(Test run 43)

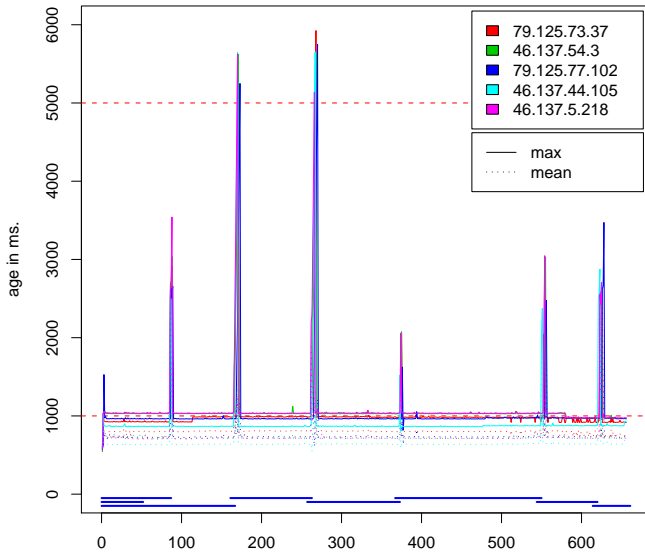
Client Data Age – 2011-06-05 23:18:57



Age in ms. of data on one client – sampling every second

(Test run 44)

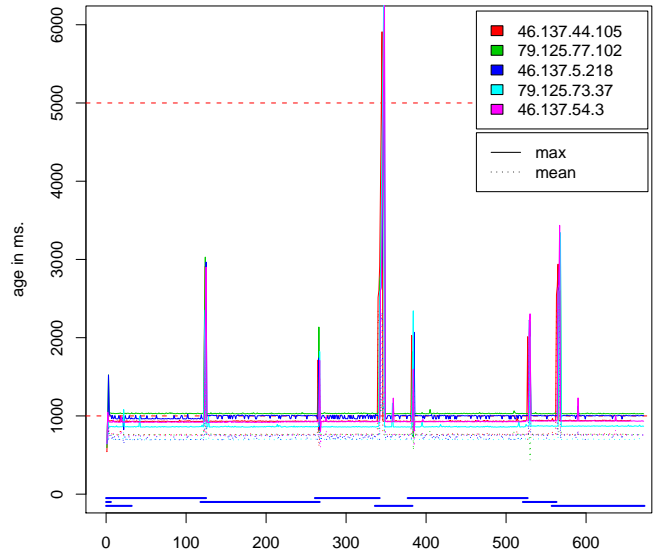
Client Data Age – 2011-06-05 23:30:47



Age in ms. of data on one client – sampling every second

(Test run 45)

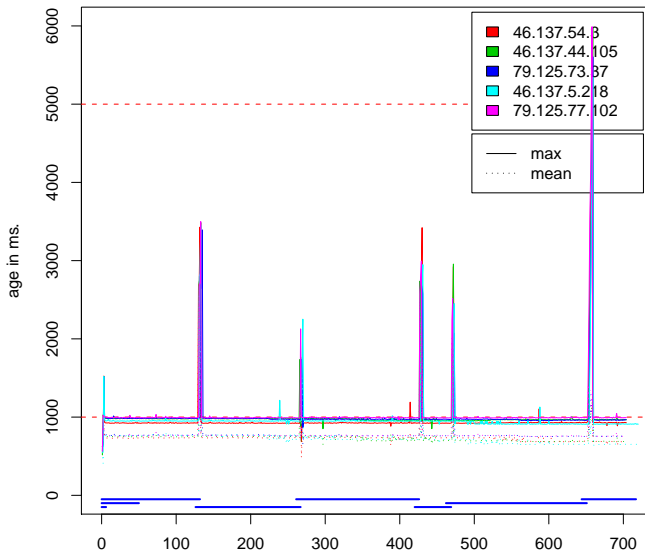
Client Data Age – 2011-06-05 23:43:01



Age in ms. of data on one client – sampling every second

(Test run 46)

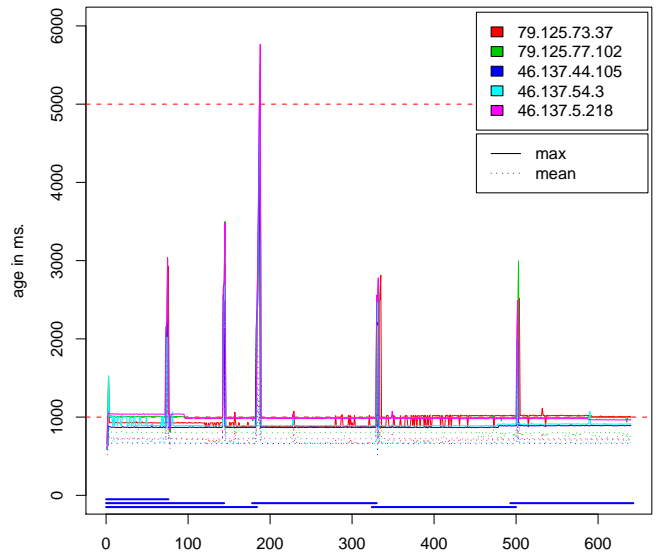
Client Data Age – 2011-06-05 23:55:40



Age in ms. of data on one client – sampling every second

(Test run 47)

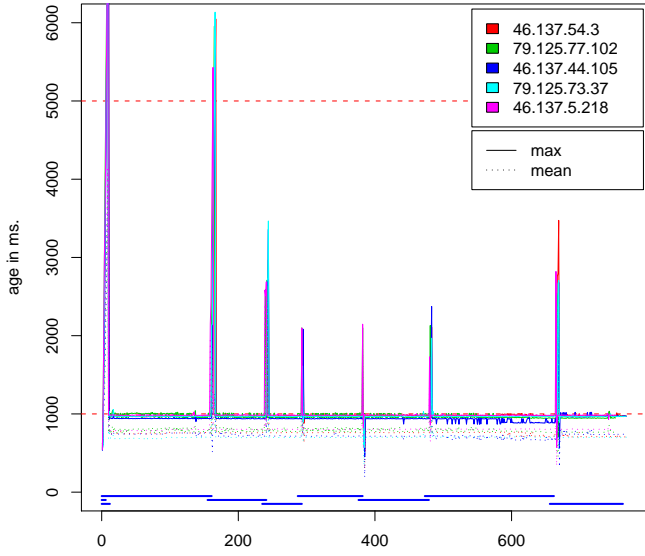
Client Data Age – 2011-06-06 00:07:10



Age in ms. of data on one client – sampling every second

(Test run 48)

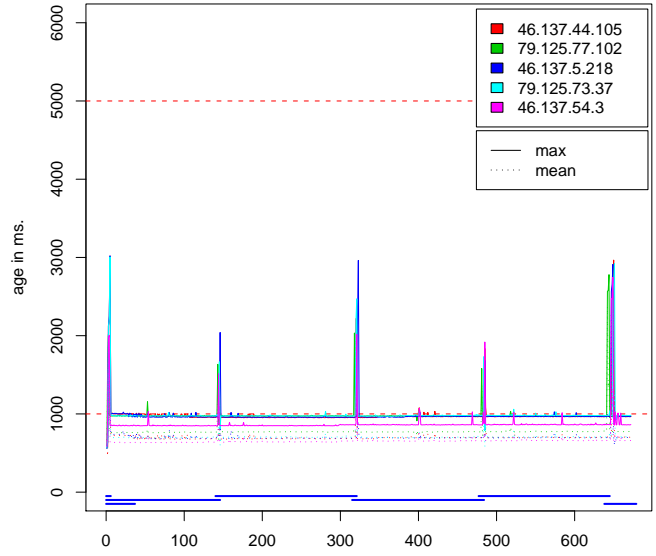
Client Data Age – 2011-06-06 00:20:48



Age in ms. of data on one client – sampling every second

(Test run 49)

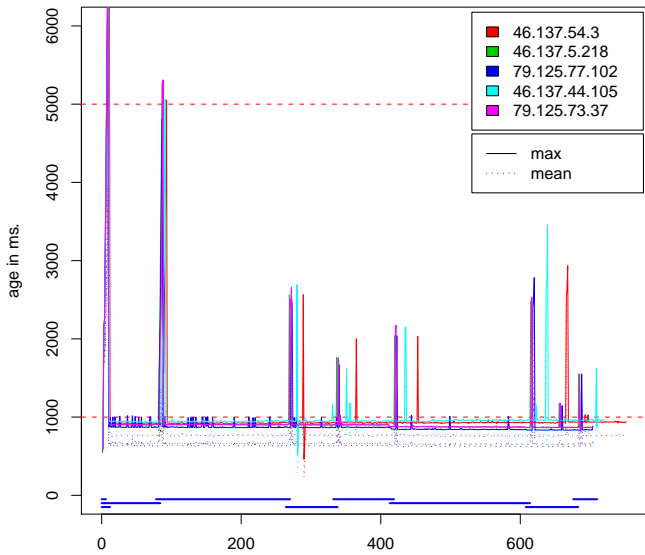
Client Data Age – 2011-06-06 00:32:54



Age in ms. of data on one client – sampling every second

(Test run 50)

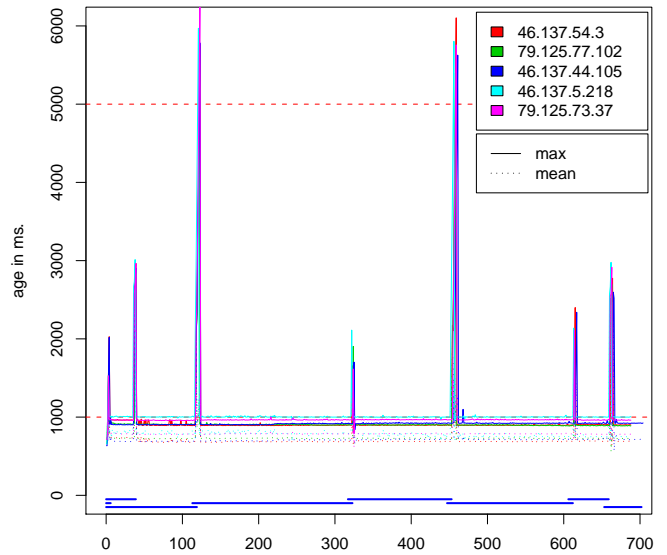
Client Data Age – 2011-06-06 00:45:25



Age in ms. of data on one client – sampling every second

(Test run 51)

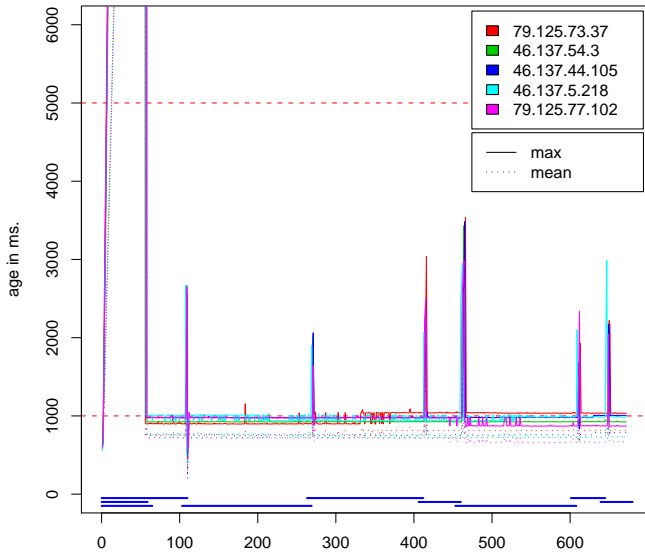
Client Data Age – 2011-06-06 00:57:55



Age in ms. of data on one client – sampling every second

(Test run 52)

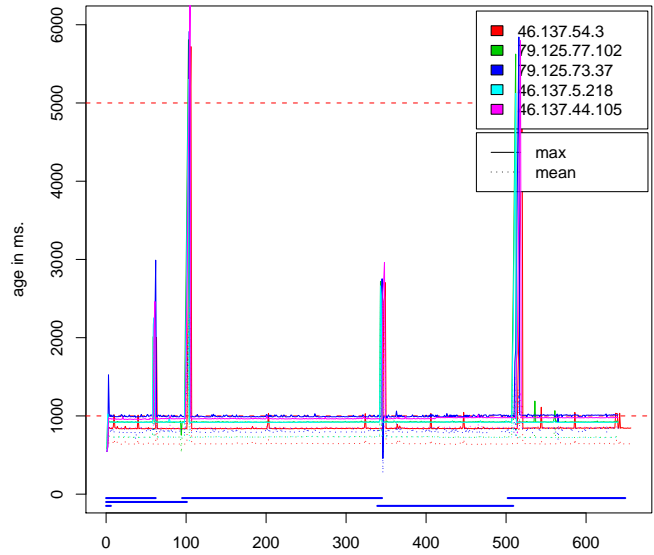
Client Data Age – 2011-06-06 01:10:03



Age in ms. of data on one client – sampling every second

(Test run 53)

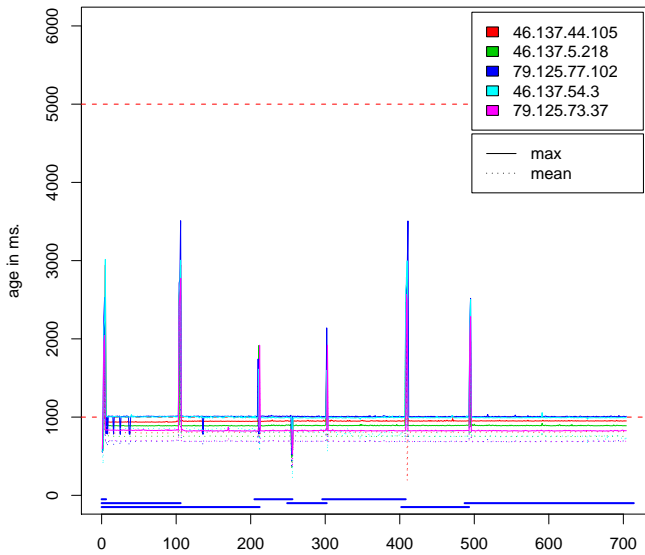
Client Data Age – 2011-06-06 01:21:39



Age in ms. of data on one client – sampling every second

(Test run 54)

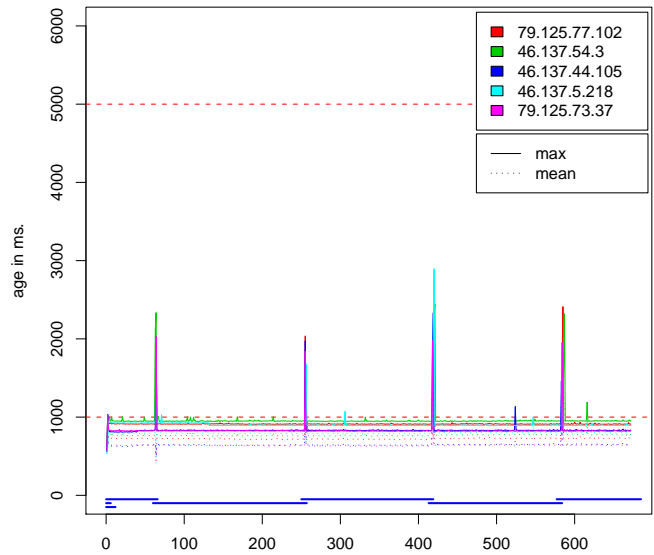
Client Data Age – 2011-06-06 01:34:21



Age in ms. of data on one client – sampling every second

(Test run 55)

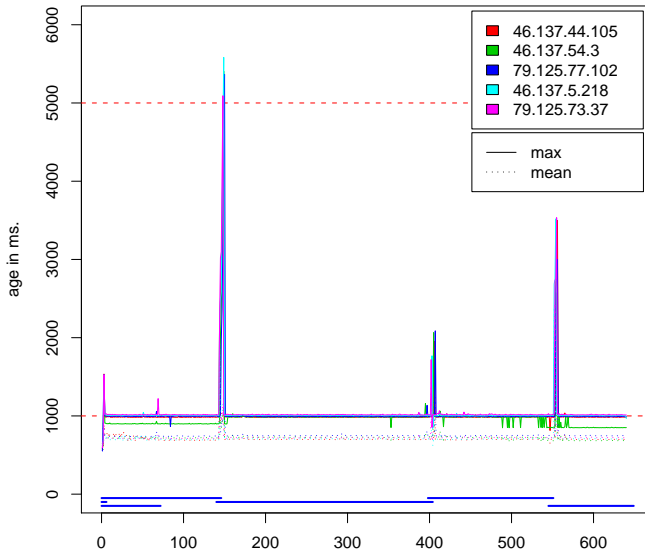
Client Data Age – 2011-06-06 01:46:34



Age in ms. of data on one client – sampling every second

(Test run 56)

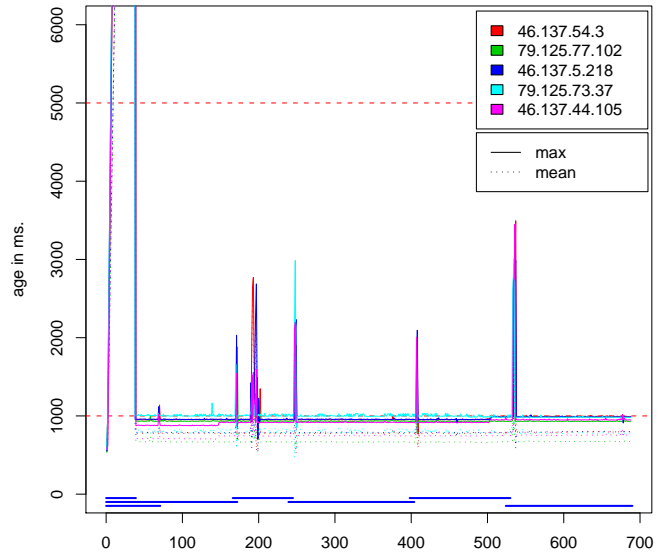
Client Data Age – 2011-06-06 01:58:15



Age in ms. of data on one client – sampling every second

(Test run 57)

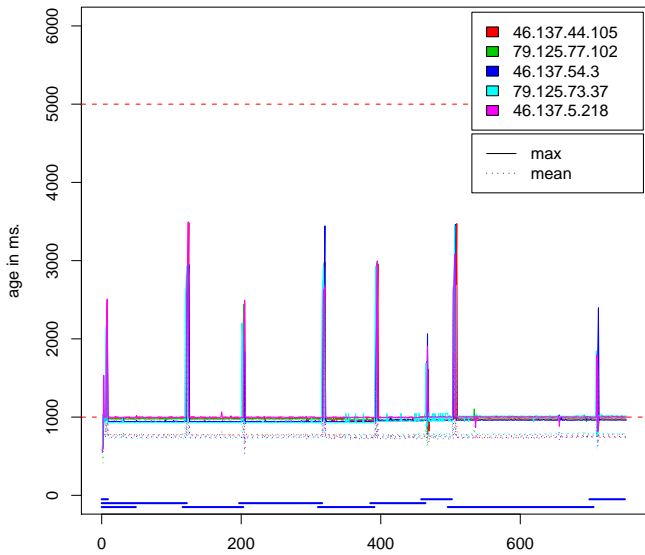
Client Data Age – 2011-06-06 02:10:33



Age in ms. of data on one client – sampling every second

(Test run 58)

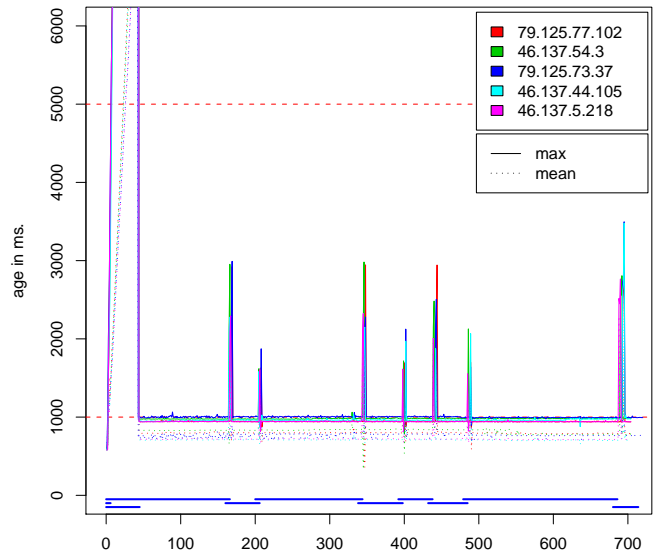
Client Data Age – 2011-06-06 02:23:55



Age in ms. of data on one client – sampling every second

(Test run 59)

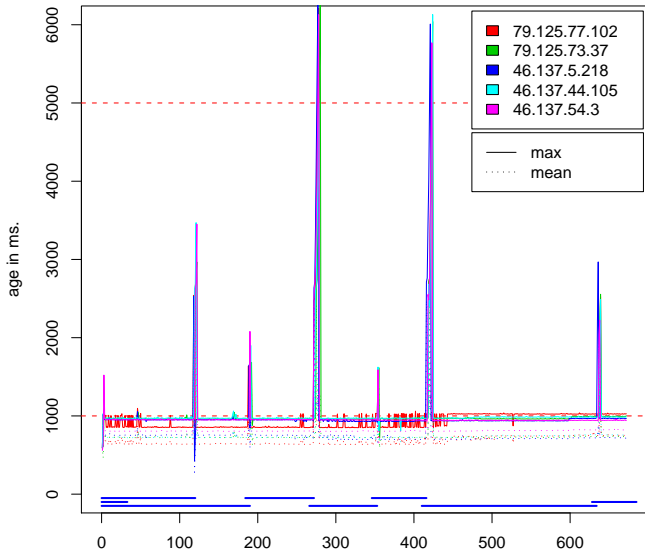
Client Data Age – 2011-06-06 02:36:39



Age in ms. of data on one client – sampling every second

(Test run 60)

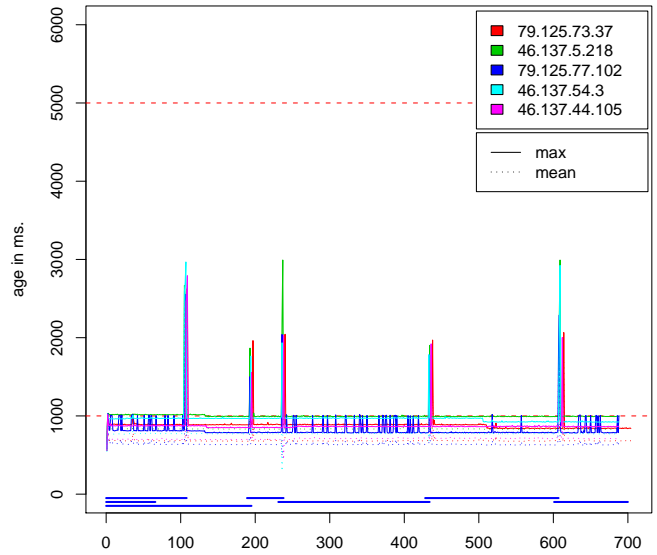
Client Data Age – 2011-06-06 02:48:50



Age in ms. of data on one client – sampling every second

(Test run 61)

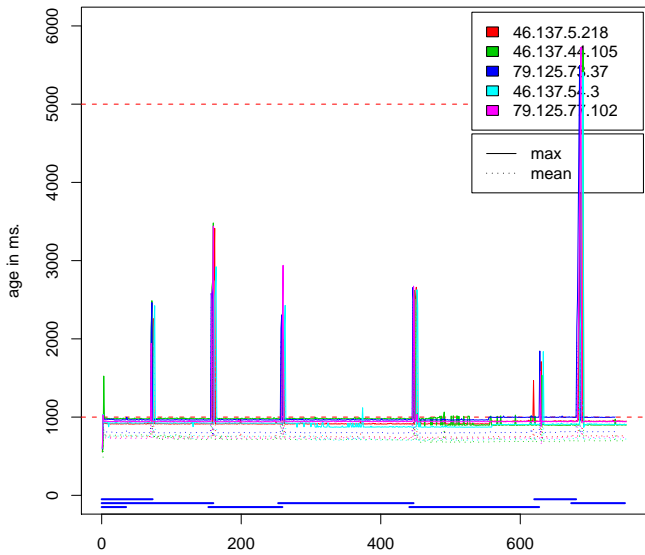
Client Data Age – 2011-06-06 03:01:17



Age in ms. of data on one client – sampling every second

(Test run 62)

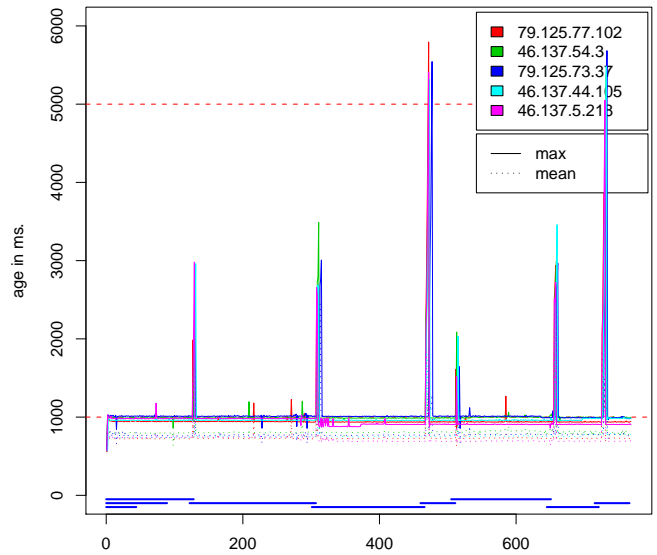
Client Data Age – 2011-06-06 03:14:40



Age in ms. of data on one client – sampling every second

(Test run 63)

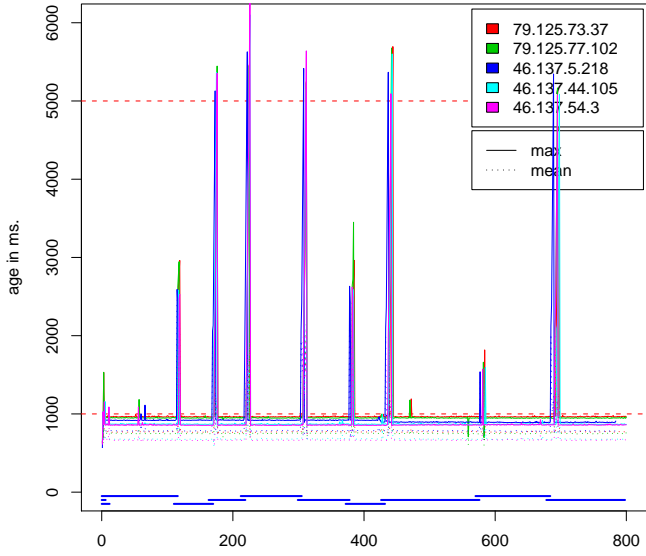
Client Data Age – 2011-06-06 03:28:20



Age in ms. of data on one client – sampling every second

(Test run 64)

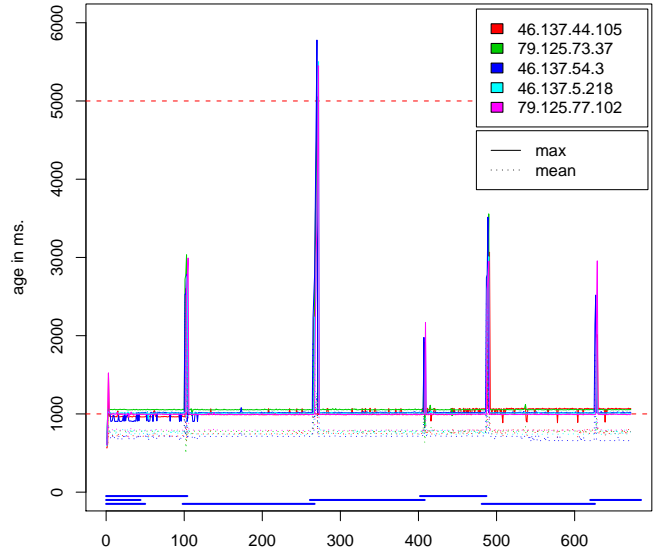
Client Data Age – 2011-06-06 03:42:32



Age in ms. of data on one client – sampling every second

(Test run 65)

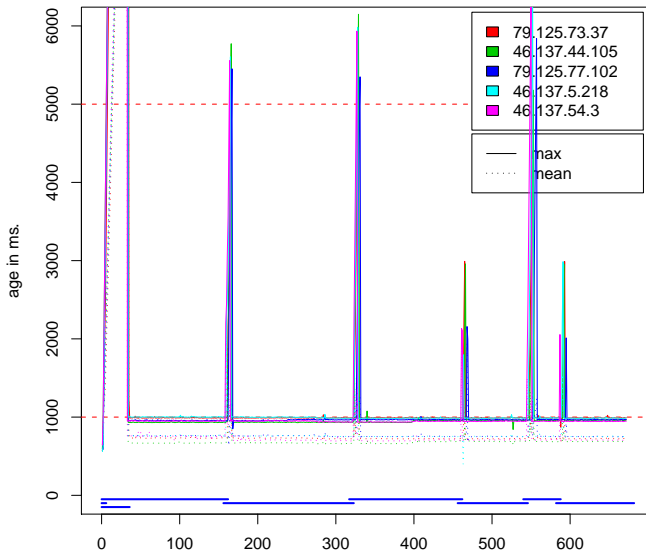
Client Data Age – 2011-06-06 03:54:46



Age in ms. of data on one client – sampling every second

(Test run 66)

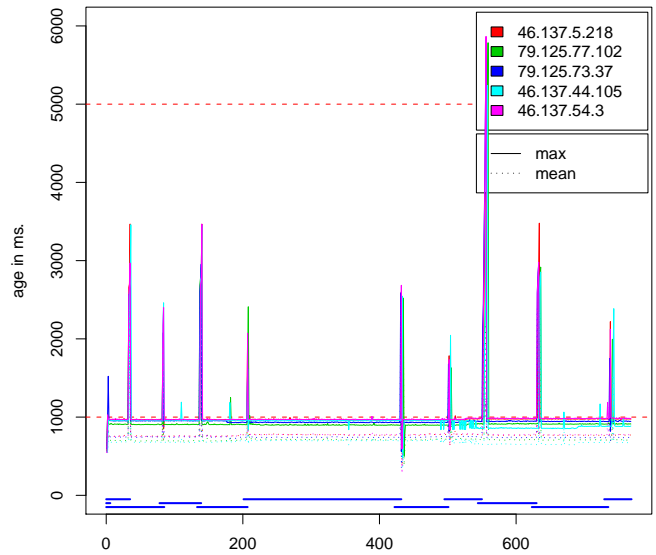
Client Data Age – 2011-06-06 04:06:53



Age in ms. of data on one client – sampling every second

(Test run 67)

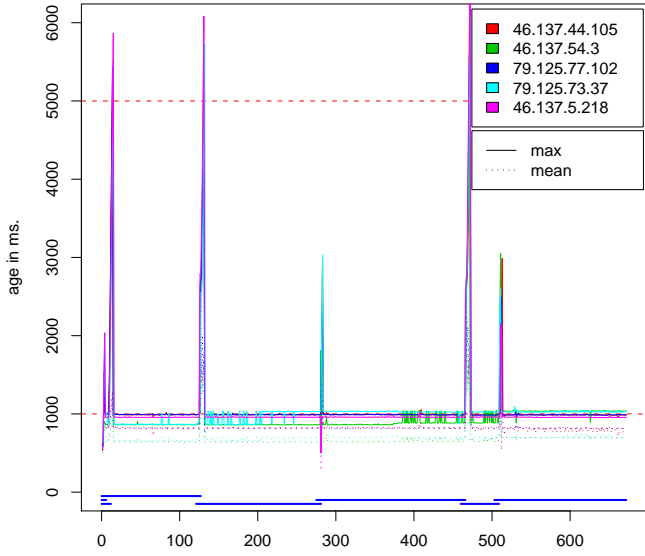
Client Data Age – 2011-06-06 04:20:40



Age in ms. of data on one client – sampling every second

(Test run 68)

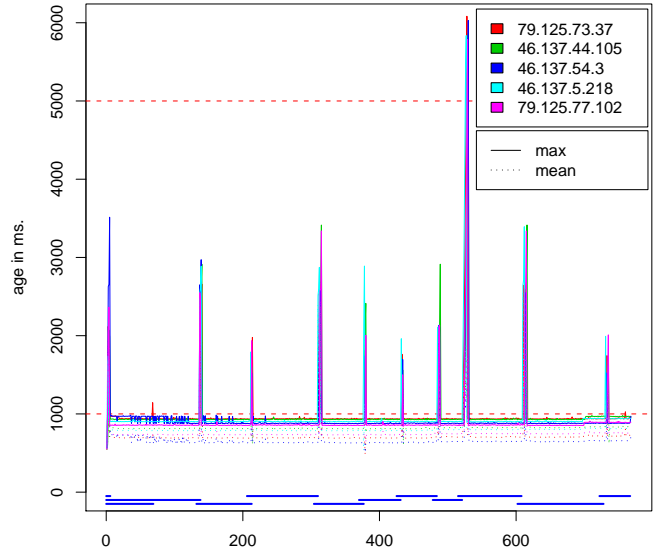
Client Data Age – 2011-06-06 04:32:42



Age in ms. of data on one client – sampling every second

(Test run 69)

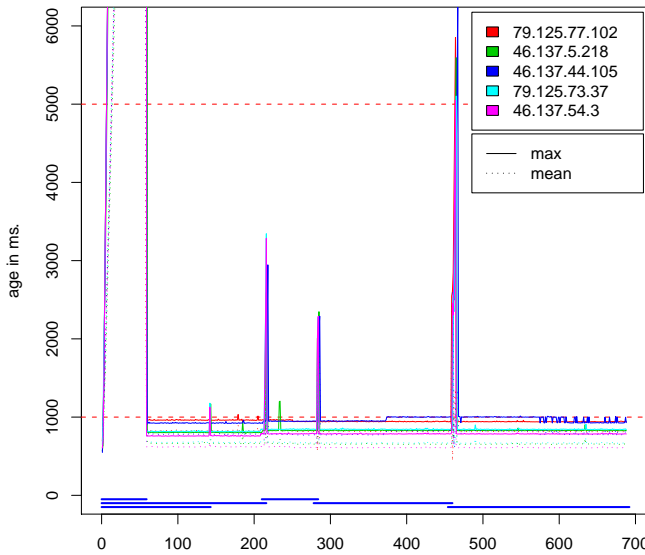
Client Data Age – 2011-06-06 04:46:19



Age in ms. of data on one client – sampling every second

(Test run 70)

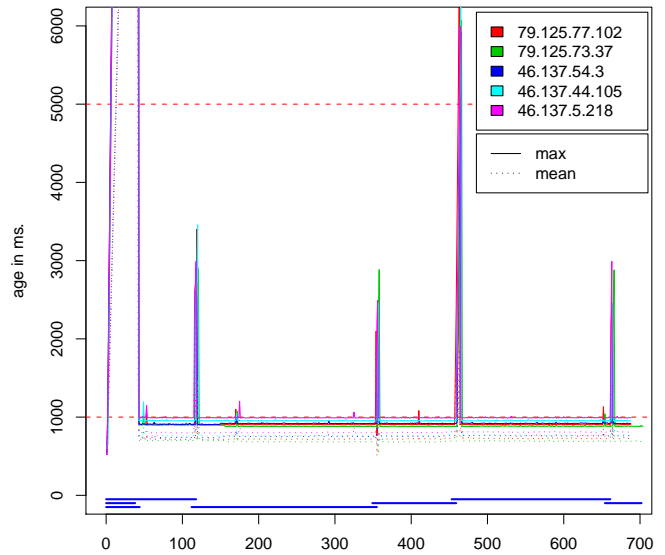
Client Data Age – 2011-06-06 04:58:37



Age in ms. of data on one client – sampling every second

(Test run 71)

Client Data Age – 2011-06-06 05:11:09

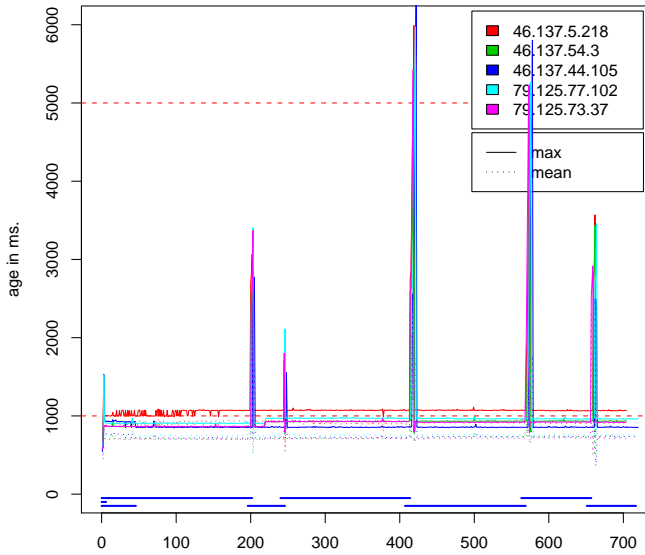


Age in ms. of data on one client – sampling every second

(Test run 72)



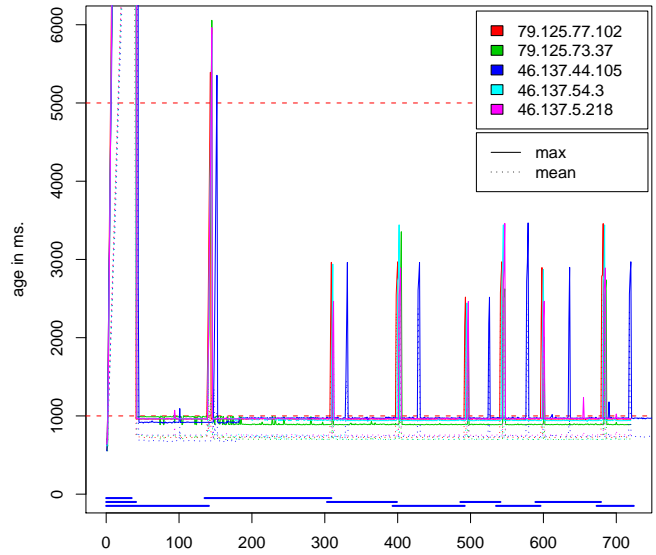
Client Data Age – 2011-06-06 05:23:53



Age in ms. of data on one client – sampling every second

(Test run 73)

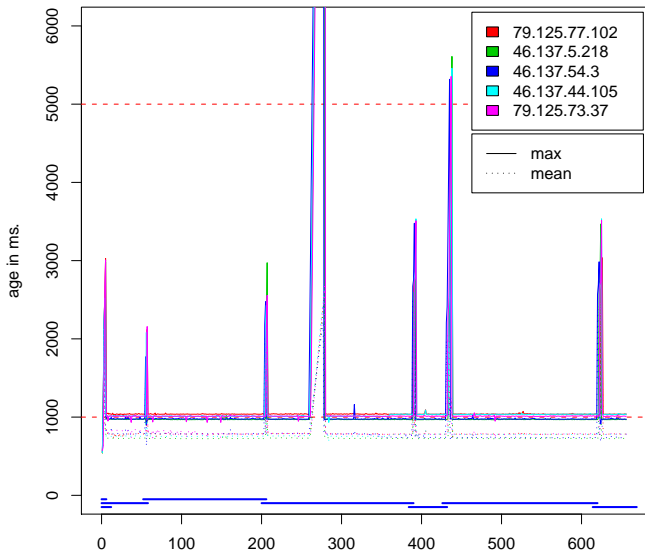
Client Data Age – 2011-06-06 05:36:45



Age in ms. of data on one client – sampling every second

(Test run 74)

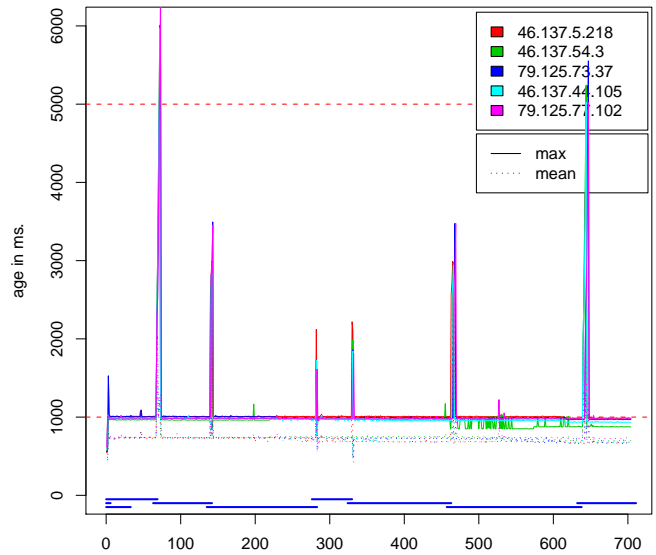
Client Data Age – 2011-06-06 05:48:40



Age in ms. of data on one client – sampling every second

(Test run 75)

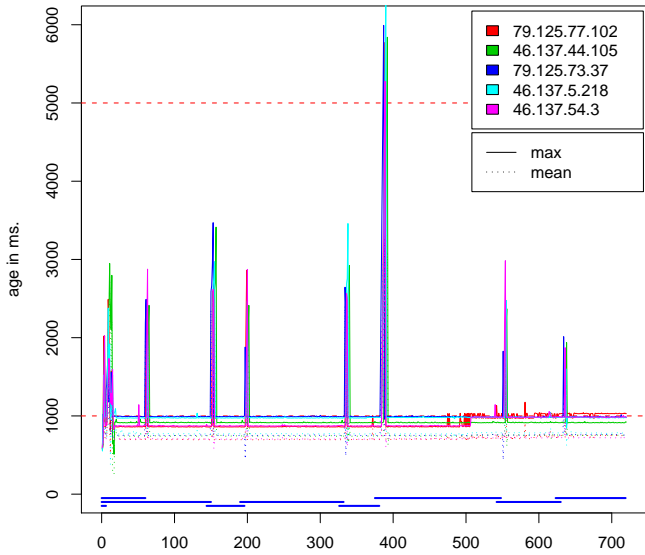
Client Data Age – 2011-06-06 06:01:19



Age in ms. of data on one client – sampling every second

(Test run 76)

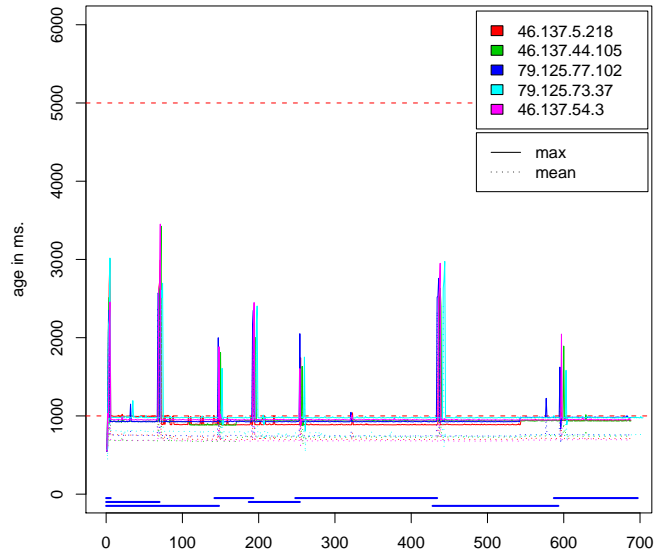
Client Data Age – 2011-06-06 06:14:06



Age in ms. of data on one client – sampling every second

(Test run 77)

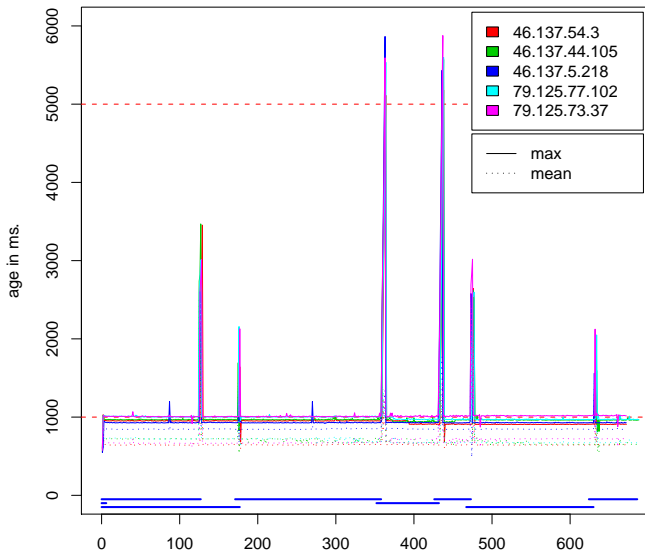
Client Data Age – 2011-06-06 06:26:30



Age in ms. of data on one client – sampling every second

(Test run 78)

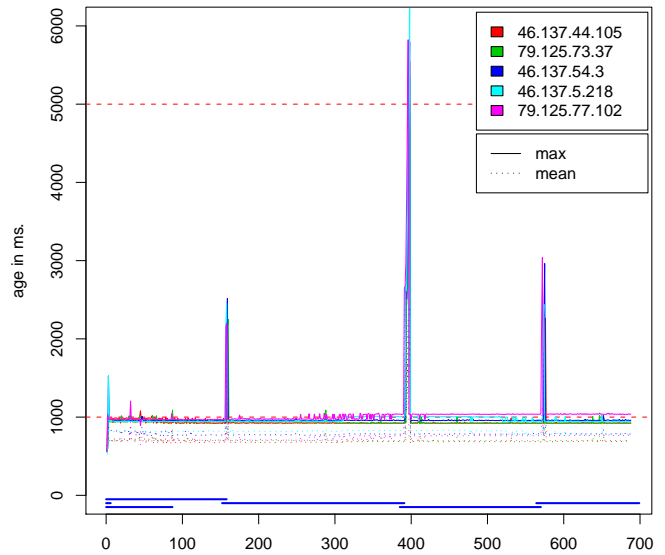
Client Data Age – 2011-06-06 06:38:41



Age in ms. of data on one client – sampling every second

(Test run 79)

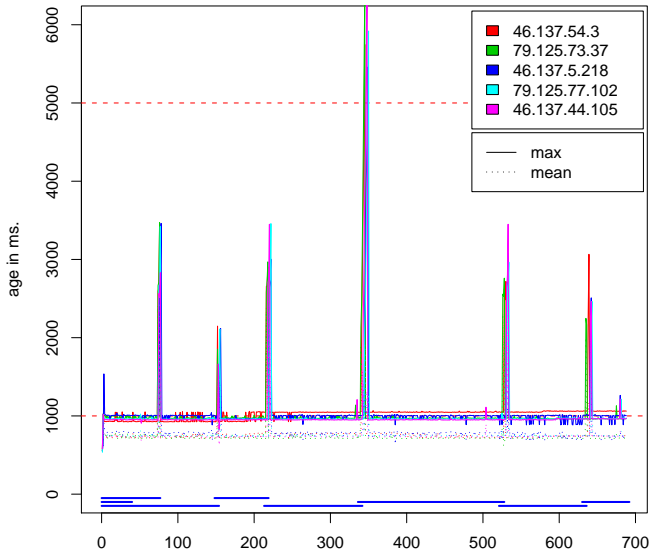
Client Data Age – 2011-06-06 06:51:07



Age in ms. of data on one client – sampling every second

(Test run 80)

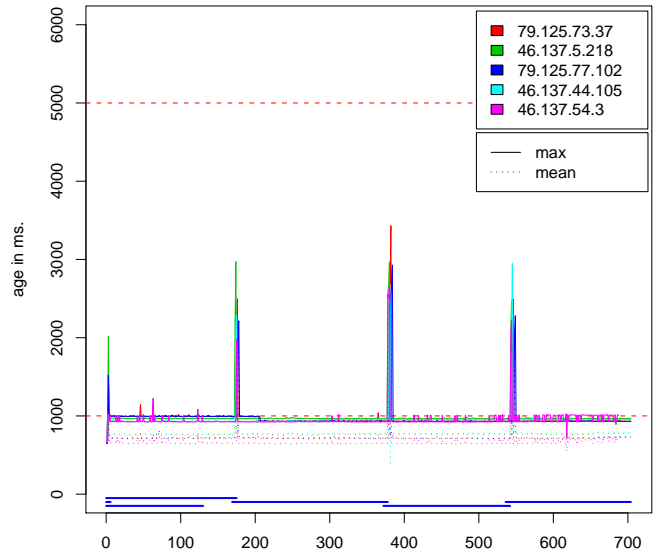
Client Data Age – 2011-06-06 07:03:25



Age in ms. of data on one client – sampling every second

(Test run 81)

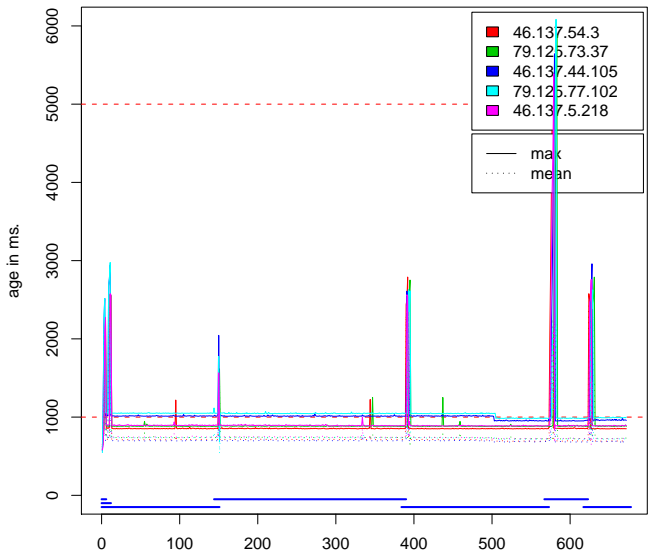
Client Data Age – 2011-06-06 07:15:54



Age in ms. of data on one client – sampling every second

(Test run 82)

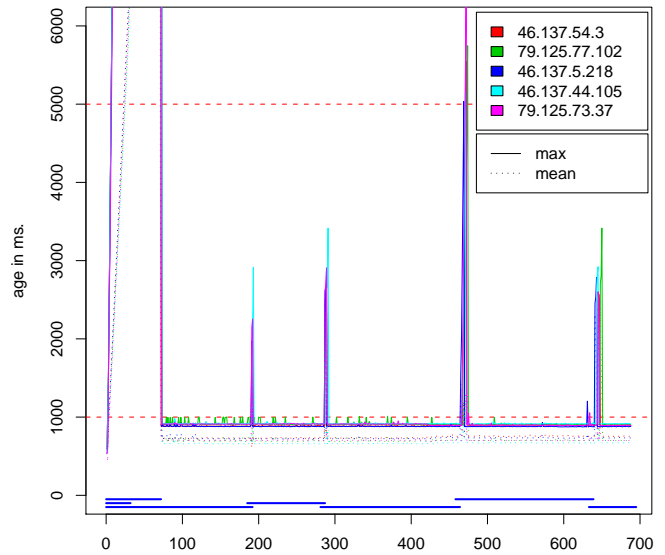
Client Data Age – 2011-06-06 07:27:58



Age in ms. of data on one client – sampling every second

(Test run 83)

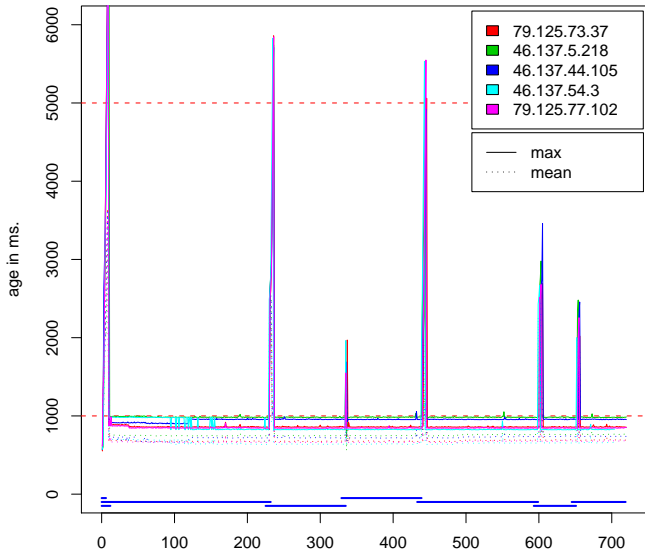
Client Data Age – 2011-06-06 07:40:18



Age in ms. of data on one client – sampling every second

(Test run 84)

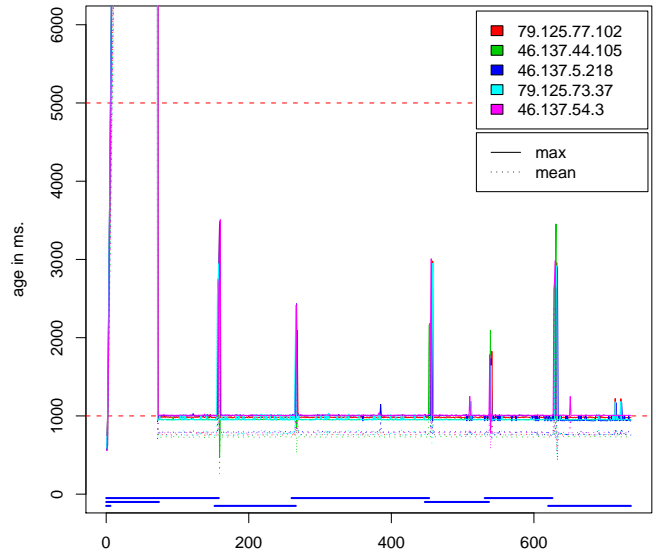
Client Data Age – 2011-06-06 07:53:03



Age in ms. of data on one client – sampling every second

(Test run 85)

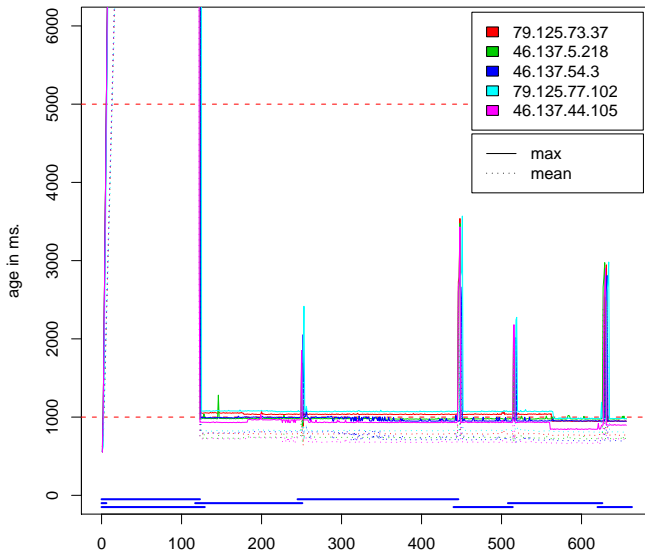
Client Data Age – 2011-06-06 08:06:08



Age in ms. of data on one client – sampling every second

(Test run 86)

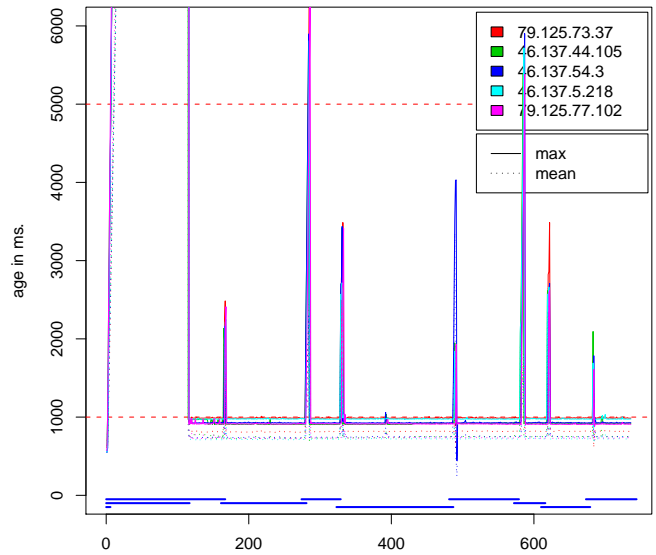
Client Data Age – 2011-06-06 08:17:57



Age in ms. of data on one client – sampling every second

(Test run 87)

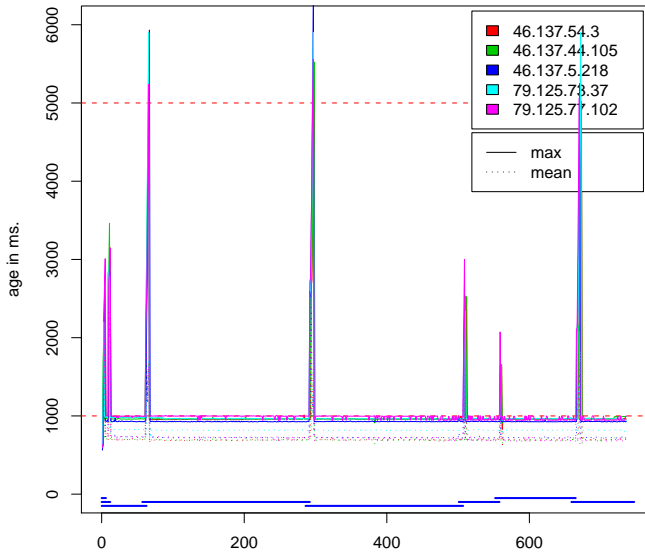
Client Data Age – 2011-06-06 08:31:07



Age in ms. of data on one client – sampling every second

(Test run 88)

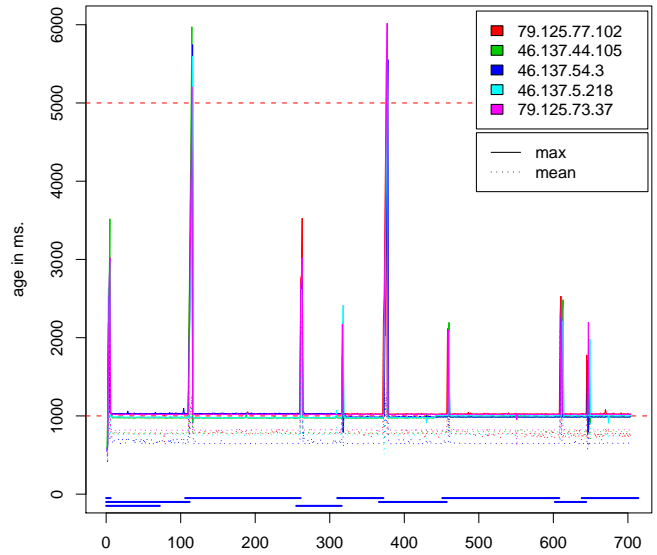
Client Data Age – 2011-06-06 08:44:20



Age in ms. of data on one client – sampling every second

(Test run 89)

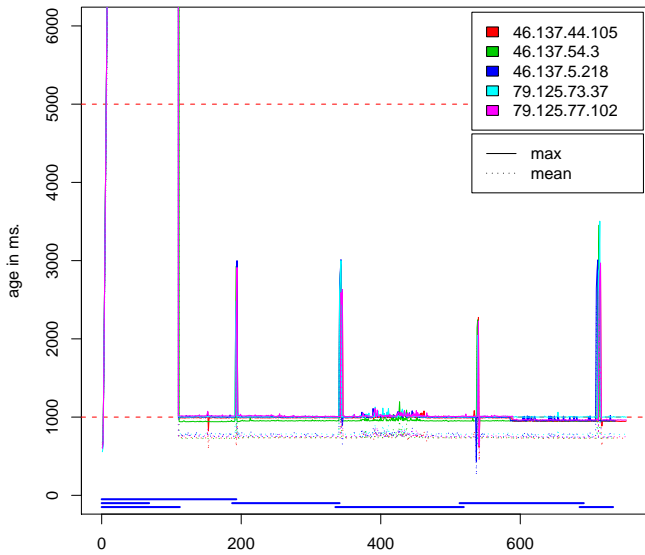
Client Data Age – 2011-06-06 08:56:59



Age in ms. of data on one client – sampling every second

(Test run 90)

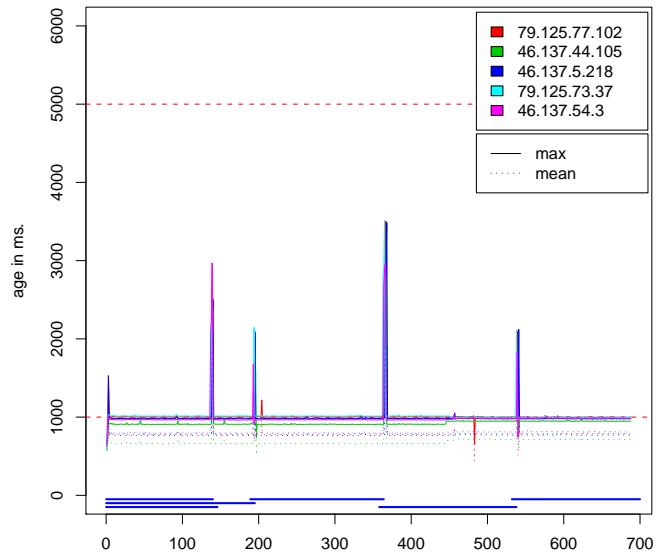
Client Data Age – 2011-06-06 09:09:53



Age in ms. of data on one client – sampling every second

(Test run 91)

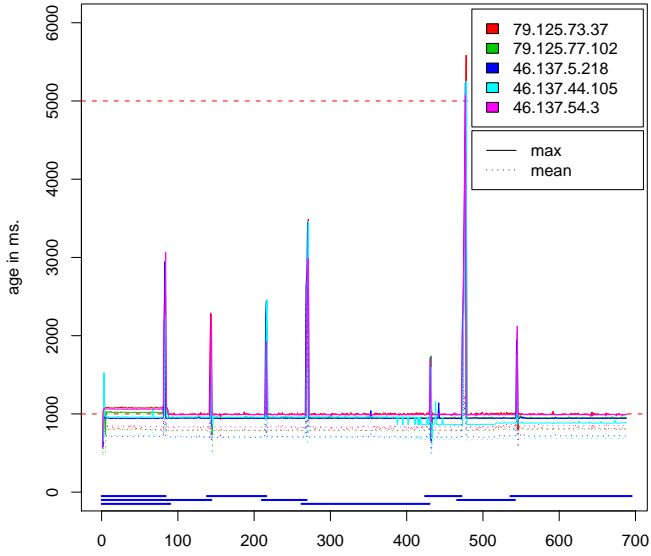
Client Data Age – 2011-06-06 09:22:19



Age in ms. of data on one client – sampling every second

(Test run 92)

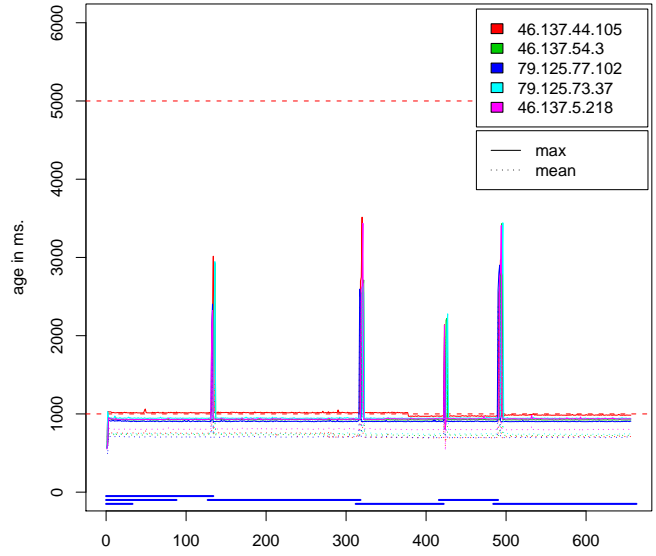
Client Data Age – 2011-06-06 09:34:38



Age in ms. of data on one client – sampling every second

(Test run 93)

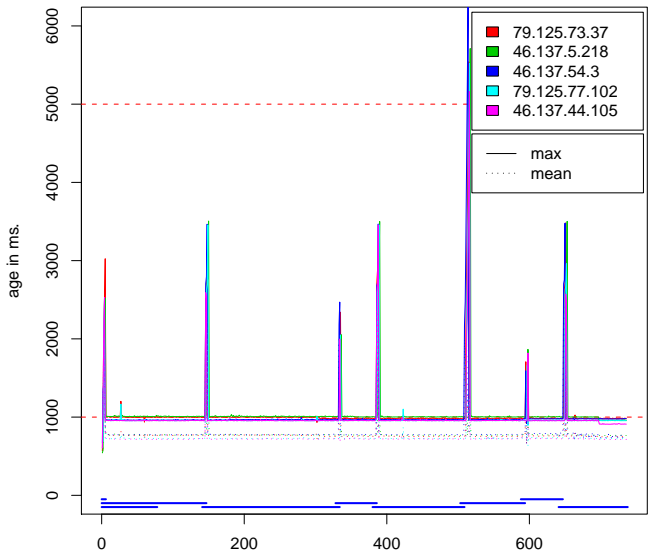
Client Data Age – 2011-06-06 09:46:27



Age in ms. of data on one client – sampling every second

(Test run 94)

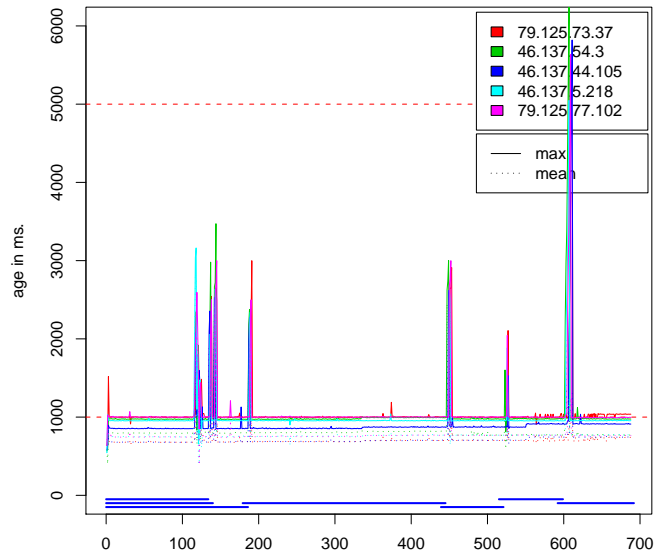
Client Data Age – 2011-06-06 09:59:31



Age in ms. of data on one client – sampling every second

(Test run 95)

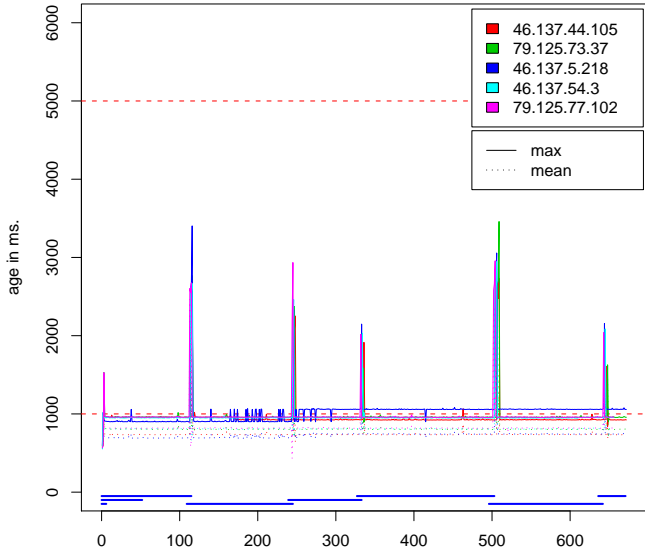
Client Data Age – 2011-06-06 10:11:49



Age in ms. of data on one client – sampling every second

(Test run 96)

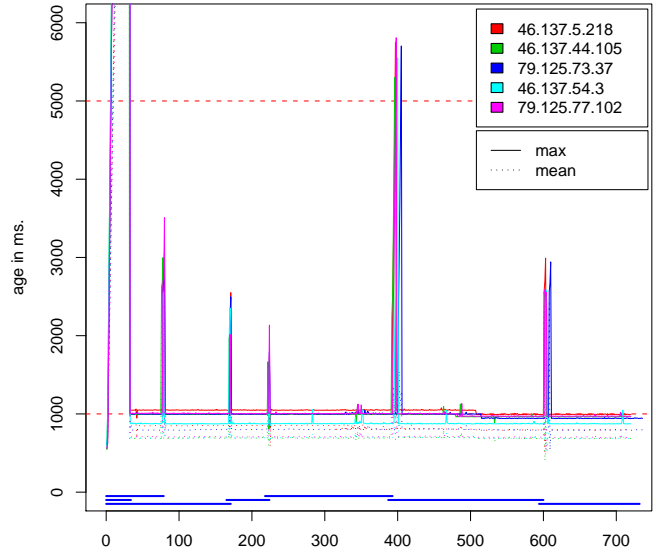
Client Data Age – 2011-06-06 10:23:46



Age in ms. of data on one client – sampling every second

(Test run 97)

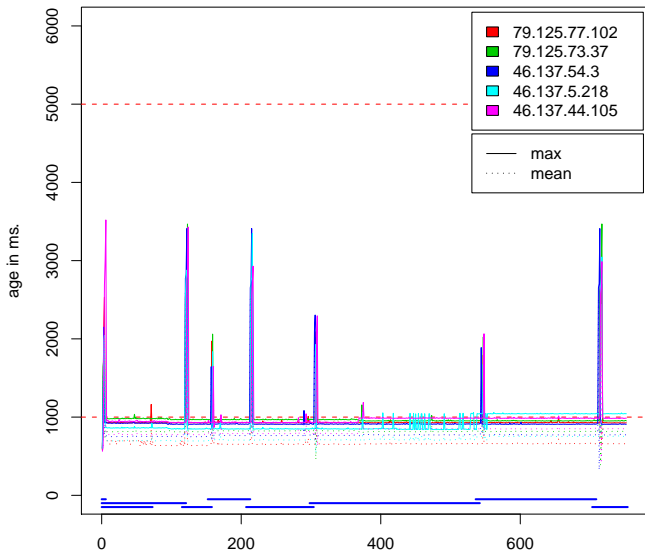
Client Data Age – 2011-06-06 10:36:39



Age in ms. of data on one client – sampling every second

(Test run 98)

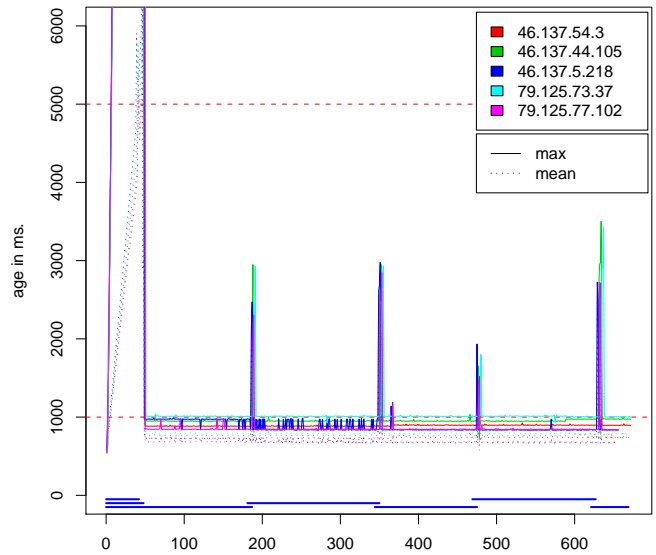
Client Data Age – 2011-06-06 10:49:59



Age in ms. of data on one client – sampling every second

(Test run 99)

Client Data Age – 2011-06-06 11:01:55



Age in ms. of data on one client – sampling every second

(Test run 100)