

Programming the Performance Co-Pilot toolkit

Nathan Scott
Performance Tools, Red Hat
August 2014

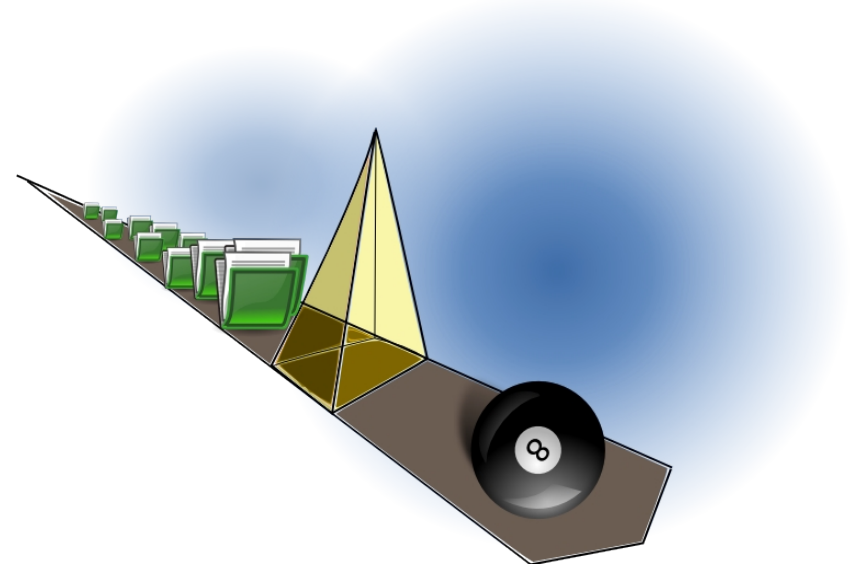
Outline

- Performance Co-Pilot (PCP)
 - Overview
 - Exploring PCP
- Extending with Python
 - Collectors
 - Monitors

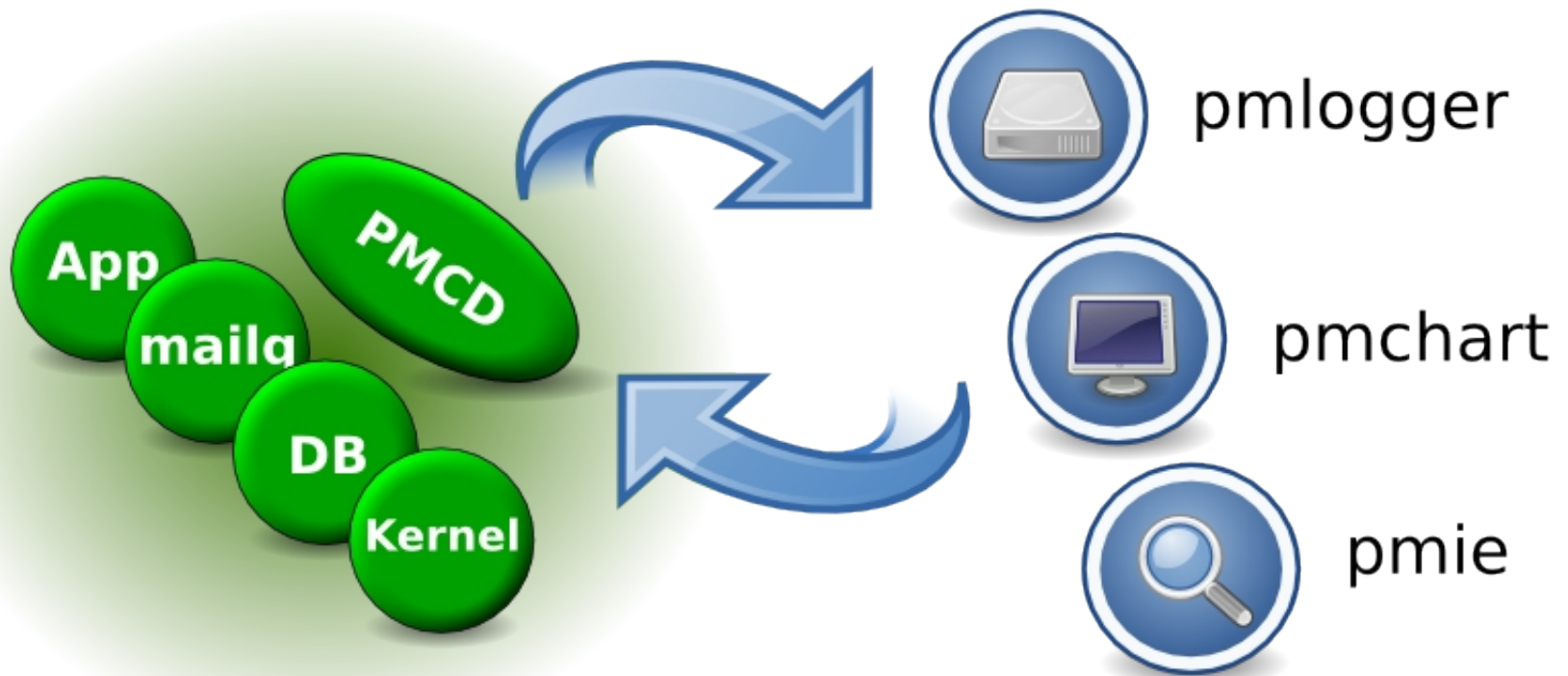


Overview

- What is PCP?
 - Open source **toolkit**
 - System-level analysis
 - Live and historical
 - Extensible (monitors, collectors)
 - Distributed



Architecture



Metrics

- **pminfo --desc -tT --fetch *disk.dev.read***

disk.dev.read [*per-disk read operations*]

Data Type: *32-bit unsigned int* InDom: *60.1*

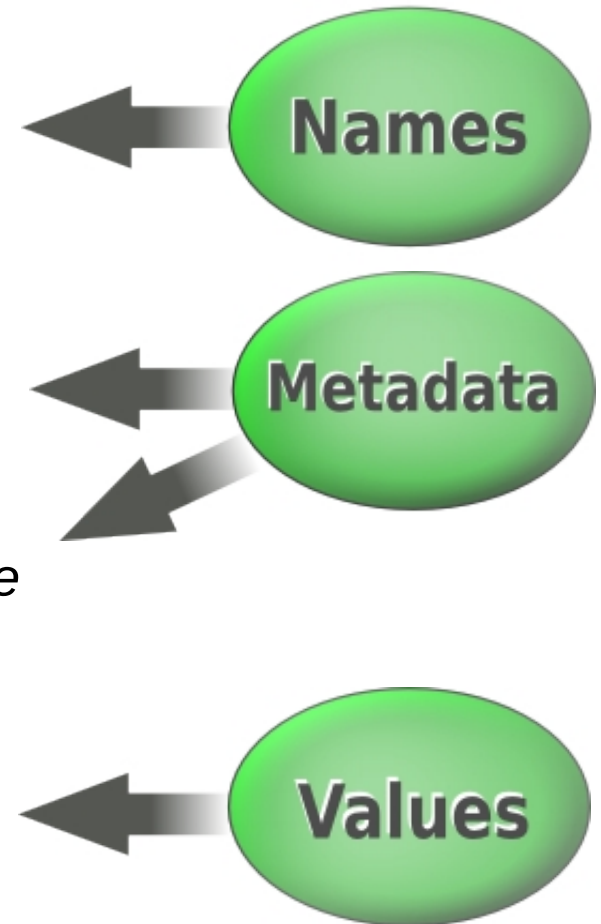
Semantics: *counter* Units: *count*

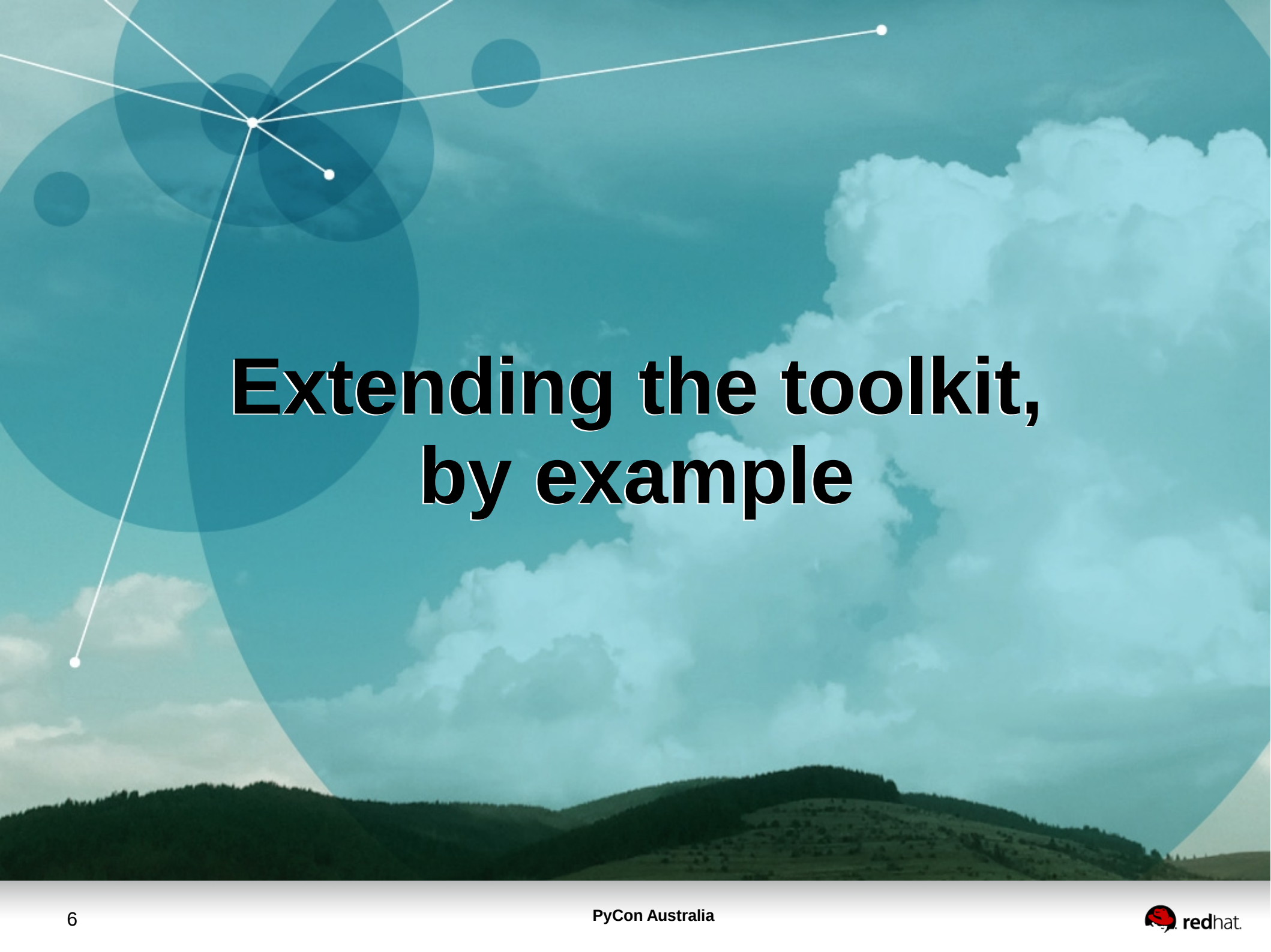
Help: *Cumulative count of disk reads since boot time*

Values:

inst [0 or "**sda**"] value **3382299**

inst [1 or "**sdb**"] value **178421**



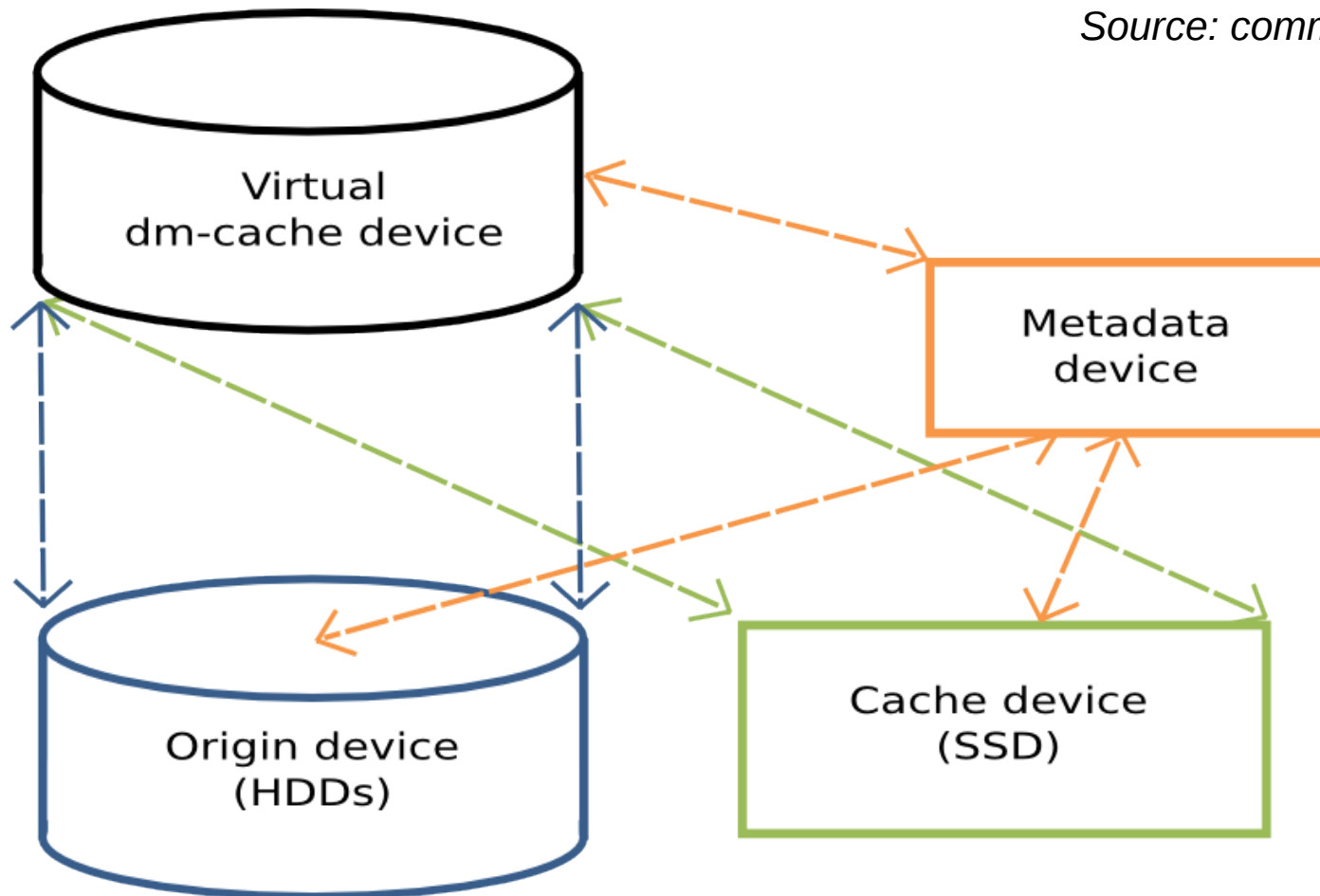
The background of the slide features a scenic landscape of rolling green hills under a vast, blue sky filled with white, fluffy clouds. Overlaid on the left side of the image is a network diagram consisting of several semi-transparent blue circles of varying sizes. These circles are interconnected by thin white lines, with some lines extending towards the edges of the frame. The overall aesthetic is clean and modern, with a focus on technology and nature.

Extending the toolkit, by example

Device Mapper Cache target

“Linux Block Caching Choices [...]” (pdf)

Source: community.dell.com



Collector extensions in Python

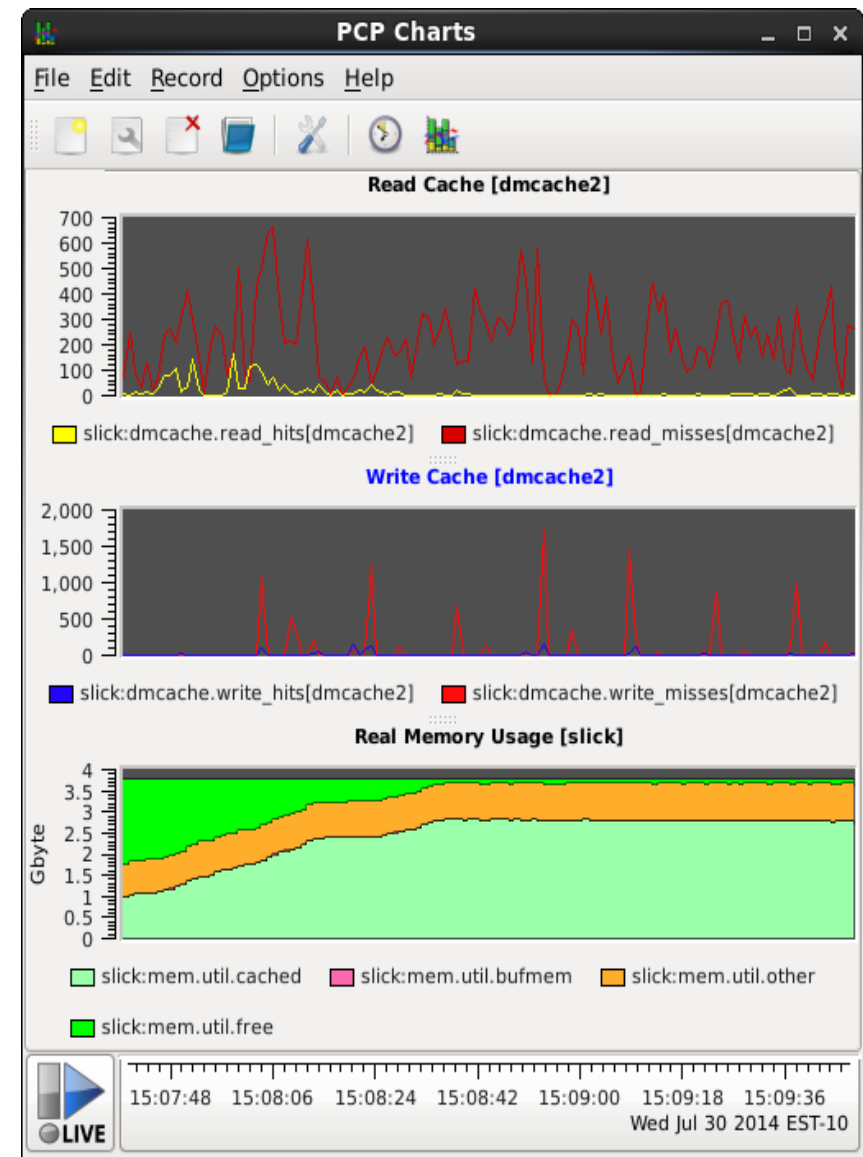
- Goal: new dmcache metrics
 - For recording, charting, console tools

```
nathans@smash:~/git/pcp
nathans@smash:~/git/pcp$ pmval -s3 --host=slick dmcache.dirty

metric:    dmcache.dirty
host:      slick
semantics:  instantaneous value
units:     Kbyte
samples:   3
interval:  1.00 sec

           dmcache2           dmcache1
           512                0
           0                  768
           1792               256

nathans@smash:~/git/pcp$
```



Collector extensions in Python

```
nathans@slick:~/git/pcp$ sudo dmsetup status --target=cache
dmcach2: 0 52606976 cache 8 410/837632 512 71/117856 786 36 344 0 0 0 0 1 write
back 2 migration_threshold 2048 mq 10 random_threshold 4 sequential_threshold 51
2 discard_promote_adjustment 1 read_promote_adjustment 4 write_promote_adjustmen
t 8
dmcach1: 0 52428800 cache 8 410/837632 512 671/117856 7574 11290 33550 507552 0
600 0 1 writeback 2 migration_threshold 2048 mq 10 random_threshold 4 sequentia
l_threshold 512 discard_promote_adjustment 1 read_promote_adjustment 4 write_pro
mote_adjustment 8
nathans@slick:~/git/pcp$
```

- Domains, Metrics, Instances
- *import pcp.pmda*
- Create a PMDA subclass
- *add_metric* (...names, IDs, units, etc...)
- Fetch callback – turns metric IDs into values

Monitor extensions in Python

- Goal: a simple, custom dmcache stats tool...


```
nathans@smash:~/git/pcp
nathans@smash:~/git/pcp$ pcp --host slick dmcache
@ Wed Jul 30 15:23:06 2014 (host slick)
---device---  ---%used---  -----reads-----  -----writes-----
             meta  cache      hit      miss      ops      hit      miss      ops
dmcachel1    0.0%  0.0%        ?        ?        ?        ?        ?        ?
dmcachel2    0.0%  0.0%        ?        ?        ?        ?        ?        ?
dmcachel1    0.0%  0.0%      0.00      0.00      0.00      0.00      0.00      0.00
dmcachel2    0.0%  0.0%    151.66    100.00    234.16      9.17     40.83     49.16
dmcachel1    0.0%  0.0%      0.00      0.00      0.00      5.38     1.41K     1.42K
dmcachel2    0.0%  0.0%    137.29    183.06    339.20      2.69     10.77     14.36
dmcachel1    0.0%  0.0%      0.00      0.00      0.00      0.00      0.00      0.00
dmcachel2    0.0%  0.0%     61.22     30.61     90.84      0.00      0.00      0.00
dmcachel1    0.0%  0.0%      0.00      0.00      0.00      0.00      0.00      0.00
dmcachel2    0.0%  0.0%     60.75     75.94    133.31      0.00      0.00      0.00
dmcachel1    0.0%  0.0%      0.00      0.00      0.00      0.00      0.00      0.00
dmcachel2    0.0%  0.0%    114.66     57.33    175.94      0.00      0.00      0.00
dmcachel1    0.0%  0.0%      0.00      0.00      0.00      6.91      0.00      0.00
dmcachel2    0.0%  0.0%     12.83     66.12     79.94      0.99    975.02    975.02
dmcachel1    0.0%  0.0%      0.00      0.00      0.00      0.00    346.24    346.24
dmcachel2    0.0%  0.0%      0.98      2.95      3.93      0.00      0.00      0.00
nathans@smash:~/git/pcp$
```

Monitor extensions in Python

- Report metrics (dmcache.* and disk.dm.*)
- *import pcp.pmcc*
- Create a *MetricGroupManager* object
- Feed it metric names + a *MetricGroupPrinter* subclass
- *MetricGroupManager* does all the heavy lifting:
 - Sampling (fetch interval, count)
 - Source (archives, local/remote hosts)
 - Time window, timezones, options, etc.

Resources

- **git clone git.pcp.io/pcp dev**
 - *src/pmdas/dmcache* (collecting code)
 - *src/pcp/dmcache* (monitoring code)
- “PCP Programmer's Guide” book
- **dbpmda(1)**
- <http://www.pcp.io>
 - Source, downloads, books, FAQ, mailing lists
- <http://developerblog.redhat.com>
 - PCP articles, demo videos



Demo & Questions