Operational Experiences with Disk Imaging in a Multi-Tenant Datacenter

Kevin Atkinson, Gary Wong, and Robert Ricci
Properties of **disk images** and their usage have consequences for:

- **Storage**
- **Caching**
- **Pre-loading**
- **Distribution**
What does the working set look like?
What does the **working set** look like?

What do the **images themselves** look like?
What does the working set look like?

What do the images themselves look like?

What are the key factors in pre-loading?
The dataset

- Four years (2009-2013): 279,972 requests
- Users: 1,301 individuals, 368 organizations
- Unique images: 714
- Emulab
  - ~600 PCs
- Facility / user image model
User Behavior
“Emulab is a pretty odd beast and its users are even weirder.”
“Emulab is a pretty odd beast and its users are even weirder.”

-Reviewer D
“Emulab is a pretty odd beast and its users are even weirder.”

Reviewer D
[Emulab user]
Facility vs. user images

<table>
<thead>
<tr>
<th>Facility</th>
<th>User</th>
</tr>
</thead>
<tbody>
<tr>
<td>55.6%</td>
<td>44.4%</td>
</tr>
</tbody>
</table>
Facility vs. user images

<table>
<thead>
<tr>
<th>Facility</th>
<th>User</th>
</tr>
</thead>
<tbody>
<tr>
<td>55.6%</td>
<td>44.4%</td>
</tr>
</tbody>
</table>
Facility vs. user images

<table>
<thead>
<tr>
<th>Facility</th>
<th>User</th>
</tr>
</thead>
<tbody>
<tr>
<td>55.6%</td>
<td>44.4%</td>
</tr>
</tbody>
</table>

1) Most users stick to facility or user images
2) Heaviest users use their own images
Image popularity
Image popularity

![Graph showing the distribution of image popularity with a steep decline indicating that a small number of images receive a large number of requests.]
Image popularity

![Graph showing image popularity with two lines representing Facility and User requests over rank. The graph has a logarithmic scale for requests on the y-axis and rank on the x-axis. The lines show a decreasing trend as rank increases.]
Image popularity

- Facility
- Exp(λ=0.143)
- User
- Exp(λ=0.026)
Image popularity

Exponential
Image popularity

```
Requests

<table>
<thead>
<tr>
<th>Facility</th>
<th>Exp((\lambda=0.143))</th>
<th>User</th>
<th>Exp((\lambda=0.026))</th>
</tr>
</thead>
</table>
```

Exponential

Heavy-Tailed

Rank

0 100 200 300 400 500 600
Image popularity

1) Facility images have a smaller, lighter tail
2) Most popular image < 13% of requests
Scaling: total images

Graph showing the number of images used against the percentage of the user base.
Scaling: total images

The graph shows the number of images used by users (red line) and facilities (blue line) as a percentage of the user base increases. The number of images used by users grows steadily, while the number of images used by facilities increases more gradually.
As userbase grows, user images dominate the totals.
Daily working set

Median = 12.00
Mean = 11.98
Std. dev. = 4.33
Daily working set

Median = 12.00
Mean = 11.98
Std. dev. = 4.33

Small image set each day → good caching potential
Scaling: working set

Mean daily working set size

Percentage of user base

User Facility
Scaling: working set

[Graph]

Percentage of user base

Mean daily working set size

User Facility
Scaling: working set

![Graph showing the mean daily working set size as a function of the percentage of the user base. The graph includes two lines: red for User and blue for Facility.](image)
Scaling: working set

Facility will max out
Scaling: working set

Facility will max out

→ In the limit, highly popular facility images account for most requests
Image Contents
Block-level similarity

Base
Block-level similarity

Base

Derived
Block-level similarity

Base

Derived
Block-level similarity

Base

Derived

Percentage of blocks that need to be written to transform the base image into derived
Block-level similarity

Derived: User image
Base: Most similar facility image
Block-level similarity

Derived: User image
Base: Most similar facility image
Block-level similarity

1) De-duplicating storage an attractive option
2) Differential loading has potential
Pre-Loading
Pre-loading: Size

![Graph showing the probability of satisfying a request vs. the ratio of free pool size to the number of images. The graph indicates that the probability increases rapidly as the ratio increases from 0 to 0.5, after which the increase becomes more gradual.](image-url)
Pre-loading: Size

Spare Capacity

Probability of satisfying request

Ratio of free pool size to number of images
Pre-loading: Size

Probability of satisfying request

Ratio of free pool size to number of images

Spare Capacity

Mostly Full
Pre-loading: Size

Spare Capacity

WSS for facility images maxes out on large facilities

Mostly Full

Ratio of free pool size to number of images
Pre-loading: Size

1) Key: Ratio of WSS to idle capacity
2) Effective when ratio is high

WSS for facility images maxes out on large facilities

Ratio of free pool size to number of images

Spare Capacity

Mostly Full

Probability of satisfying request
Pre-loading: Rate

![Graph showing the probability of satisfying requests vs. reload rate, normalized to mean arrival rate. The graph exhibits an inflection point around reload rate 1.0, indicating a sharp increase in the probability of satisfying requests.]
Pre-loading: Rate

![Graph showing the probability of satisfying requests against normalized reload rate. The graph highlights a sharp increase at a reload rate close to 1.0.](chart.png)
Pre-loading: Rate

Invest in fast, scalable imaging
Conclusions
General conclusions

❖ Deduplicating, two-tier storage attractive
  ❖ Caching can be effective
    ❖ Image lifespan, idle periods
❖ Treat facility and user images differently
  ❖ Facility better targets for pre-loading
❖ Differential loading requires new strategies
  ❖ Potential savings, outline of optimization problem
❖ Images per organization, WSS per week
Explore the data, reproduce our results:

http://aptlab.net/p/tbres/nsdi14
No dominant images
No dominant images

No image dominates long-term, popular images change frequently
Image lifespan

![Histogram showing the frequency of image lifespans in days. The x-axis represents image lifetime in days, ranging from 0 to 1400, and the y-axis represents frequency. The bars indicate the distribution of how often images last for different lifespans.]
Image lifespan

A few days

Frequency

Image lifetime (days)
Image lifespan

A few days

Four Years
Image lifespan

A few days

Four Years

Two-tiered storage system attractive
Savings from deltas

![Graph showing savings from deltas](image)
Images per organization
Idle images
WSS per week

Median = 30.00
Mean = 30.17
Std. dev. = 6.43
### Top images

<table>
<thead>
<tr>
<th>Image</th>
<th>Downloads</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHL90-STD [D]</td>
<td>21,993</td>
<td>7.9%</td>
</tr>
<tr>
<td>FEDORA10-STD</td>
<td>18,042</td>
<td>6.4%</td>
</tr>
<tr>
<td>UBUNTU10-STD</td>
<td>14,402</td>
<td>5.1%</td>
</tr>
<tr>
<td>RHL90-STD</td>
<td>13,182</td>
<td>4.7%</td>
</tr>
<tr>
<td>FC4-UPDATE</td>
<td>12,097</td>
<td>4.3%</td>
</tr>
<tr>
<td>u 715/10</td>
<td>11,156</td>
<td>4.0%</td>
</tr>
<tr>
<td>FBSD410-STD</td>
<td>8,916</td>
<td>3.2%</td>
</tr>
<tr>
<td>FEDORA8-STD</td>
<td>8,153</td>
<td>2.9%</td>
</tr>
<tr>
<td>u 237/69</td>
<td>7,512</td>
<td>2.7%</td>
</tr>
<tr>
<td>u 296/35</td>
<td>7,179</td>
<td>2.6%</td>
</tr>
<tr>
<td>u 787/24</td>
<td>6,243</td>
<td>2.2%</td>
</tr>
<tr>
<td>UBUNTU70-STD</td>
<td>6,021</td>
<td>2.2%</td>
</tr>
<tr>
<td>UBUNTU12-64-STD</td>
<td>5,834</td>
<td>2.1%</td>
</tr>
</tbody>
</table>
Size considerations

❖ Small facilities with few idle disks
❖ Pre-loading not valuable
❖ Large facilities - focus on:
  ❖ Scalable reloading mechanisms
  ❖ Prediction and optimization for user requests