AWS Communications – Project Report

<table>
<tr><th>Policy</th><th>Experiment</th><th>RTT(Upload)</th><th>RTT(Download)</th></tr>
<tr><td rowspan="2">random</td><td>Seoul (Exp. 1)</td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>267.279
61.35</td><td>298.893
0.04595</td><td>163.307
0.08675</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>252.106
0.07318</td><td>298.203
0.06615</td><td>163.423
0.07589</td></tr></table></td></tr>
<tr><td>Ireland/
Seoul (Exp. 2)</td>
<td><table><tr><td>Seoul</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>278.9513
75.09214</td><td>185.77025
0.04866</td><td>122.07625
0.66647</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>252.10075
0.09307</td><td>299.888380
.06514</td><td>158.07129
0.05197</td></tr></table></td></tr>
<tr><td rowspan="2">two
random</td><td>Seoul (Exp. 1)</td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>256.154
39.8966</td><td>291.575
0.46018</td><td>158.017
.053434</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>250.456
.123881</td><td>293.374
0.68110</td><td>158.23
0.4350</td></tr></table></td></tr>
<tr><td>Ireland/
Seoul (Exp. 2)</td>
<td><table><tr><td>Seoul</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>274.869
70.0618</td><td>183.878
0.33083</td><td>119.592
0.07868</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>251.9791
0.083702</td><td>300.61975
0.0400598</td><td>157.8935
0.074314</td></tr></table></td></tr>
</table>

We would like to note that the very first RTT of an experiment would occasionally be very high, resulting in our standard deviations for Dublin and Seoul sometimes being high despite very little frequency in any values other than that first one.

<table>
<tr><td>Policy</td><td>Experiment</td><td>RTT(Upload)</td><td>RTT(Download)</td></tr>
<tr><td rowspan="2">local</td><td>Seoul (Exp. 1)</td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>281.228
87.2092</td><td>294.749
0.07776</td><td>158.163
0.05796</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>241.553
0.07751</td><td>295.886
0.07258</td><td>158.069
0.048497</td></tr></table></td></tr>
<tr><td>Ireland/
Seoul (Exp. 2)</td>
<td><table><tr><td>Seoul</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>275.962
73.06785</td><td>184.019
0.85658</td><td>119.447
0.05457</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>252.41113
0.2044158</td><td>295.5104
0.0322753</td><td>157.80888
0.0734311</td></tr></table></td></tr>
<tr><td rowspan="2">Local
closest</td><td>Seoul (Exp. 1)</td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>255.168
13.3508</td><td>294.239
0.04766</td><td>158.396
0.05130</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>249.798
0.07116</td><td>301.264
0.06328</td><td>158.055
0.113142</td></tr></table></td></tr>
<tr><td>Ireland/
Seoul (Exp. 2)</td>
<td><table><tr><td>Seoul</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>273.81
67.0297</td><td>183.862
0.05970</td><td>123.686
0.025411</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>252.232
0.188</td><td>299.921
0.086</td><td>158.053
0.9845</td></tr></table></td></tr>
<tr><td rowspan="2">Local
farthest</td><td>Seoul (Exp. 1)</td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>270.02
60.895</td><td>295.95
0.0665</td><td>158.128
0.10646</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>249.901
0.08779</td><td>295.339
0.05531</td><td>157.836
0.09535</td></tr></table></td></tr>
<tr><td>Ireland/
Seoul (Exp. 2)</td>
<td><table><tr><td>Seoul</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>273.918
71.8509</td><td>183.819
0.038634</td><td>123.881
0.094254</td></tr></table></td>
<td><table><tr><td>Dublin</td><td>Sao Paulo</td><td>Mumbai</td></tr><tr><td>252.137
0.076</td><td>299.804
0.174</td><td>157.789
0.271</td></tr></table></td></tr>
</table>

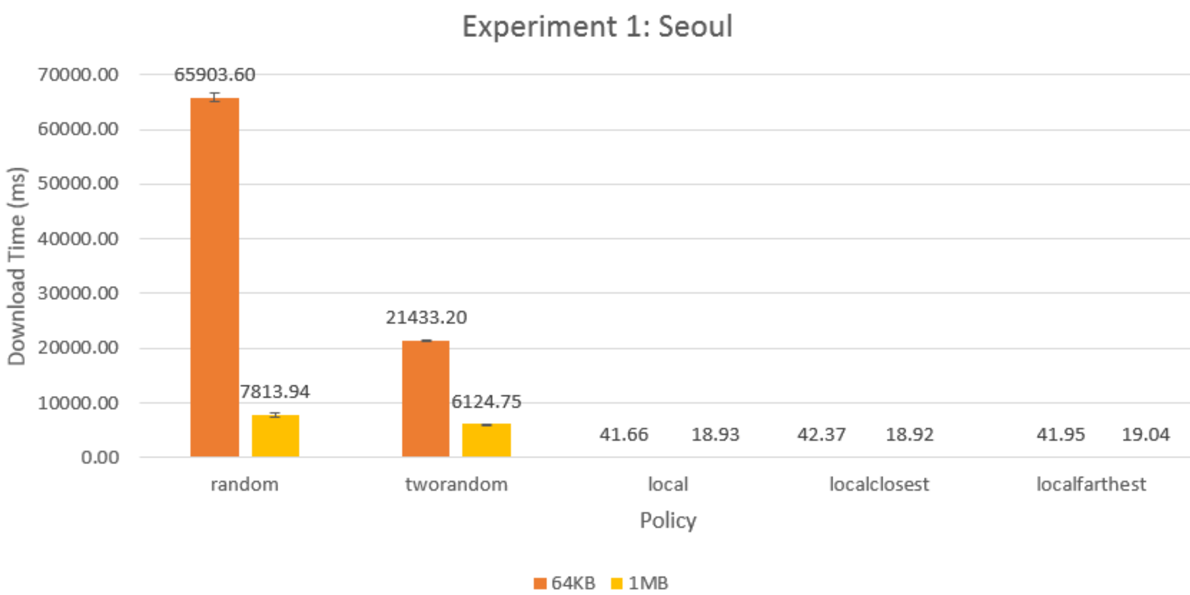

Figure 1. The above graph represents the results of three runs of Experiment 1, uploading and downloading from Seoul. The average download times of 10 1MB files using various policies are represented as bars and the standard deviations are plotted as error bars.

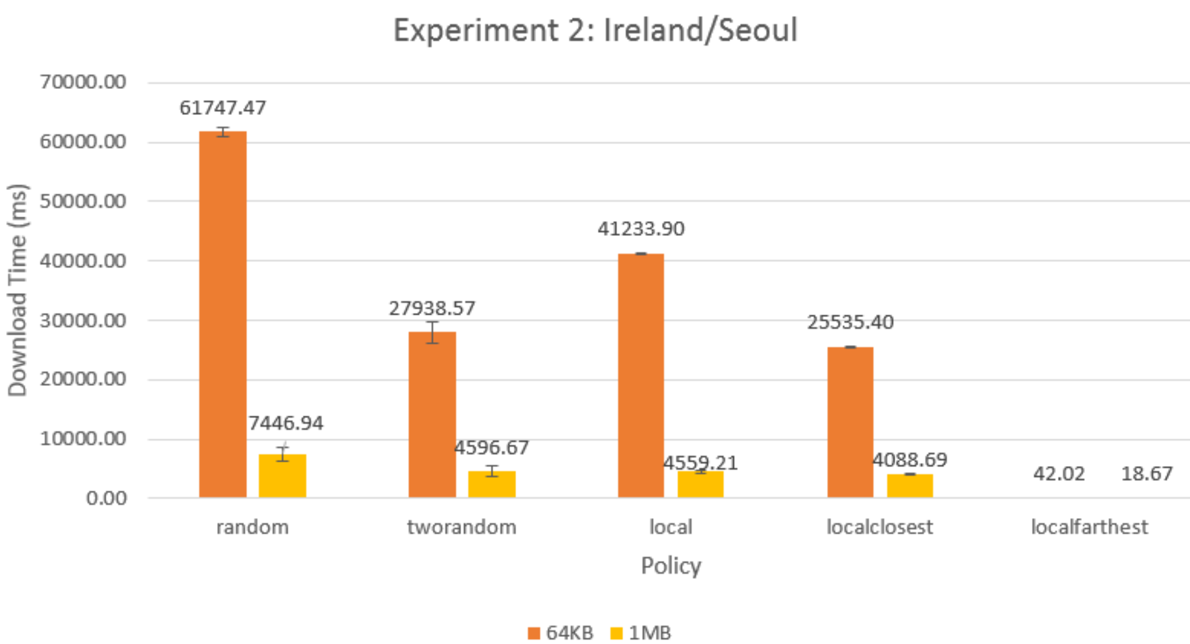

Figure 2. The above graph represents the results of three runs of Experiment 2, uploading from Ireland and downloading from Seoul. Each bar represents the average download time of 10 1MB files using various policies across 4 servers. The standard deviations are plotted as error bars.

For reference, we include the following 2 tables of the standard deviation values for Experiment 1 and Experiment 2 though they are already plotted as error bars on our graphs above. (Experiment 1 values are on the left, Experiment 2 on the right).

STD	64KB	1MB
random	717.175	329.467
tworandom	204.750	8.334
local	0.255	0.338
localclosest	0.512	0.251
localfarthes	0.735	0.041

STD	64KB	1MB
random	704.678	1037.764
tworandom	1886.816	899.924
local	138.492	234.884
localclosest	27.267	140.145
localfarthest	0.127	0.067

Figure 3. The standard deviations specified in table format for both experiments.

Findings and Conclusions

1. Correlation between observed RTT and throughput/bandwidth

There is a very strong assumed correlation between RTT and throughput/bandwidth. It is common sense that they should be inversely proportional. We can see this when we observe how quickly we are able to download data from a data center that is local and subsequently has a smaller RTT compared to when downloading from a datacenter further away (as shown in Figure 2 under policy local).

2. Relative performance of the policies for 1-site experiments.

The results from local, local closest, and local farthest policies all downloaded at about the same, shorter time. The results from random and two-random policies differed from both each other and from the other three policies. The random policy results in the largest download times, as expected, because the scattered blocks mean that a decent proportion of blocks will be located somewhere that takes longer to download from (in our case an example of such a location is Sao Paulo). The two-random policy decreases the time considerably because placing two copies of the block when only four servers exist greatly increases the chance that a block is local or very close. The local policy performs extremely well because a local download is extremely fast. The local closest policy performs the same as the local policy because the extra copy on the “closest server,” which in this case is Mumbai, is still farther than our local copy and so we download locally again. The same logic applies to the local farthest policy, where the extra copy is located in Sao Paulo (even farther from our local than Mumbai).

3. Relative performance of the policies for 2-site experiments.

The results vary greatly across the policies in this 2-site experiment. We can observe the same comparison between the random and two-random policies that we did in the 1-site experiment. This comparison is that the random policy takes the longest due to the placing of blocks in all four locations at random, and only having one copy (which forces us to download some proportion of blocks from the farthest center to us, Sao Paulo since we are downloading from Seoul). The second copy of the blocks in the two-random policy allows the RTT when fetching some blocks to be shorter due to more options when downloading blocks, resulting in a smaller download time. The local policy for our experiment uploads to Ireland, which we can see has one of the higher RTTs from Seoul, which causes our local policy experiment to have high download times. These download times are worse than the two random policy download times, which is expected since in the two random policy we can assume a large proportion (over the half we could assume in random) of the blocks are placed either locally or in Mumbai, which are both closer to Seoul than Ireland is. The local closest policy uploads to both Ireland and Mumbai, where Mumbai happens to be the closest datacenter to Seoul other than itself. This is why our local closest policy download times are faster than our local policy download times, and our two random policy times. Finally, our local farthest policy uploads to Ireland and to Seoul itself, so our downloads become local and run extremely fast.

4. Block size affecting the download time for each policy.

We can logically believe and see from our data that block size is inversely proportional to download time. As we are able to upload/download larger blocks, we have to make fewer round trips to gather all of our data and this is why our 1 MB experiments all have faster download times than their 64 KB counterparts.

5. Overall comments based on results

In this experiment, we learned that the policy we use to upload our data has drastically different results no matter which policy we use, but that the random policies have less variation across different data centers in comparison to local policies that become very “hit or miss”. We also learned that the RTTs between data centers is a very accurate depiction for how uploads/downloads between those data centers will occur.

6. Further experiments that could be performed.

An additional experiment we would like to run would be to run our uploader from Sao Paulo and download from Seoul to witness the local closest policy in comparison to the two random policy. We believe this would be interesting because local closest would upload to Sao Paulo and Ireland, which are both very far from Seoul. In this scenario, it is possible that local closest performs worse than two random.

Another additional experiment to perform could be to flood particular servers with requests and see at what point the client should choose to download the data from a server further away. From this we could draw comparisons between the locality of a server and its availability, finding at which point it makes sense to no longer rely on servers due to their proximity. It would be interesting to see what the effect of multiple downloads from a single server would have on its own ability to download and upload data.