redhat

CGroups
with an EAP example

Twin Cities Users Group
Q4/2012

What are Control Groups?

- CGroups are a way to allocate resources to processes running on a system
- CGroups are hierarchical and can be dynamically added, changed and removed
- CGroups are made up of several subsystems also called **Resource Controllers**

Resource Controllers

- **blkio** — this subsystem sets limits on input/output access to and from block devices such as physical drives (disk, solid state, USB, etc.).
- **cpu** — this subsystem uses the scheduler to provide cgroup tasks access to the CPU.
- **cpuacct** — this subsystem generates automatic reports on CPU resources used by tasks in a cgroup.
- **cpuset** — this subsystem assigns individual CPUs (on a multicore system) and memory nodes to tasks in a cgroup.
- **devices** — this subsystem allows or denies access to devices by tasks in a cgroup.
- **freezer** — this subsystem suspends or resumes tasks in a cgroup.
- **memory** — this subsystem sets limits on memory use by tasks in a cgroup, and generates automatic reports on memory resources used by those tasks.
- **net_cls** — this subsystem tags network packets with a class identifier (classid) that allows the Linux traffic controller (tc) to identify packets originating from a particular cgroup task.
- **net_prio** — this subsystem provides a way to dynamically set the priority of network traffic per network interface.
- **ns** — the *namespace* subsystem.

How do you install CGroups?

- Only available in RHEL 6
- Part of RHEL 6 Kernel
- Upstream since 2.6.24
- You must install userspace tools
  - Install libcgroup
    - yum -y install libcgroup

Starting Services

- **CGCONFIG** – is the service that mounts the Cgroup file system as defined in /etc/cgconfig.conf
  - service cgconfig start
  - chkconfig cgconfig on
- **CGRED** – is the service that starts the cgrulesengd daemon – which moves tasks into cgroups according to paramaters set int /etc/cgrules.conf
  - service cgred start
  - chkconfig cgred on
- Disable subsystems at boot time: kernel parameter
  - kernel cgroup_disable=blkio,ns

What is enabled?

[root@rh6-huge etc]# cat /proc/cgroups

#subsys_name	hierarchy	num_cgroups	enabled
cpuset	65	2	1
ns	0	1	1
cpu	66	2	1
cpuacct	67	1	1
memory	68	2	1
devices	69	1	1
freezer	70	1	1
net_cls	71	1	1
blkio	72	1	1
perf_event	0	1	1
net_prio	0	1	1

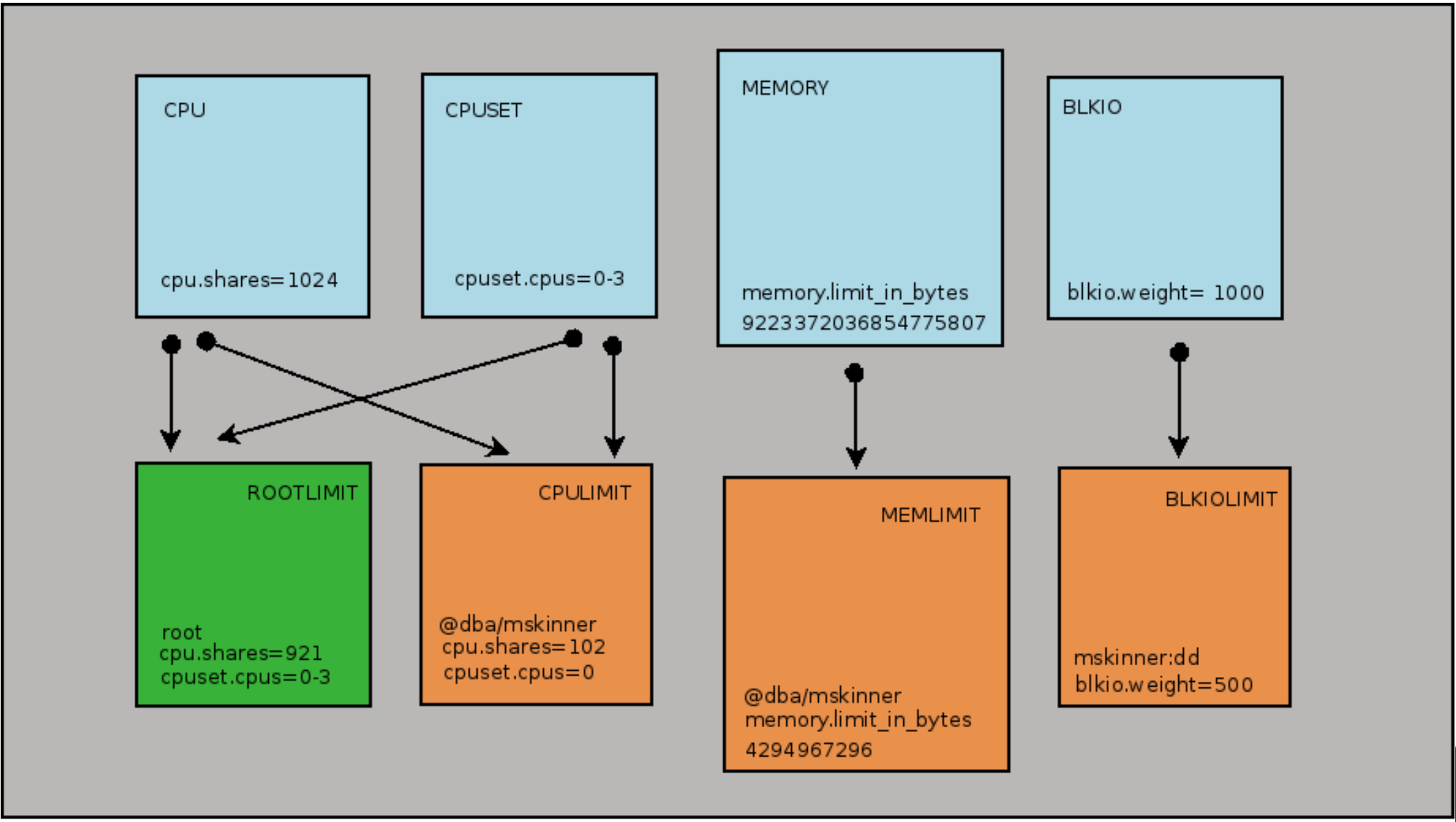
List the Controller Resources

- **LSSUBSYS** – list the subsystems
  - [root@rh6-huge etc]# lssubsys -am
    - ns
    - perf_event
    - net_prio
    - cpuset /cgroup/cpuset
    - cpu /cgroup/cpu
    - cpuacct /cgroup/cpuacct
    - memory /cgroup/memory
    - devices /cgroup/devices
    - freezer /cgroup/freezer
    - net_cls /cgroup/net_cls
    - blkio /cgroup/blkio

Simple Memory Example: create limits and rules

- Create rule to limit user mskinner or group dba to max memory usage of 4G
- In /etc/cgconfig.conf:
  - group memlimit {
  - memory {
  - memory.limit_in_bytes = 4G;
  - }
  - }
- This creates a cgroup that has a max memory limit of 4G
- In /etc/cgrules.conf:
  - mskinner memory memlimit/
  - or
  - @dba memory memlimit/
- Restart cgconfig and cgred

CGroup Examples : Memory



Simple Memory Example: run tests

[mskinner@rh6-huge test]$ ./memtest-4gb

allocated 4095 MB

allocated 4095 MB

DONE - fully allocated 4096 MB

[mskinner@rh6-huge test]$ ./memtest-16gb

allocated 4095 MB

allocated 4095 MB

Killed

/var/log/messages

Nov 28 10:53:33 rh6-huge kernel: Memory cgroup out of memory: Kill process 2592 (memtest-16gb) score 1000 or sacrifice child

Simple CPU Example: how many cores

- What type of CPU do I have – numa layout?
  - [root@rh6-huge /]# lscpu
  - Architecture: x86_64
  - CPU op-mode(s): 32-bit, 64-bit
  - Byte Order: Little Endian
  - CPU(s): 4
  - On-line CPU(s) list: 0-3
  - ... [more]

Simple CPU Example: create user limits

- Create rule to allow user mskinner minimum 10% of CPU
- /etc/cgconfig.conf:
  - group cpulimit {
  - 	cpuset {
  - 		cpuset.cpus = 0;
  - 	}
  - 	cpu {
  - 		cpu.shares = 102;
  - 	}
  - }

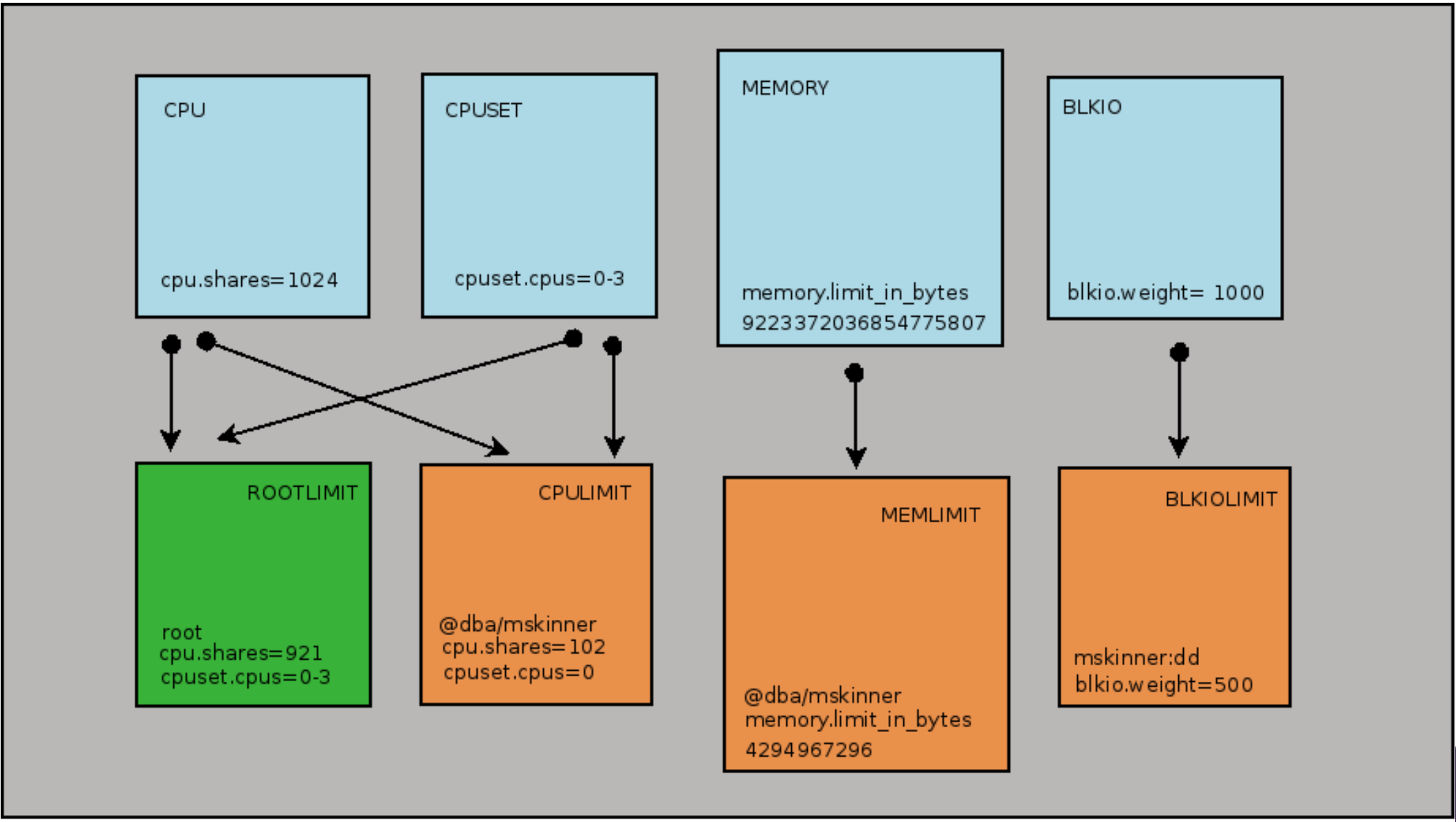
Simple CPU Example: create user limits

- Create rule to allow user mskinner minimum 10% of CPU
- /etc/cgconfig.conf:
  - group cpulimit {
  - 	cpuset {
  - 		cpuset.cpus = 0;
  - 	}
  - 	cpu {
  - 		cpu.shares = 102;
  - 	}
  - }

Simple CPU Example: create root limits

- Create rule to allow user root 90% CPU
  - /etc/cgconfig.conf:
  - group rootlimit {
  - cpuset {
  - cpuset.cpus = 0-3;
  - }
  - cpu {
  - cpu.shares = 921;
  - }
  - }

CGroup Examples : Memory



Simple CPU Example: edit rules

- /etc/cgrules.conf:
  - remove @dba memory memlimit/ *cgred will match first rule only!!!!
  - @dba cpu,cpuset cpulimit/
  - root cpu,cpuset rootlimit/

Simple CPU Example: restart services

- Restart cgconfig and cgred
- **NOTE : If your in the /cgroups mount your restart will FAIL!**
- [root@rh6-huge etc]# service cgconfig restart
  - Stopping cgconfig service: cgclear failed with Device or resource busy [OK]
  - Starting cgconfig service: Loading configuration file /etc/cgconfig.conf failed
  - Cgroup mounting failed
  - Failed to parse /etc/cgconfig.conf [FAILED]

Simple CPU Example: show groups

- Show all Cgroups configured
  - [root@rh6-huge etc]# lscgroup
    - cpuset:/
    - cpuset:/cpulimit
    - cpu:/
    - cpu:/cpulimit
    - cpuacct:/
    - memory:/
    - memory:/memlimit
    - devices:/
    - freezer:/
    - net_cls:/
    - blkio:/

Simple CPU Example: check our resources

- Check our CPU Shares - %100 = 1024 shares
  - mskinner = %10
    - [root@rh6-huge etc]# cgget -r cpu.shares cpulimit
    - cpulimit:
    - cpu.shares: 102
  - Root = %90
    - [root@rh6-huge etc]# cgget -r cpu.shares rootlimit
    - rootlimit:
    - cpu.shares: 921

Simple CPU Example: run the test

- Quick test, full Linux kernel compile with 8 threads:
  - [mskinner@rh6-huge linux-3.6.8]$ make -j8 vmlinux
  - [root@rh6-huge linux-3.6.8]$ make -j8 vmlinux

Top – mskinner vs root

8381	root	20	0	192m	83m	5964	R	46.5	0.3	0:01.51	cc1
8408	root	20	0	170m	61m	5948	R	46.2	0.3	0:01.49	cc1
8524	root	20	0	153m	44m	5588	R	15.9	0.2	0:00.48	cc1
8557	root	20	0	137m	25m	2812	R	6.6	0.1	0:00.20	cc1
8560	root	20	0	136m	25m	2828	R	6.0	0.1	0:00.18	cc1
7402	**mskinner**	**20**	**0**	**177m**	**69m**	**5788**	**R**	**4.6**	**0.3**	**0:00.84**	**cc1**
7879	**mskinner**	**20**	**0**	**165m**	**53m**	**2856**	**R**	**4.6**	**0.2**	**0:00.53**	**cc1**
8127	**mskinner**	**20**	**0**	**150m**	**38m**	**3028**	**R**	**4.6**	**0.2**	**0:00.35**	**cc1**
8354	**mskinner**	**20**	**0**	**132m**	**20m**	**2812**	**R**	**4.6**	**0.1**	**0:00.17**	**cc1**
7540	**mskinner**	**20**	**0**	**155m**	**46m**	**5748**	**R**	**4.3**	**0.2**	**0:00.69**	**cc1**

mskinner 6x slower than root

---

root time:
real 5m27.713s
user 14m21.436s
sys 2m22.314s

mskinner time:
real 32m28.691s
user 17m57.014s
sys 4m20.604s

Simple CPU Example: where are tasks?

- [root@rh6-huge cpulimit]# cat /cgroup/cpu/cpulimit/tasks
  - 20576
  - 24510
  - 24511
  - 24512
  - 24513
- [root@rh6-huge rootlimit]# cat /cgroup/cpu/rootlimit/tasks
  - 1457
  - 1681
  - 6115
  - 9723
  - 9735
  - 13219

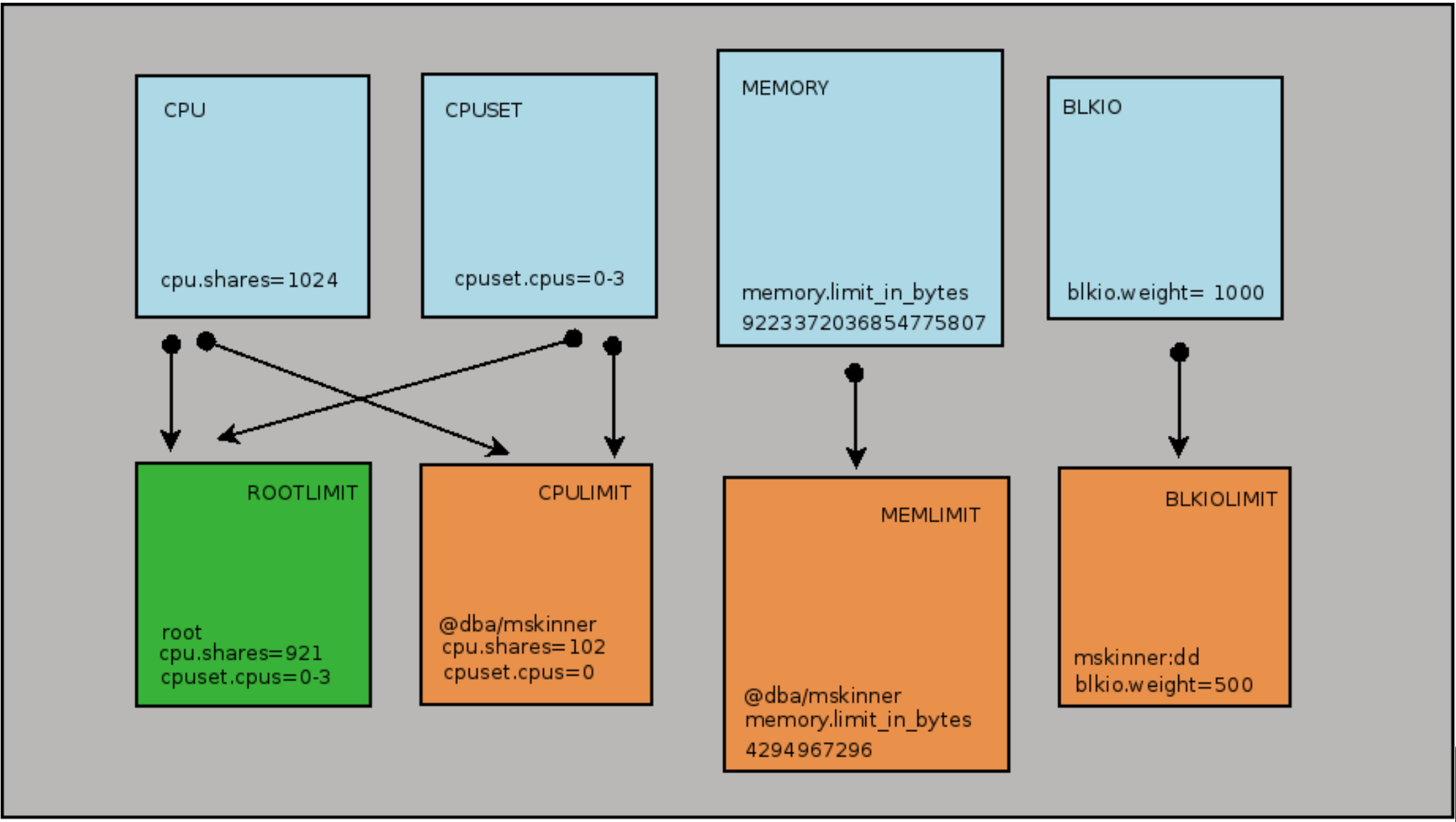
Simple Blkio Example: create limits and rules

- Create rule for 50% block IO for user mskinner running any dd command
- /etc/cgconfig.conf
  - group blkiolimit {
  - 	blkio {
  - 		blkio.weight = 500;
  - 	}
  - }
- /etc/cgrules.conf
  - mskinner:dd	blkio	blkiolimit/

Simple Blkio Example: restart services

- Restart cgconfig and cgred

CGroup Examples : BLKIO



Simple Blkio Example: check limits

[root@rh6-huge /]#cgget blkiolimit

.

.

.

blkio.weight: 500

Simple Blkio Example: run tests

- [root@rh6-huge ~]# time dd if=/dev/zero of=file_1 bs=1M count=1024
  - 1024+0 records in
  - 1024+0 records out
  - 1073741824 bytes (1.1 GB) copied, 13.4497 s, 79.8 MB/s
  - real 0m13.460s
  - user 0m0.008s
  - sys 0m1.439s

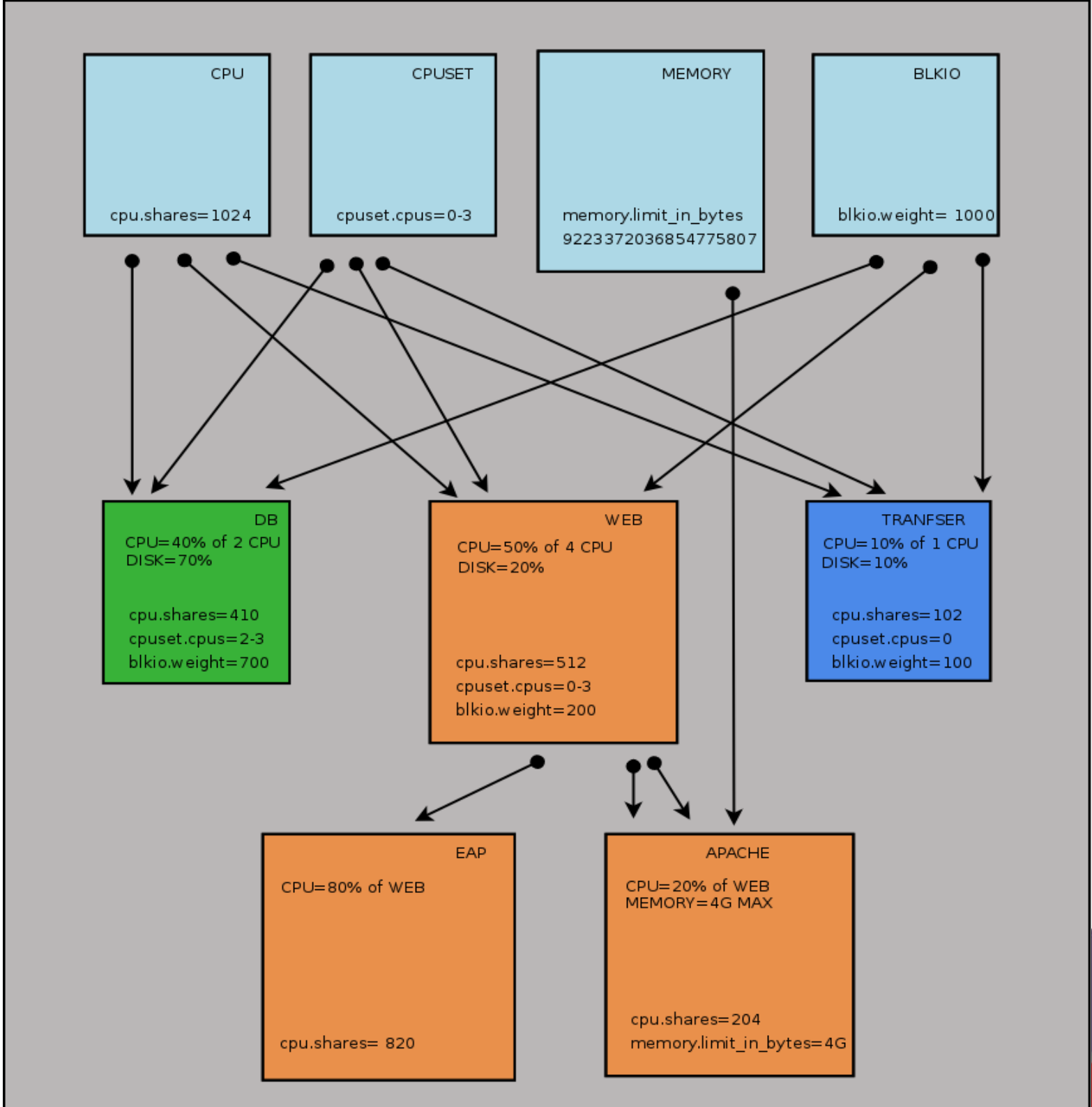
Hint: Make sure to do a "sync" and "echo 3 > /proc/sys/vm/drop_caches"

- [mskinner@rh6-huge ~]$ time dd if=/dev/zero of=file_1 bs=1M count=1024
  - 1024+0 records in
  - 1024+0 records out
  - 1073741824 bytes (1.1 GB) copied, 30.4172 s, 35.3 MB/s
  - real 0m30.709s
  - user 0m0.004s
  - sys 0m5.095s

Putting it all together!

- **WEB Parent**
  - 50% of 4 cores, 20% of DISK
- **Jboss EAP Service**
  - 80% of WEB CPU
- **HTTP Service**
  - 20% of WEB CPU, with max 4G MEMORY
- **Mysql Service**
  - 40% of 2 cores, 70% of DISK
- **SFTP Service**
  - 10% of 1 core, 10% of DISK

EAP CGroup Examples

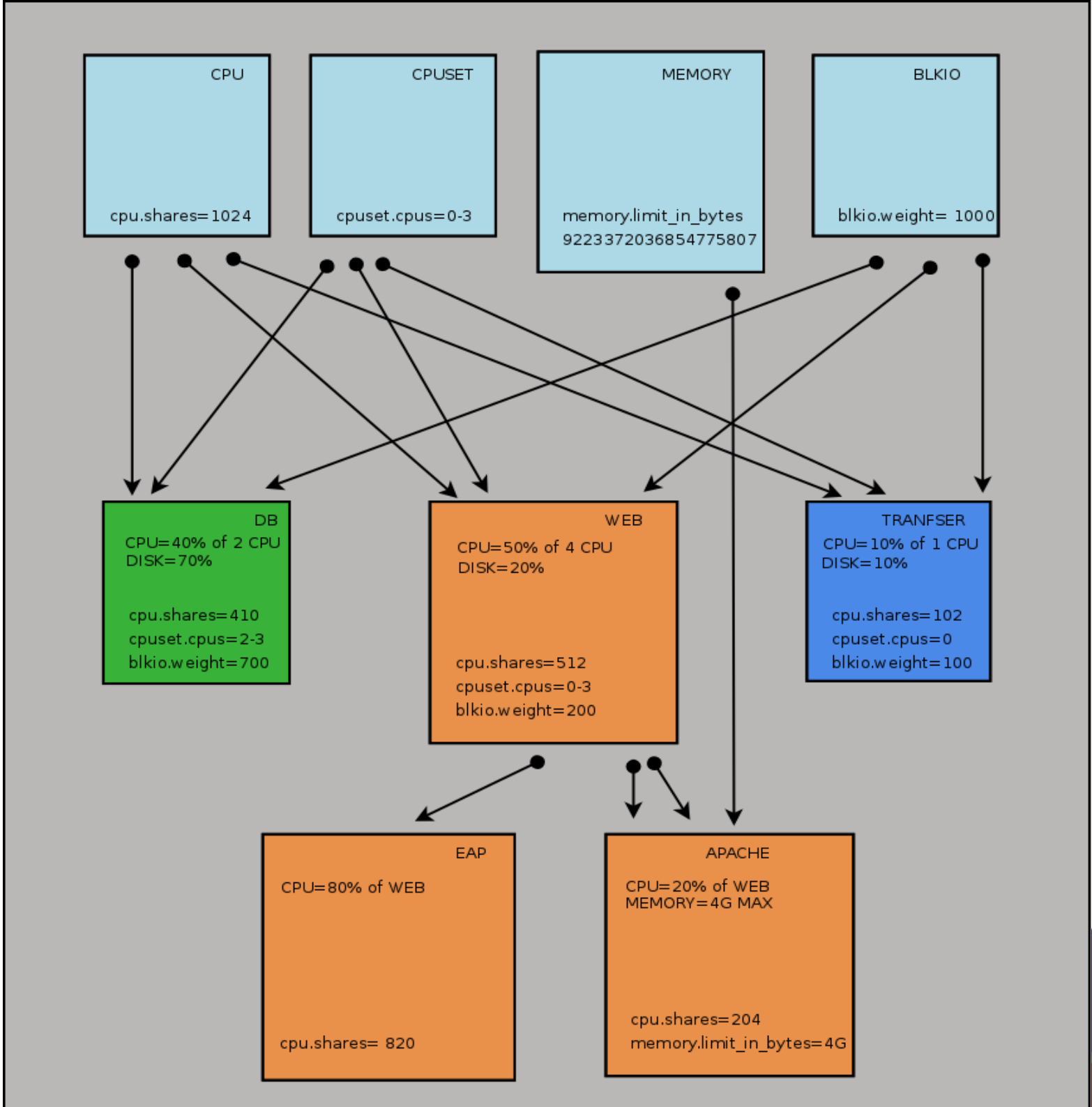


WEB Parent (50% of 4 cores, 20% of DISK)

- /etc/cgconfig.conf
  - group web {
  - cpuset {
  - cpuset.cpus = 0-3;
  - }
  - cpu {
  - cpu.shares = 512;
  - }
  - blkio {
  - blkio.weight = 200;
  - }
  - }

EAP CGroup Examples

WEB Parent

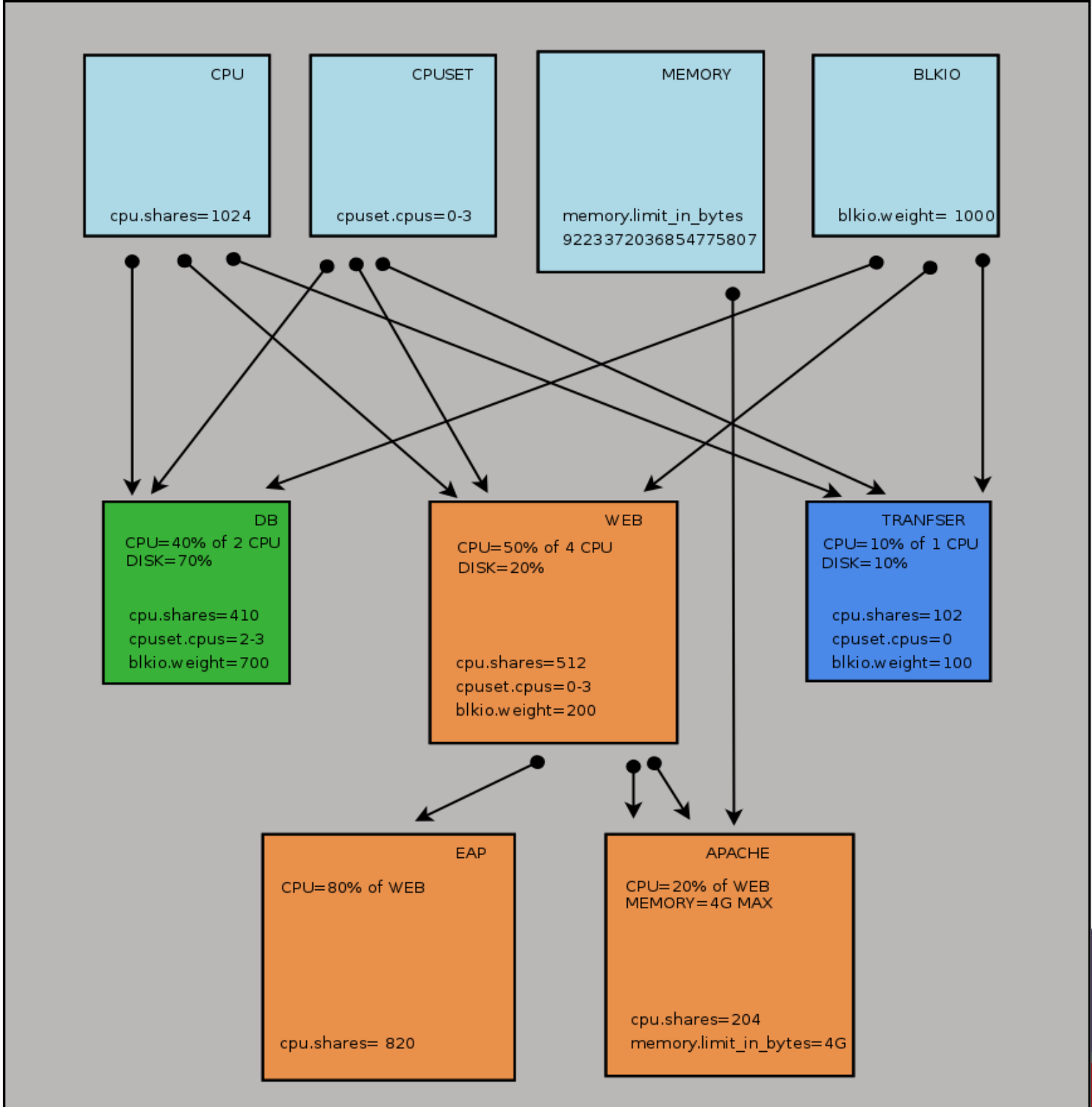


Jboss EAP Service (80% of WEB CPU)

- Create user/group for EAP Service
  - useradd eap
- Run EAP as user eap
- /etc/cgconfig.conf
  - group web/eap {
  - cpu {
  - cpu.shares = 820;
  - }
  - }
- /etc/cgrules.conf
  - @eap cpu web/eap

EAP CGroup Examples

EAP Service

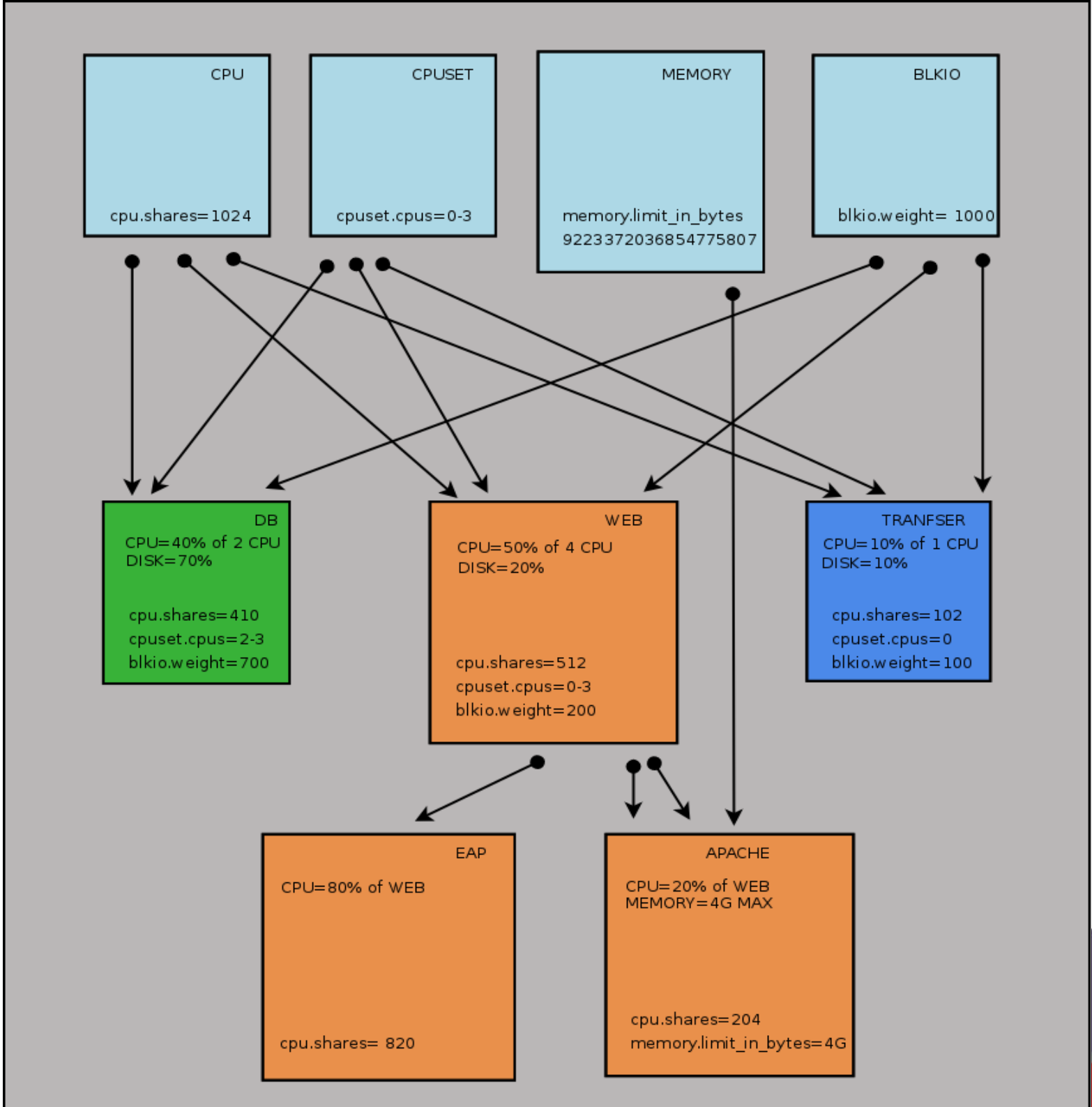


HTTP Service
(20% of WEB CPU, max 4G MEMORY)

- /etc/cgconfig.conf
  - group web/apache {
  - cpu {
  - cpu.shares = 204;
  - }
  - memory {
  - memory.limit_in_bytes = 4G;
  - }
  - }
- /etc/cgrules.conf
  - @apache cpu,memory web/web

EAP CGroup Examples

HTTP Service

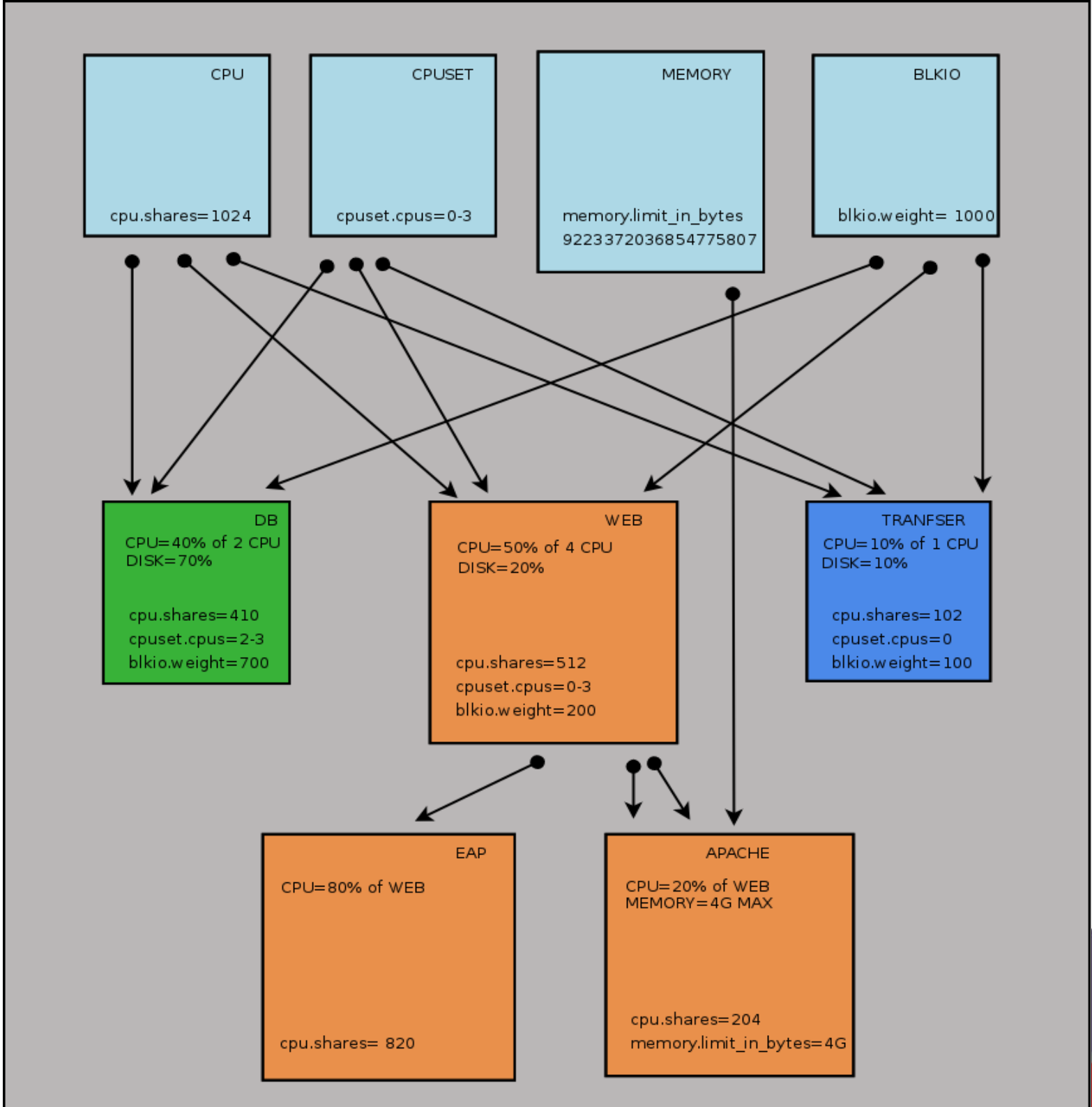


Mysql Service (40% of 2 cores, 70% of DISK)

- /etc/cgconfig.conf
  - group db {
  - cpuset {
  - cpuset.cpus = 2-3;
  - }
  - cpu {
  - cpu.shares = 700;
  - }
  - blkio {
  - blkio.weight = 700;
  - }
  - }
- /etc/cgrules.conf
  - @mysql cpu,cpuset,blkio db

EAP CGroup Examples

DB Service

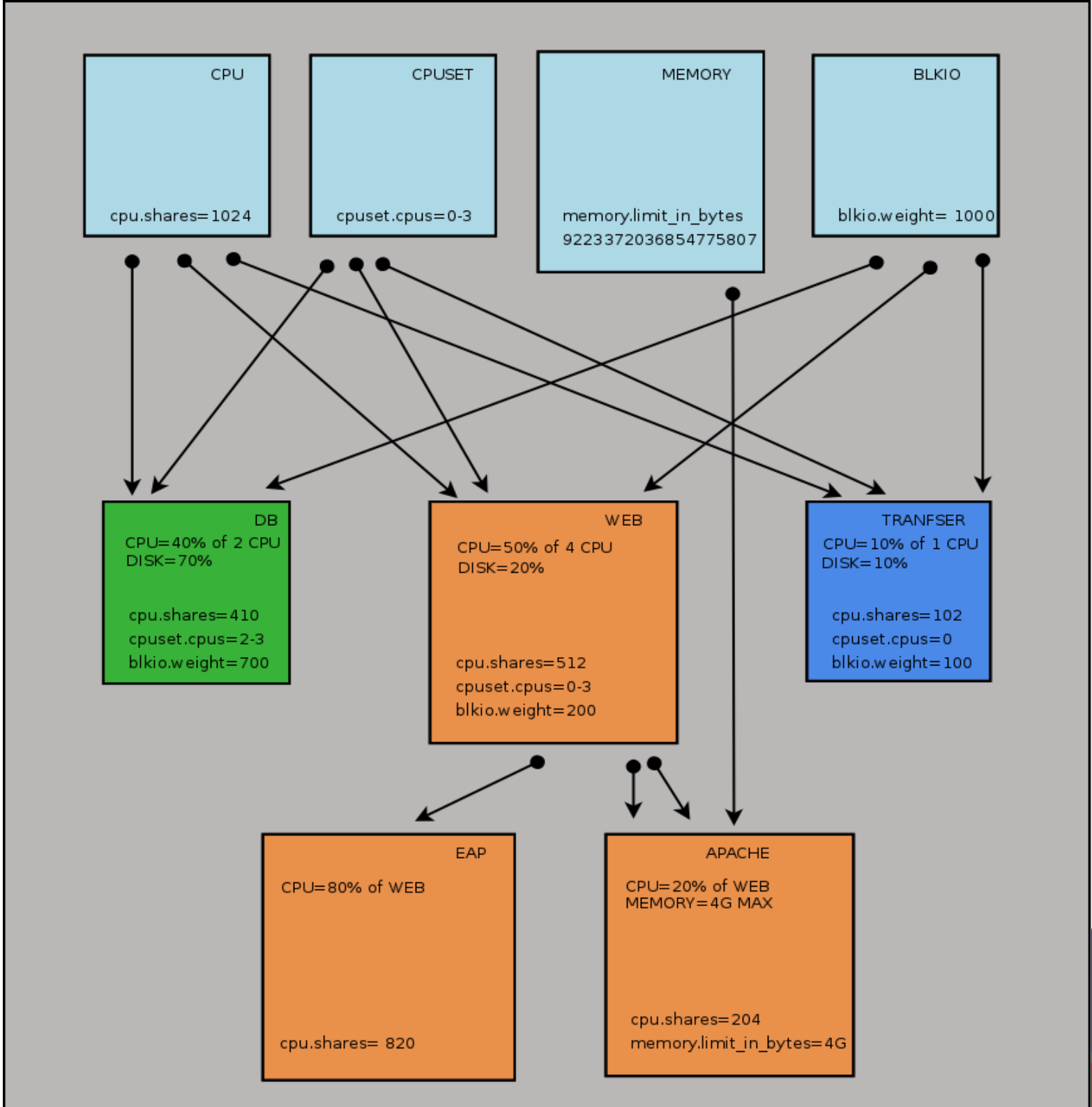


SFTP Service (10% of 1 core, 10% of DISK)

- /etc/cgconfig.conf
  - group transfer {
  - cpuset {
  - cpuset.cpus = 0;
  - }
  - cpu {
  - cpu.shares = 102;
  - }
  - blkio {
  - blkio.weight = 100;
  - }
  - }
- /etc/cgrules.conf
  - root:sftp-server cpu,cpuset,blkio transfer

EAP CGroup Examples

SFTP Service



More commands

- **cgclassify –** move running tasks to given cgroup
- **cgexec –** run the task in a given cgroup
- **cgset –** set the parameters of a cgroup
- **cgget –** print parameters of a cgroup
- **cgcreate –** create a new cgroup
- **cgdelete –** remove a cgroup

References:

https://access.redhat.com/knowledge/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Resource_Management_Guide

https://access.redhat.com/knowledge/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Resource_Management_Guide/ch-Subsystems_and_Tunable_Parameters.html

http://www.kernel.org/doc/Documentation/cgroups/