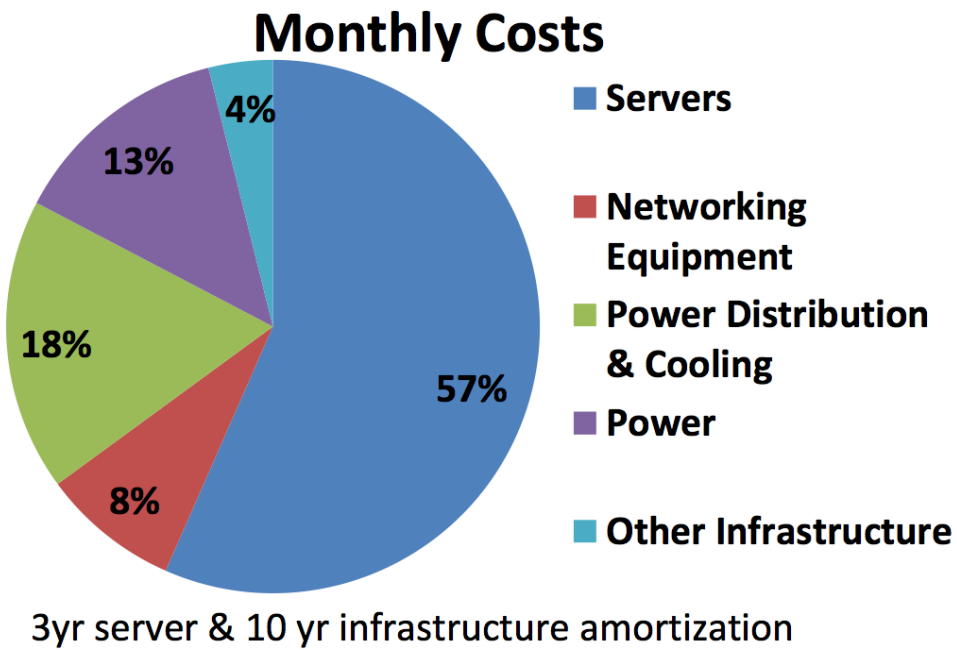
Tutorial 7

1. James Hamilton of Amazon Web Services gave the following price breakdown of a monthly costs for a commercial (non-Amazon) datacenter in his HPTS 2011 talk (`http://mvdirona.com/jrh/talksandpapers/JamesHamilton_HPTS2011.pdf`).



Summing up the power related costs equals $13\% + 18\% = 31\%$ of the datacenter monthly costs.

   a) If we assume a highly energy efficient server design could remove 25% of the monthly power related costs, how much a saving percentage wise would it be to the overall monthly datacenter costs?

   b) If we assume a highly cost optimized server design could remove 25% of the monthly server costs, how much a saving percentage wise would it be to the overall monthly datacenter costs?

   c) If we assume a highly optimized software design would save 25% of both the required monthly server costs as well as the monthly power related costs, how much a saving percentage wise would it be to the overall monthly datacenter costs?

2. One of the difficulties in scaling out distributed systems is that of worst case latency. Imagine for example a Web search application, where the search index is distributed over N servers, and in order to give the final search result to the client one query is sent to each one of the N servers and summarized together in a centralized fashion before giving the client back any results. Assume that the response time of each of the servers is a Bernoulli process with response time of 2 ms for 99% of the queries, and the response time of 100 ms for 1% of the queries (due to e.g, hard disk seek latencies, Java Garbage collection, or other bookkeeping processes running on the same server). Assume for simplicity there are no other delays.

   a) What is the expected latency for a client query in the case $N = 10$?

   b) What is the expected latency for a client query in the case $N = 100$?

   c) What is the expected latency for a client query in the case $N = 1000$?

   Note: Large scale web search systems often do not wait for all servers to reply before returning (initial) search results in order to avoid some of the worst case latency issues.