# Benchmarking experiment flash report

György Barabás

The raw results for all first-download data from the benchmarking experiment are in Figure 1. Here we have a grid of panels, with 7 rows and 3 columns. The columns are three different servers used for downloading. The rows indicate storage platform and level of erasure coding (if applicable). Within each panel, file size is along the x-axis and retrieval time along the y-axis, both on the log scale. Colors show retrieval strategies: when erasure coding is absent, blue means NONE, whereas in the presence of erasure coding, it means DATA. Yellow indicates the RACE strategy. Each point is a single download event, and the points have been arranged to reflect the general shape of their distribution.

The main take-aways from this figure are:

- Arweave's download times increase the slowest, though it also takes longer to download small files from it.
- IPFS download times increase somewhat faster.
- Swarm increases the fastest, so for large files it is the least efficient. This was expected, given its underlying DISC model.
- Weirdly, the level of erasure coding does not appear to have much of an effect on download speeds.
- The DATA and RACE retrieval strategies do lead to differences—these, however, are not as predictable as one might have expected. For larger files, sometimes RACE is faster, sometimes DATA.

One thing looks slightly strange. For Swarm and INSANE levels of erasure coding, under the RACE strategy and for 100MB files, the data are spread out very widely. In fact, on each of the three servers, 10% of these downloads were performed suspiciously fast:

| platform | erasure | strategy | size | server | 10% percentile download time (seconds) |
|---|---|---|---|---|---|
| Swarm | INSANE | RACE | 100MB | Server 1 | 1.93 |
| Swarm | INSANE | RACE | 100MB | Server 2 | 4.19 |
| Swarm | INSANE | RACE | 100MB | Server 3 | 4.22 |

We should double-check this, because it feels strange that 100MB of data can be downloaded from Swarm in less than two seconds.

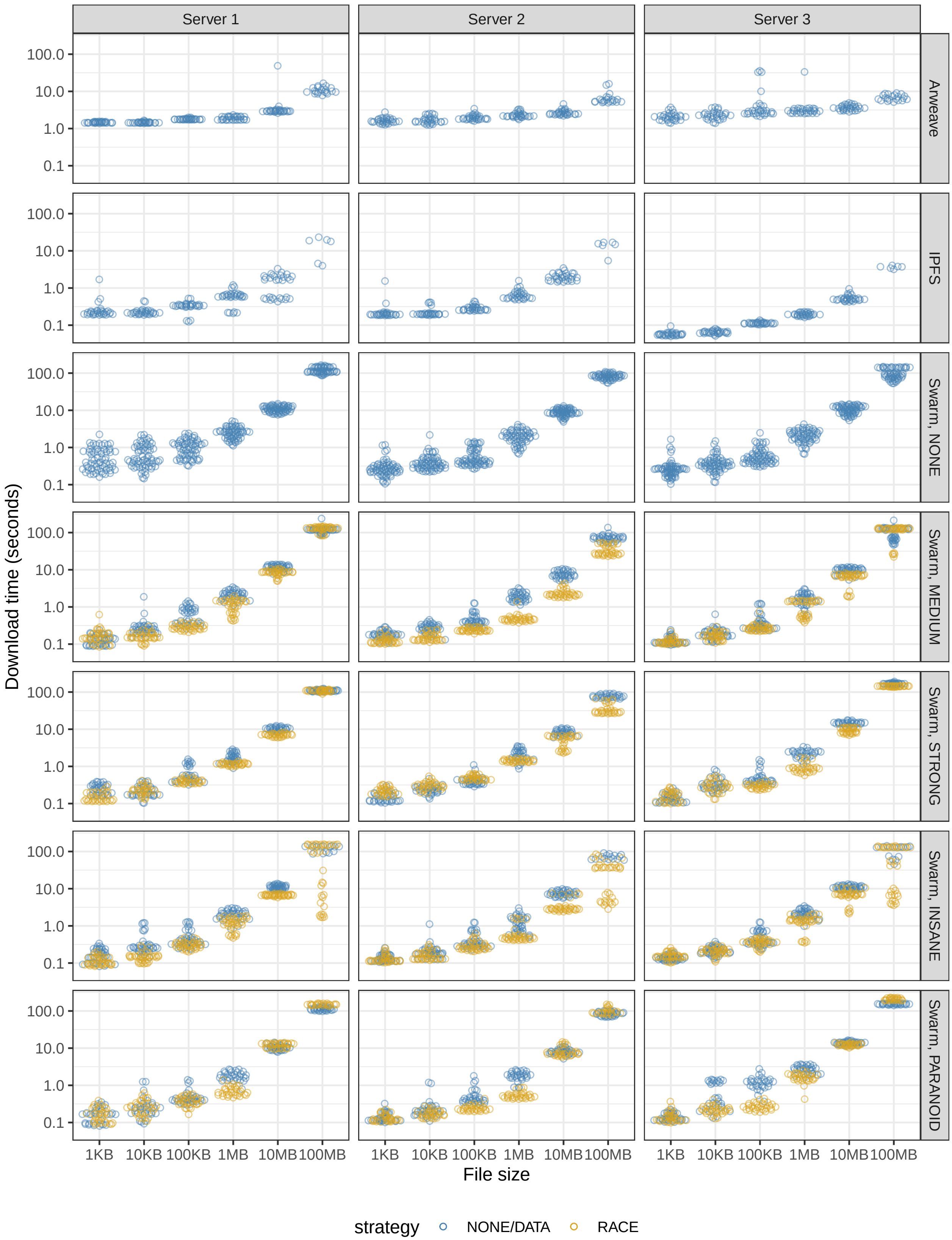


Figure 1: Empirical results from the benchmarking experiment

Otherwise, taking these data as they are, Figure 2 shows model-predicted extrapolations made from them. As seen, RACE is on average not better than DATA for large files (the dashed lines lie above the solid ones for files over 1GB). The models used in this figure are always generalized linear mixed models, but their structure is different for the three platforms. For Arweave and IPFS, the model reads

```
time ~ log(size)^3 + (1 + log(size)^3 | server)
```

with a Gaussian family and a log-link function, where `time` is the download time in seconds and `size` is the file size in kilobytes. The log-link function means we are effectively estimating log(time) as a function of $\log^3$(size), with a random intercept and slope provided by the identity of the server on which the downloads were performed. Since Swarm additionally has erasure coding and different retrieval strategies, the fitted model is also more complex, although it is still a Gaussian generalized linear mixed model with a log-link function of the form

```
time ~ log(size)^2 + erasure + strategy + log(size)^2:erasure
     + log(size)^2:strategy + erasure:strategy + (1 + erasure | server)
```

In words, we predict log(time) using the main effects of $\log^2$(size), erasure level, and strategy, and all their possible two-way interactions. Plus we add a random intercept and slope (for erasure level) based on server identity. In this model, erasure level is not a factor but a covariate reflecting the corresponding baseline probability of a faulty chunk retrieval event—that is, it's taken as 0 for no erasure coding, as 0.01 for MEDIUM, 0.05 for STRONG, 0.1 for INSANE, and 0.5 for PARANOID. (Model selection indicated this as the best choice among a few other candidates, such as using 0-1-2-3-4, or using the fraction of chunks that are parity chunks.)

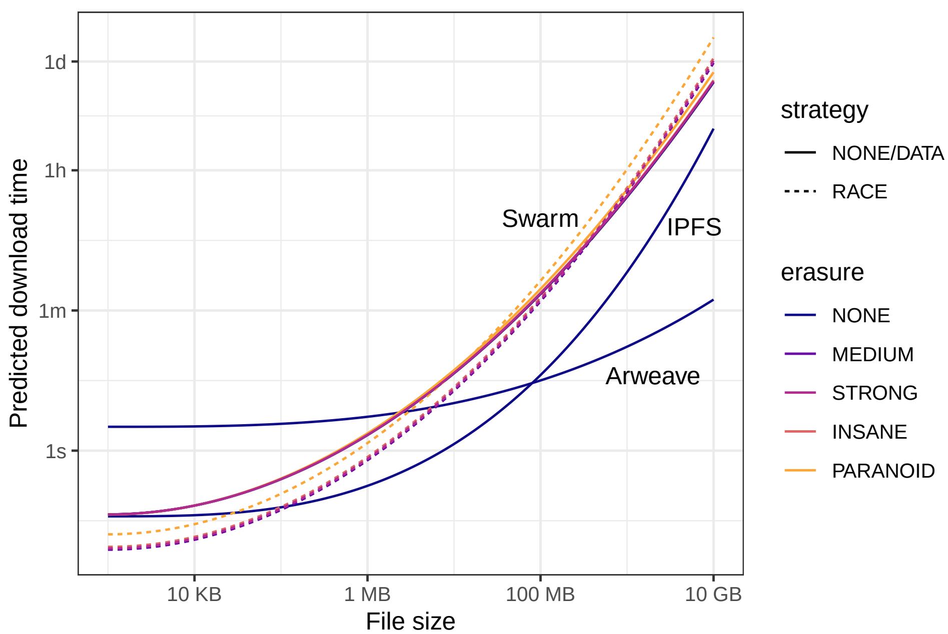


Figure 2: Model-based predictions for mean file retrieval times. Here random effects are averaged over and therefore server identity no longer plays a role.