I/O Workload Characterization

Zachary Kurmas -- Georgia Tech
kurmasz@cc.gatech.edu
Kim Keeton -- HP Labs
kkeeton@hpl.hp.com
Ralph Becker-Szendy -- HP Labs

Motivation

- In traces, the ratio of information to bits is low
  - Traces contain much more data than most people need
- The essence of workload characterization:
  - Determine what information people need
  - Figure out how to represent it
  - Verify that the characterization does, in fact, capture all the important information

Characterization for SSP

- SSP at HP is researching how to optimally configure storage systems for best (cost-effective) performance
  - Would like an analytic model to predict performance for different configurations
  - Would like a concise input for this model
  - "Information" is the data in the trace that affects performance

Introduction

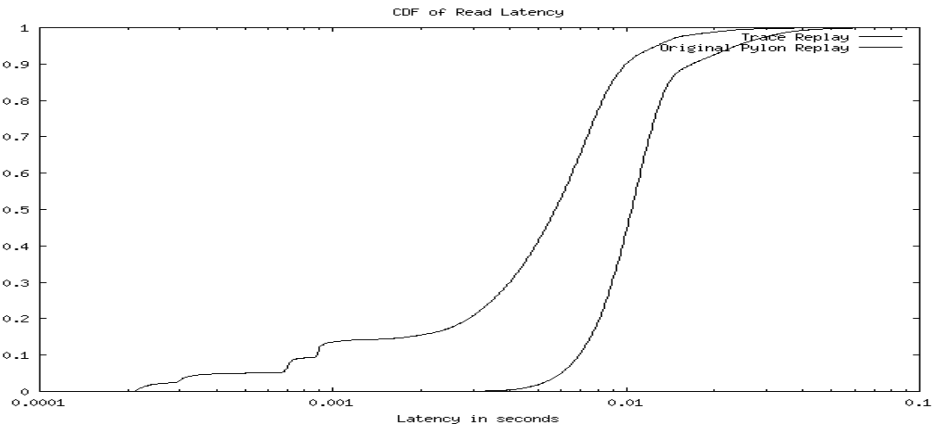
- We are developing an iterative method by which we learn how to characterize workloads
  - We are able to easily
    - Test the quality of the characterization
    - Isolate the effects of individual characteristics
      - to determine which information is missing
    - Add missing information to the characterizations

Roadmap

- Motivation and Introduction
- Description of Method
- Results from first 3 iterations
- Future Work
- Related Work
- Conclusions

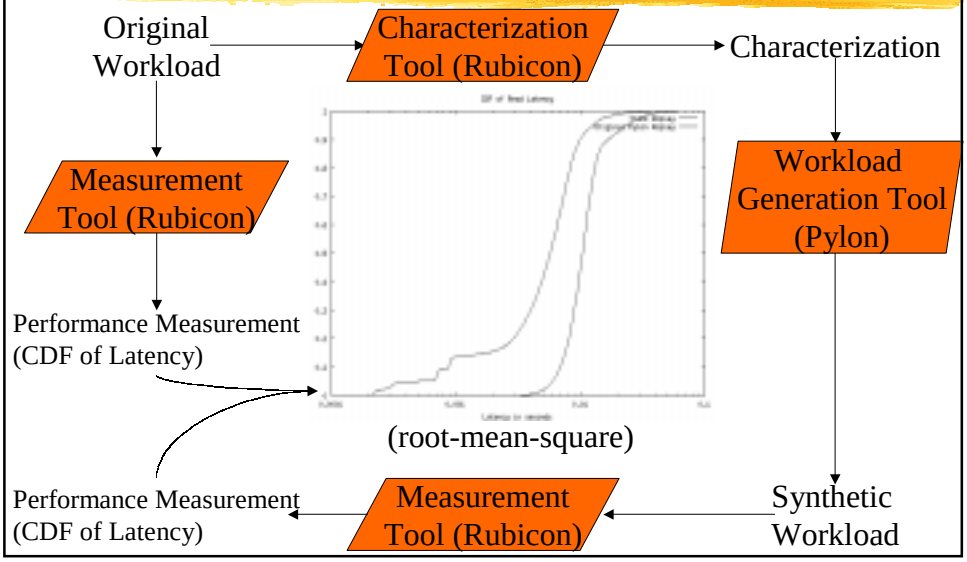
Verification

- How do we know if we have enough information?
  - If any workload with the same characterization has the same performance (latencies)
  - If we can generate another workload with the same performance

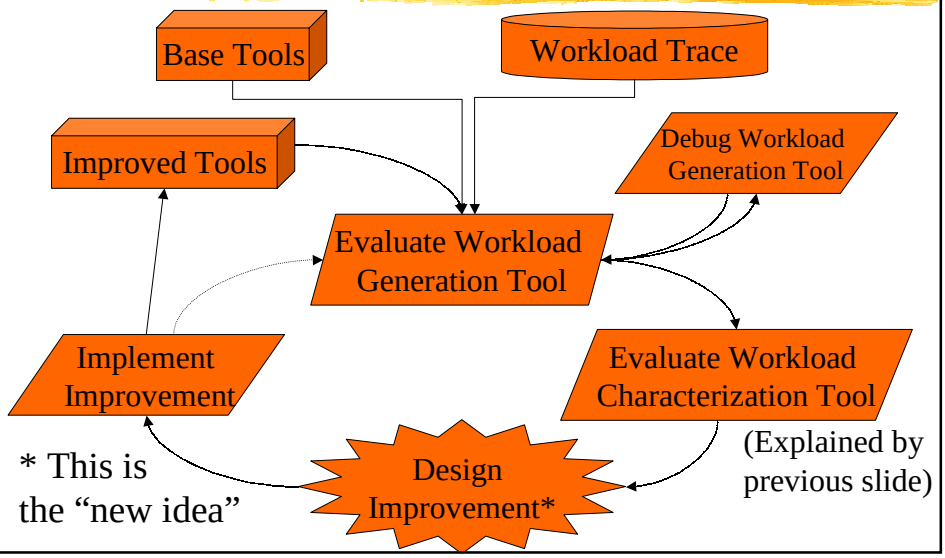


- Performance compared using CDFs of latency
  - (Cumulative Distribution Function)
  - Difference between CDFs quantified using root-mean-squared metric.
    - fancy way of measuring the area between the curves
    - mean of square of horizontal distance between graphs

Validation Method



Iterative Process



Characterizing Performance

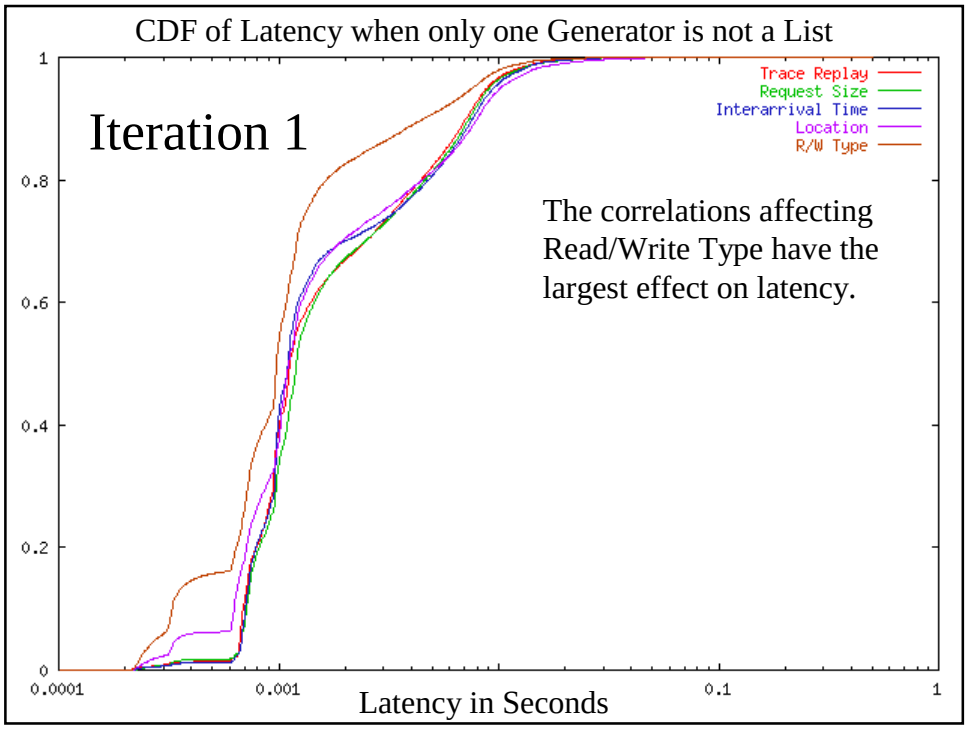
- Each I/O Request has four parameters:
  - Location, Request Size, Type (Read/Write), and Interarrival Time
- A workload is a sequence of requests
- Performance of a workload is determined by
  - Distribution of values for each parameter
  - Correlations within and between parameters' values
- "Useful" characterization must describe all "important" distributions and correlations

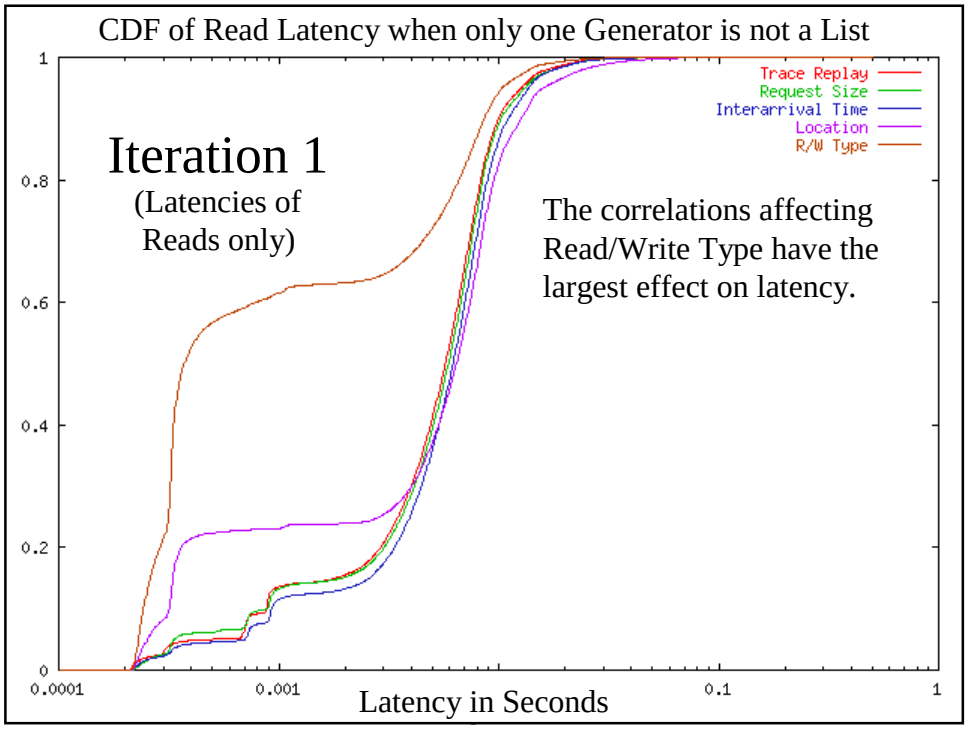
Workload Generator

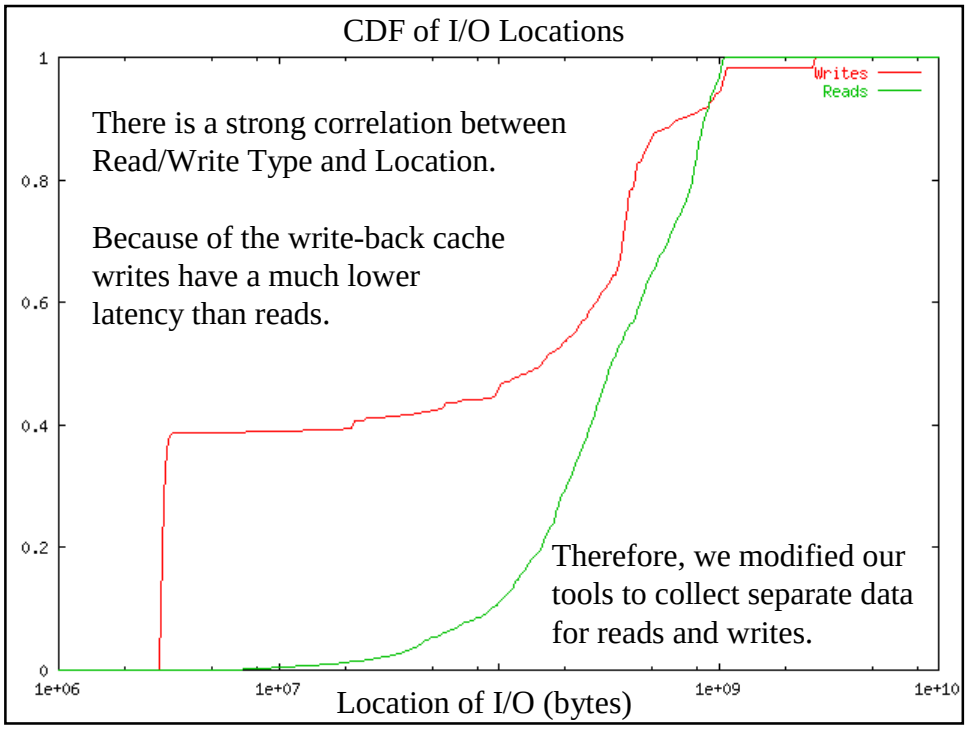
- The workload generator chooses values for each parameter separately.
  - If the generator chooses values by reading them from the trace of the original workload, then the synthetic trace is the same as the original.
  - By reading some parameters from this list (thus, holding them "constant") and choosing other values randomly, we can remove some correlations, and thus, test their importance.

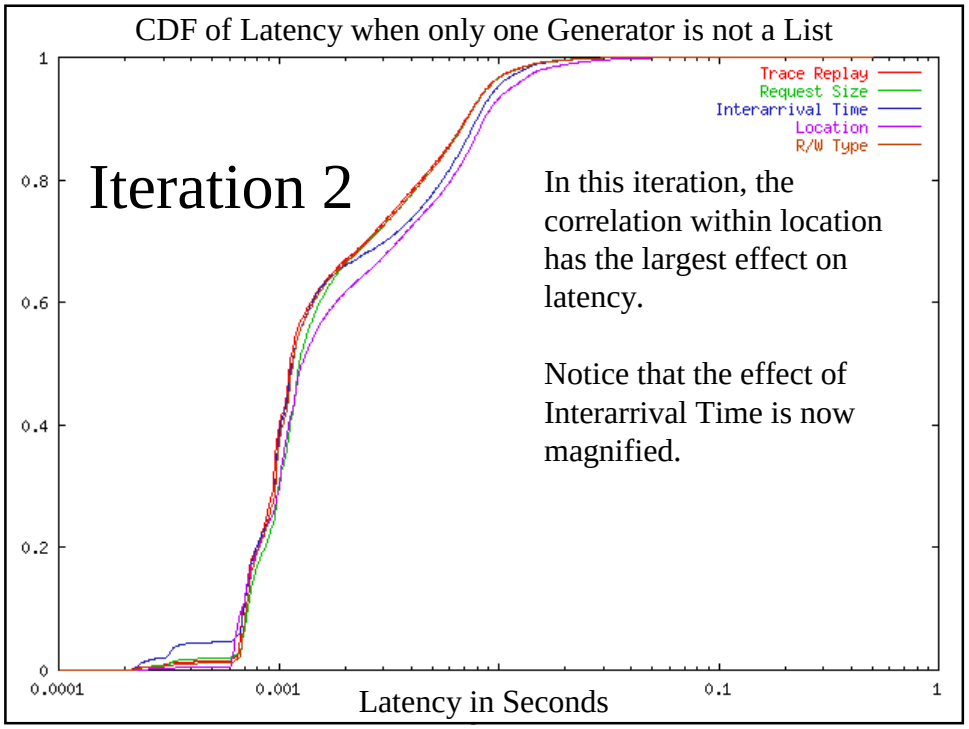
Research Environment

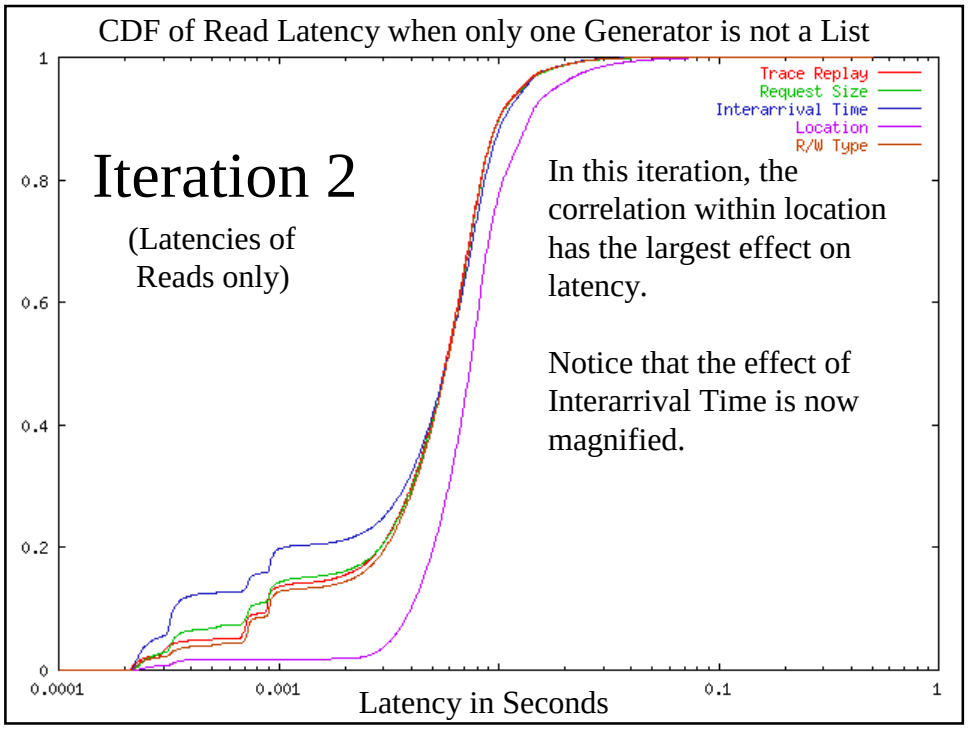
- Workload: Trace of Open Mail
  - e-mail application for 15,000 users
  - Mean request rate: 75.52 I/Os per second
  - Mean request size: 7115 bytes
  - Mean throughput: 524.5KB per second
- Storage System: Optimus disk array
  - Max I/Os per second: about 100
  - Write-back cache
    - Thus, writes are "free"

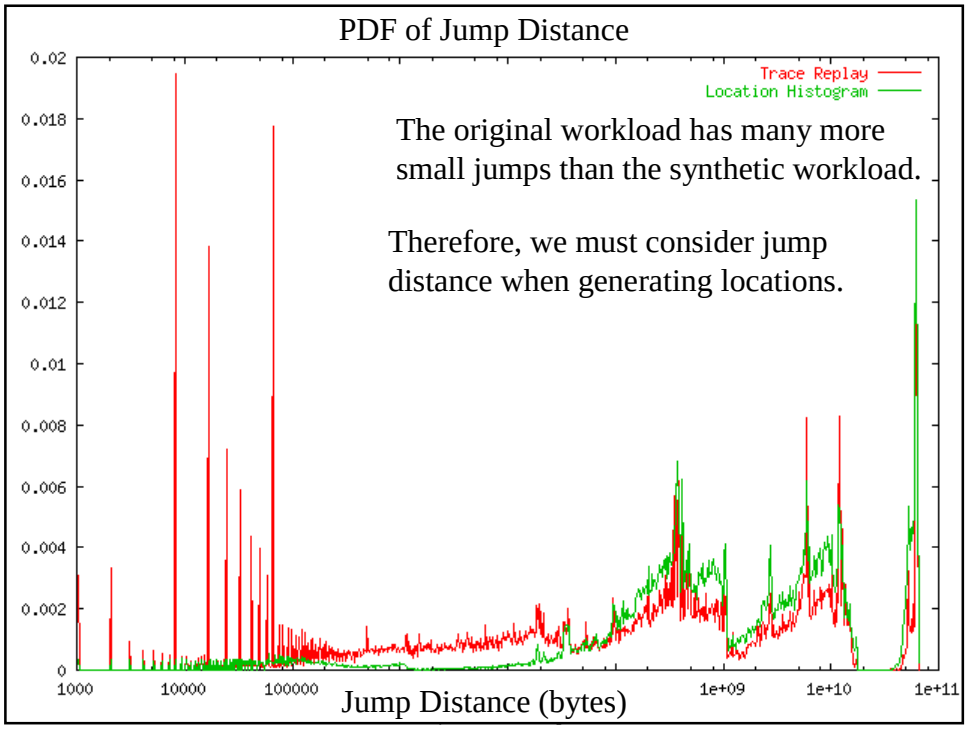

CDF of Latency when only one Generator is not a List
Iteration 1
The correlations affecting Read/Write Type have the largest effect on latency.
Trace Replay
Request Size
Interarrival Time
Location
R/W Type
1
0.8
0.6
0.4
0.2
0
0.0001
0.001
0.1
1
Latency in Seconds


CDF of Read Latency when only one Generator is not a List
Iteration 1
(Latencies of Reads only)
The correlations affecting Read/Write Type have the largest effect on latency.
Trace Replay
Request Size
Interarrival Time
Location
R/W Type
1
0.8
0.6
0.4
0.2
0
0.0001
0.001
0.1
1
Latency in Seconds


CDF of I/O Locations
Writes
Reads
There is a strong correlation between Read/Write Type and Location.
Because of the write-back cache writes have a much lower latency than reads.
Therefore, we modified our tools to collect separate data for reads and writes.
1
0.8
0.6
0.4
0.2
0
1e+06
1e+07
1e+09
1e+10
Location of I/O (bytes)


CDF of Latency when only one Generator is not a List
Trace Replay
Request Size
Interarrival Time
Location
R/W Type
Iteration 2
In this iteration, the correlation within location has the largest effect on latency.
Notice that the effect of Interarrival Time is now magnified.
1
0.8
0.6
0.4
0.2
0
0.0001
0.001
0.1
1
Latency in Seconds


CDF of Read Latency when only one Generator is not a List
Iteration 2
(Latencies of Reads only)
In this iteration, the correlation within location has the largest effect on latency.
Notice that the effect of Interarrival Time is now magnified.
Trace Replay
Request Size
Interarrival Time
Location
R/W Type
1
0.8
0.6
0.4
0.2
0
0.0001
0.001
0.1
1
Latency in Seconds


PDF of Jump Distance
The original workload has many more small jumps than the synthetic workload.
Therefore, we must consider jump distance when generating locations.
Trace Replay
Location Histogram
0.02
0.018
0.016
0.014
0.012
0.01
0.008
0.006
0.004
0.002
0
1000
10000
100000
1e+09
1e+10
1e+11
Jump Distance (bytes)

Jump Distance

- Two simple and naive attempts failed:
  - Choosing location based on a distribution of jump distance rather than location; and
  - Choosing a specified percentage of locations from the jump distance distribution and the rest from the location distribution.
- Because many threads are writing to each disk, we suspect that a per-process jump distance does not accurately account for the observed spatial locality.

Future Work

- Develop a better method of generating locations
- We suspect that Interarrival Time/ burstiness will be the next big issue.
  - Much other research in this area
- Test our method on many different workloads

Related Work

- Many people have studied one or two parameters:
  - Ganger -- Location and Interarrival Time
  - Faloutsos -- Interarrival Time / Burstiness
  - Gomez and Santonja -- Location
- We will consider how to incorporate these results into our framework.

Conclusions

- We presented a new methodology for characterizing a workload.
- Using this methodology we can easily
  - Verify that the characterization has captured all the "important" information
  - Isolate the effects of individual parameters
    - and decide where to make improvements
  - Improve the characterization