Job Efficiency

Adam Lyon

25 March 2009

Abstract

Examine job efficiency.

File	/Users/adam/Development/SamGrid/jobEfficiency/jobEff.Rnw
Branch	master
Tag	d0gdpi20090326
Local Modifications	

Commit	Who	Date	What
c902256 (tag: d0gdpi20090326, heads/master)	Adam Lyon	Thu Mar 26 02:30:38 2009 -0500	Basic job timings
be791b1	Adam Lyon	Wed Mar 25 17:02:40 2009 -0500	First import

1 Preliminaries

Load in the data from the samgrid monitoring.

```
> z <- gzcon(url("http://www-d0.fnal.gov/~samgrid/jobStats/samgfwd04.fnal.gov_jobXmlSummary.gz"))
> raw <- textConnection(readLines(z))
> close(z)
> d <- read.table(raw, head = T)
> close(raw)
```

Fix the time columns to something readable.

```
> mkTime <- function(col) {
      tmp <- as.integer(col)
      class(tmp) <- c("POSIXt", "POSIXct")
      tmp
  }
> d$last_status_ptime <- mkTime(d$last_status_time)
> d$last_progress_ptime <- mkTime(d$last_progress_time)
> d$last_fileTransfer_ptime <- mkTime(d$last_fileTransfer_time)
> d$createdp <- mkTime(d$created)
> d$last_progress_stime <- as.character(d$last_progress_ptime)

> nrow(d)
[1] 2762

> names(d)
 [1] "lid"                      "nDeclared"                "last_fileTransfer_fcp"    "facility"
 [5] "last_fileTransfer_source" "kernelCpu"                "last_status_time"         "last_progress_time"
 [9] "userCpu"                  "created"                  "last_fileTransfer_time"   "gid"
[13] "last_progress"            "last_status"              "effCpu"                   "nStored"
[17] "last_exitCode"            "wallTime"                 "targetedResource"         "last_fileTransfer_status"
[21] "last_status_ptime"        "last_progress_ptime"      "last_fileTransfer_ptime"  "createdp"
[25] "last_progress_stime"
```

```
> head(d)
     lid nDeclared last_fileTransfer_fcp           facility last_fileTransfer_source kernelCpu last_status_time
1 466944         0               cms_raw d0cabosg2.fnal.gov        d0srv071.fnal.gov   Unknown       1238021996
2 466945         0               cms_raw d0cabosg2.fnal.gov        d0srv072.fnal.gov   Unknown       1238021998
3 466946         0               cms_raw d0cabosg2.fnal.gov        d0srv077.fnal.gov   Unknown       1238021996
4 466947         0               cms_raw d0cabosg2.fnal.gov        d0srv077.fnal.gov   Unknown       1238021998
5 466948         0               cms_raw d0cabosg2.fnal.gov        d0srv077.fnal.gov   Unknown       1238021996
6 466949         0               cms_raw d0cabosg2.fnal.gov        d0srv071.fnal.gov   Unknown       1238021998
  last_progress_time userCpu    created last_fileTransfer_time                                     gid
1         1238023297 Unknown 1238014824             1238023139 diesburg_d0mino01.fnal.gov_155523_21917
2         1238024533 Unknown 1238014824             1238024391 diesburg_d0mino01.fnal.gov_155513_21878
3         1238025491 Unknown 1238014824             1238025174 diesburg_d0mino01.fnal.gov_155523_21917
4         1238024323 Unknown 1238014825             1238024141 diesburg_d0mino01.fnal.gov_155513_21878
5         1238023663 Unknown 1238014825             1238023435 diesburg_d0mino01.fnal.gov_155523_21917
6         1238024690 Unknown 1238014825             1238024403 diesburg_d0mino01.fnal.gov_155513_21878
                    last_progress last_status  effCpu nStored last_exitCode wallTime   targetedResource
1 Starting task: reconstructed.sh      active Unknown       0            -1  Unknown d0cabosg2.fnal.gov
2 Starting task: reconstructed.sh      active Unknown       0            -1  Unknown d0cabosg2.fnal.gov
3 Starting task: reconstructed.sh      active Unknown       0            -1  Unknown d0cabosg2.fnal.gov
4 Starting task: reconstructed.sh      active Unknown       0            -1  Unknown d0cabosg2.fnal.gov
5 Starting task: reconstructed.sh      active Unknown       0            -1  Unknown d0cabosg2.fnal.gov
6 Starting task: reconstructed.sh      active Unknown       0            -1  Unknown d0cabosg2.fnal.gov
  last_fileTransfer_status   last_status_ptime last_progress_ptime last_fileTransfer_ptime            createdp
1                COMPLETED 2009-03-25 17:59:56 2009-03-25 18:21:37     2009-03-25 18:18:59 2009-03-25 16:00:23
2                COMPLETED 2009-03-25 17:59:58 2009-03-25 18:42:13     2009-03-25 18:39:51 2009-03-25 16:00:24
3                COMPLETED 2009-03-25 17:59:56 2009-03-25 18:58:11     2009-03-25 18:52:54 2009-03-25 16:00:24
4                COMPLETED 2009-03-25 17:59:58 2009-03-25 18:38:43     2009-03-25 18:35:41 2009-03-25 16:00:24
5                COMPLETED 2009-03-25 17:59:56 2009-03-25 18:27:43     2009-03-25 18:23:55 2009-03-25 16:00:25
6                COMPLETED 2009-03-25 17:59:58 2009-03-25 18:44:50     2009-03-25 18:40:03 2009-03-25 16:00:25
  last_progress_stime
1 2009-03-25 18:21:37
2 2009-03-25 18:42:13
3 2009-03-25 18:58:11
```

```
4 2009-03-25 18:38:43
5 2009-03-25 18:27:43
6 2009-03-25 18:44:50
```

Let's remove the entries that don't have a progress time.

```
> noProgress <- d[d$last_progress_time == -1, ]
> d <- d[d$last_progress_time > -1, ]
> noProgress
        lid nDeclared last_fileTransfer_fcp facility last_fileTransfer_source kernelCpu last_status_time
2531 466710         0               Unknown  Unknown                  Unknown   Unknown       1238015658
     last_progress_time userCpu    created last_fileTransfer_time                                    gid last_progress
2531                 -1 Unknown 1237997618                     -1 diesburg_d0mino01.fnal.gov_105635_28864       Unknown
     last_status  effCpu nStored last_exitCode wallTime   targetedResource last_fileTransfer_status   last_status_ptime
2531      active Unknown       0            -1  Unknown d0cabosg2.fnal.gov                  Unknown 2009-03-25 16:14:17
     last_progress_ptime last_fileTransfer_ptime            createdp last_progress_stime
2531 1969-12-31 17:59:59     1969-12-31 17:59:59 2009-03-25 11:13:37 1969-12-31 17:59:59

> nrow(d)
[1] 2761
```

Note that the times for `last_status_time` are messed up because R is treating the times as factors. The right way to fix that is to just save an integer from the xmldb program.

What is the range of dates for jobs?

```
> range(d$last_progress_ptime)
[1] "2009-03-20 23:55:32 CDT" "2009-03-26 00:22:06 CDT"
```

What is the break down of active vs. running jobs? `job_status` is not always accurate. Jobs that are marked as active may have an exit code. And jobs that don't have an exit code may in fact be done. Probably the most reliable test is to check the `last_progress`.

```
> print(xtable(table(d$last_progress, d$last_exitCode)), floating = F)
```

	-1	0	1
Finished setup, launching d0runjob	2	0	0
Finished staging input file all_2_0000250060_068.raw	1	0	0
Finished staging input file all_2_0000250060_140.raw	1	0	0
Finished staging input file all_2_0000250060_196.raw	1	0	0
Finished staging job files	55	0	0
Starting task: reconstructed.sh	1983	0	0
Unknown	0	0	0
User executable exited with code 0	0	710	0
User executable exited with code 1	0	0	8

Look at the guys that have exit code of 1

```
> b <- d[d$last_exitCode == 1, c("gid", "lid", "nDeclared", "nStored", "last_progress_stime",
      "wallTime")]
> print(xtable(b), floating = F)
```

	gid	lid	nDeclared	nStored	last_progress_stime	wallTime
201	diesburg_d0mino01.fnal.gov_171652_30428	467144	0	0	2009-03-26 00:12:47	5:10:26
330	diesburg_d0mino01.fnal.gov_171725_30555	467274	0	0	2009-03-25 22:13:37	1:56:22
909	diesburg_d0mino01.fnal.gov_225711_20082	462977	0	0	2009-03-25 22:37:41	44:48:57
1073	diesburg_d0mino01.fnal.gov_115054_6053	464355	0	0	2009-03-25 18:07:40	22:22:13
1120	diesburg_d0mino01.fnal.gov_152145_26936	464428	0	0	2009-03-25 23:51:33	27:20:23
1339	diesburg_d0mino01.fnal.gov_015417_6788	465236	0	0	2009-03-25 19:09:05	15:13:59
1341	diesburg_d0mino01.fnal.gov_015904_6927	465238	0	0	2009-03-25 23:40:59	19:44:07
2076	diesburg_d0mino01.fnal.gov_074332_13280	466217	0	0	2009-03-25 18:27:47	6:37:44

2 Job Progress

Let's look at job progress information.

```
> z <- gzcon(url("http://www-d0.fnal.gov/~samgrid/jobStats/samgfwd04.fnal.gov_jobXmlProgress.gz"))
> raw <- textConnection(readLines(z))
> close(z)
> p <- read.table(raw, head = T)
> close(raw)

> mkTime <- function(col) {
      tmp <- as.integer(col)
      class(tmp) <- c("POSIXt", "POSIXct")
      tmp
  }
> p$startTimeP <- mkTime(p$startTime)
> p$startTimeS <- as.character(p$startTimeP)

> nrow(p)
[1] 13028

> head(p)
                                    gid    lid         what  startTime elapsed name   size                  app
1 diesburg_d0mino01.fnal.gov_155523_21917 466944    toPending 1238014824 1747.78   NA     NA                 <NA>
2 diesburg_d0mino01.fnal.gov_155523_21917 466944     toLaunch 1238016572 4626.33   NA     NA                 <NA>
3 diesburg_d0mino01.fnal.gov_155523_21917 466944 fileTransfer 1238021202   57.00   NA 600852         binary_fetch
4 diesburg_d0mino01.fnal.gov_155523_21917 466944 fileTransfer 1238022995  144.00   NA 637380 dzero_reconstruction
5 diesburg_d0mino01.fnal.gov_155513_21878 466945    toPending 1238014824 1747.69   NA     NA                 <NA>
6 diesburg_d0mino01.fnal.gov_155513_21878 466945     toLaunch 1238016572 4708.23   NA     NA                 <NA>
      fcp            source          startTimeP          startTimeS
1    <NA>              <NA> 2009-03-25 16:00:23 2009-03-25 16:00:23
2    <NA>              <NA> 2009-03-25 16:29:31 2009-03-25 16:29:31
3 cms_rte d0srv071.fnal.gov 2009-03-25 17:46:42 2009-03-25 17:46:42
4 cms_raw d0srv071.fnal.gov 2009-03-25 18:16:35 2009-03-25 18:16:35
5    <NA>              <NA> 2009-03-25 16:00:24 2009-03-25 16:00:24
6    <NA>              <NA> 2009-03-25 16:29:31 2009-03-25 16:29:31
```

Let's plot some timings.

Plot the file delivery time (see Fig. 1 on the following page).

```
> print(histogram(~elapsed/60/60 | source * fcp, data = p, breaks = 20, subset = what == "fileTransfer",
      type = "count", xlab = "Hours"))
```

Let's look at d0srv071 for files that take more than 10 minutes to deliver (see Fig. 2 on page 9).

```
> print(histogram(~elapsed/60/60, data = p, breaks = 20, subset = what == "fileTransfer" & source ==
      "d0srv071.fnal.gov" & elapsed > 10 * 60, type = "count", xlab = "Hours"))
```

Get counts by 30 minutes for d0srv071 and raw fcp queue.

```
> p71 <- with(p, p[what == "fileTransfer" & source == "d0srv071.fnal.gov" & fcp == "cms_raw",
      ])
> nrow(p71)
[1] 886

> table(p71$elapsed > 10 * 60)/nrow(p71)
    FALSE      TRUE
0.2573363 0.7426637

> table(p71$elapsed > 60 * 60)/nrow(p71)
    FALSE      TRUE
0.6862302 0.3137698

> table(p71$elapsed > 2 * 60 * 60)/nrow(p71)
     FALSE       TRUE
0.95485327 0.04514673
```

Let's look at other things

How many hours does Reco run (see Fig. 3 on page 10).

```
> print(histogram(~elapsed/60/60, data = p, subset = what == "runReco", breaks = 50, type = "count",
      xlab = "Hours"))
```

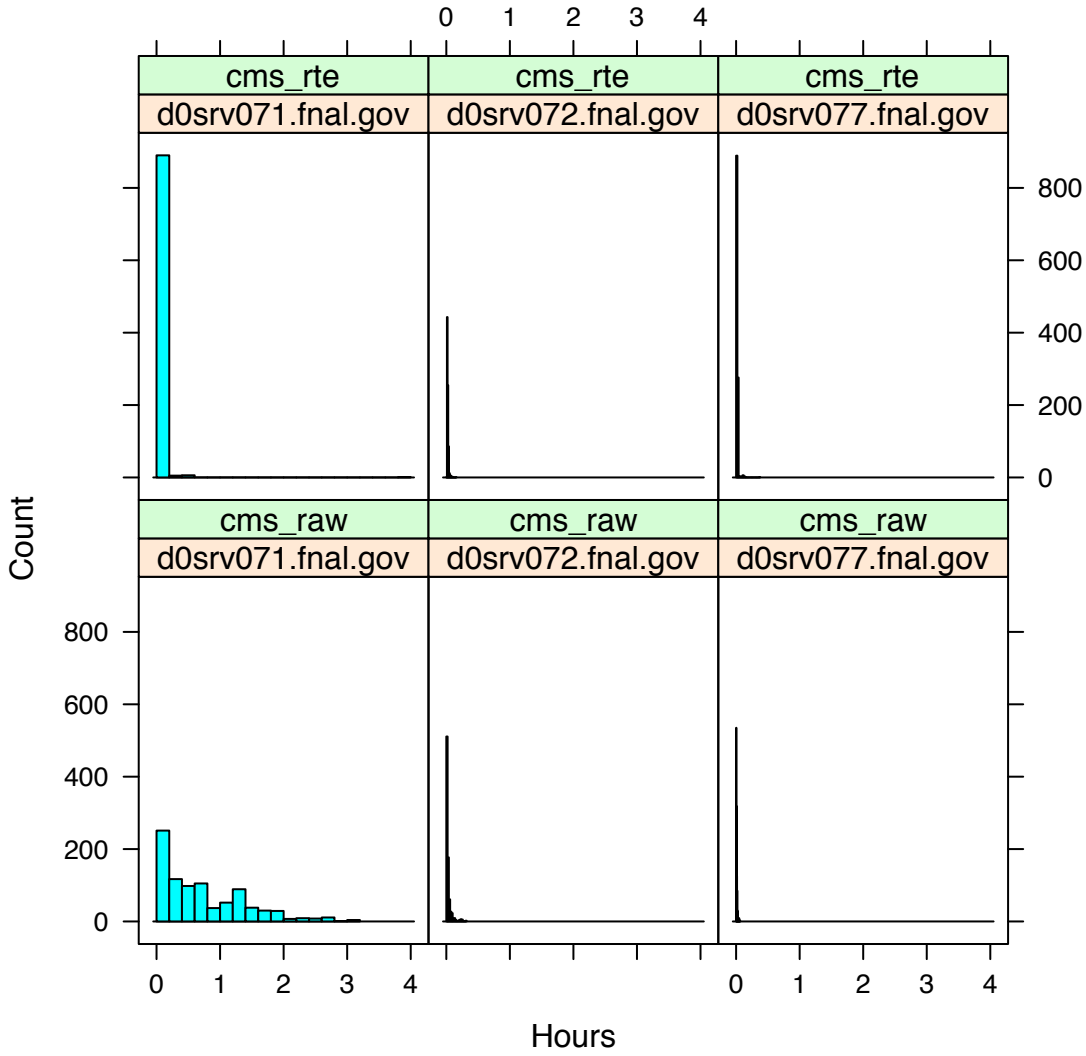

Figure 1: File delivery time

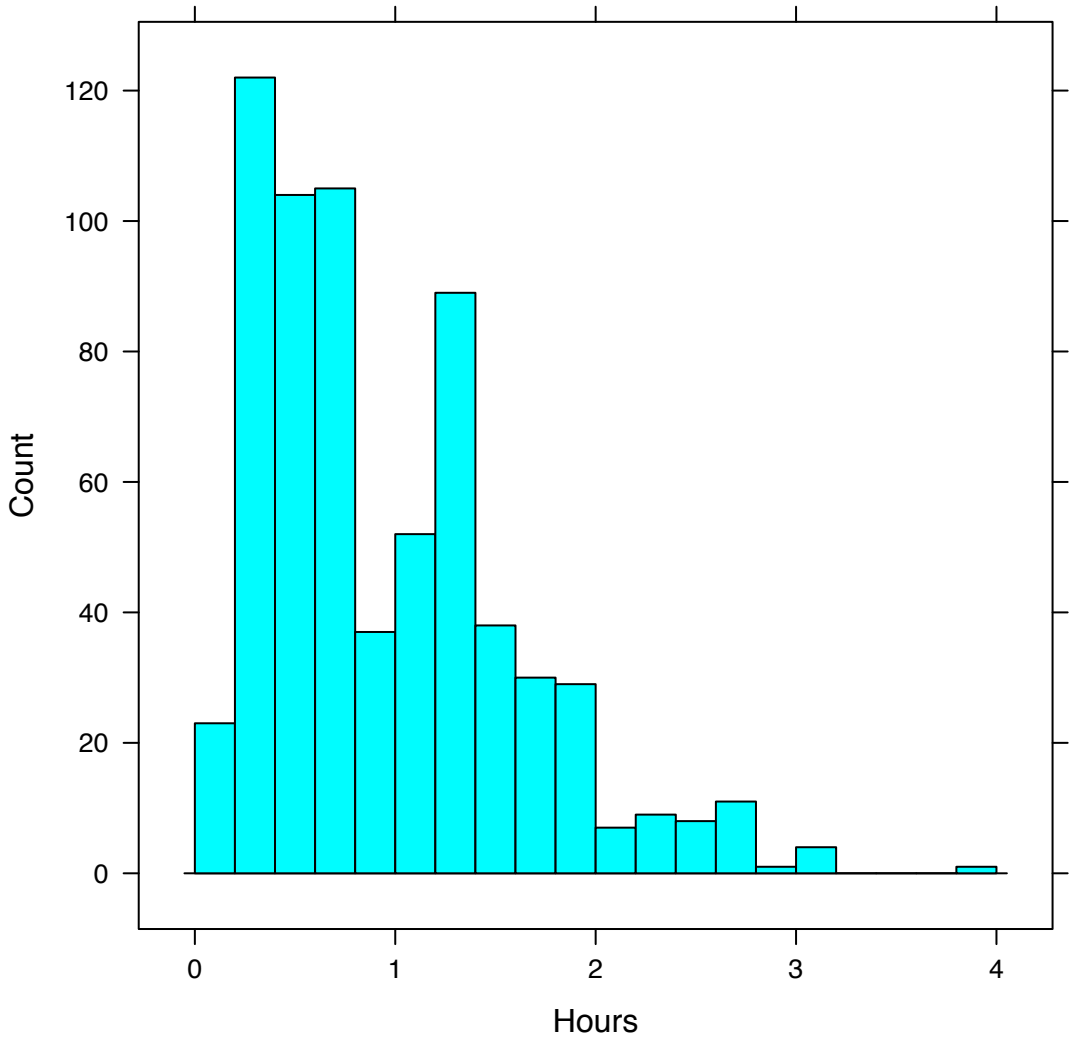

Figure 2: Slow file deliveries (more than 10 minutes) from d0srv071.fnal.gov

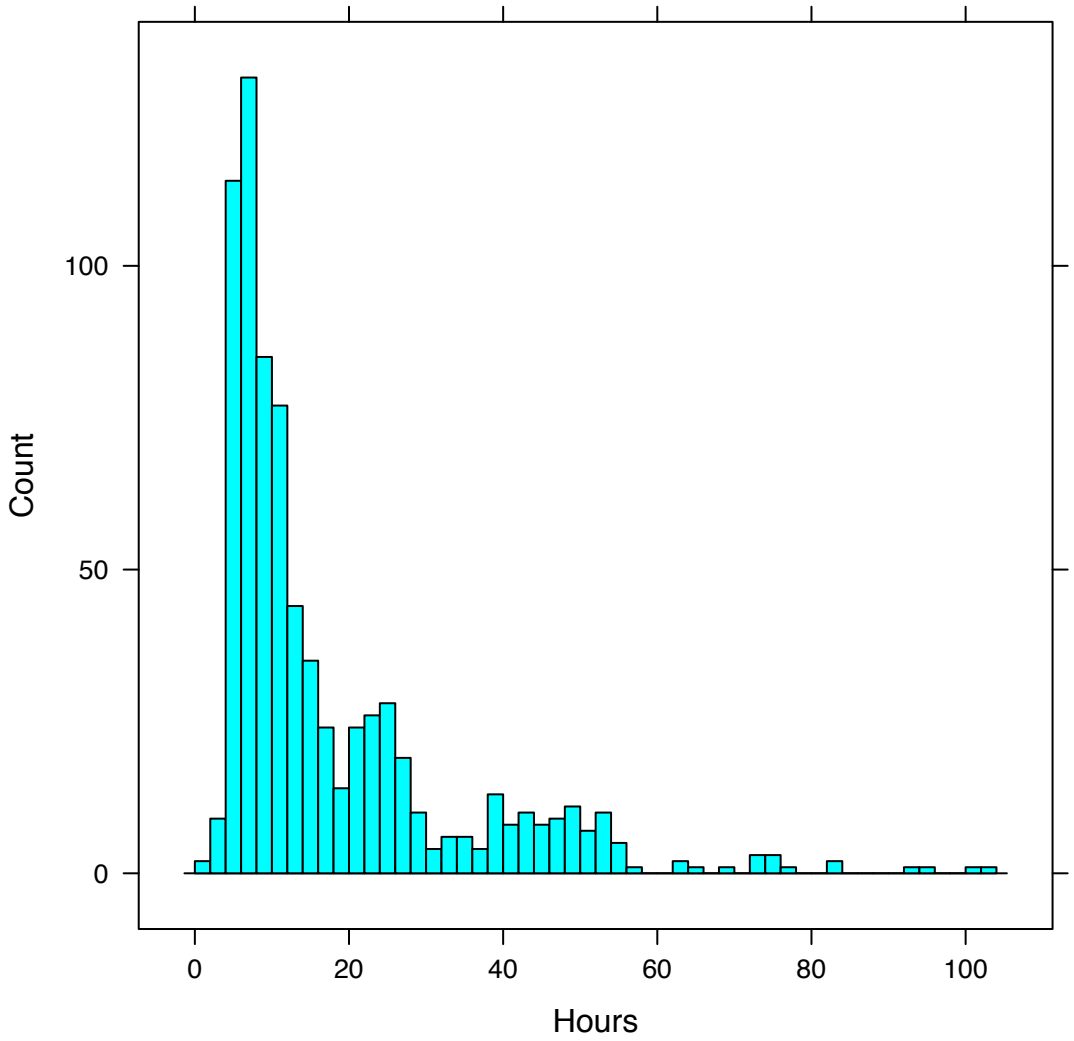

Figure 3: Reco running time

And the time to store the data back into SAM (see Fig. 4 on the next page).

```
> print(histogram(~elapsed/60, data = p, subset = what == "storeToSam" & elapsed < 60 * 10, breaks = 50,
      type = "count", xlab = "Minutes"))
```

And the time to store the data back into SAM for long transfers (see Fig. 5 on page 13).

```
> print(histogram(~elapsed/60/60, data = p, subset = what == "storeToSam" & elapsed > 60 * 10,
      breaks = 50, type = "count", xlab = "Hours"))
```

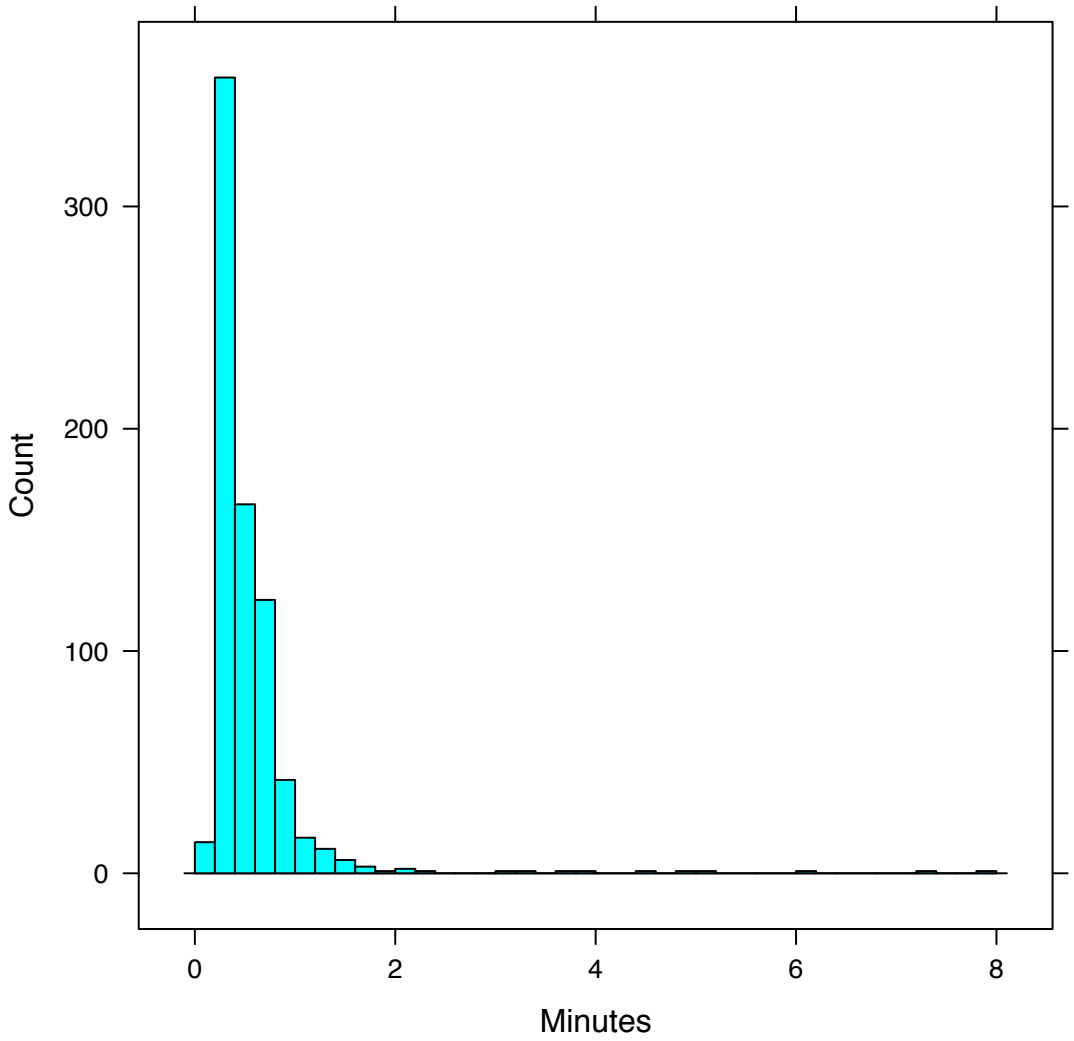

Figure 4: Time to store back into SAM

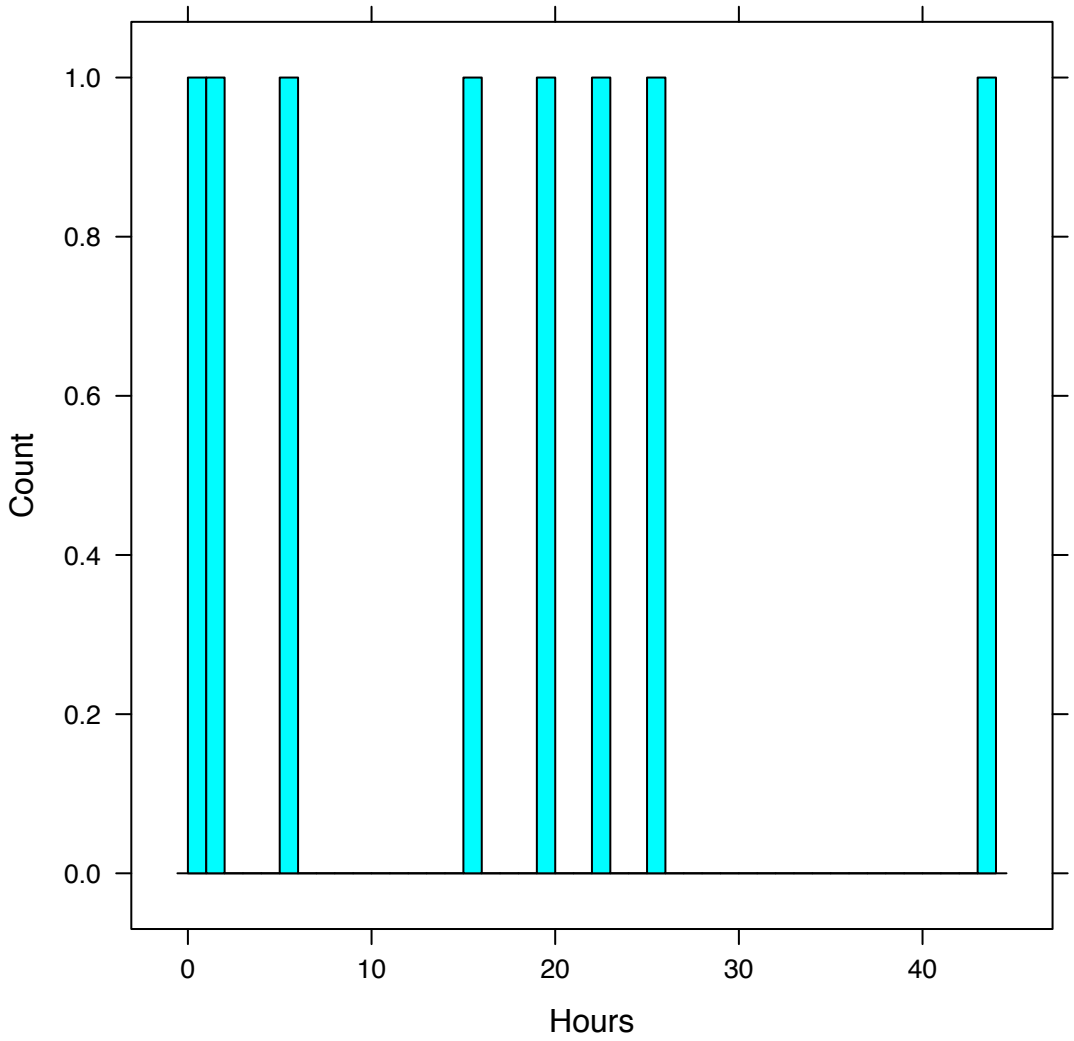

Figure 5: Time to store back into SAM for long transfers