Subject: [PATCH 0/3][RFC] Containers: Pagecache accounting and control subsystem (v1)
Posted by Vaidyanathan Srinivas on Wed, 21 Feb 2007 14:24:51 GMT
View Forum Message <> Reply to Message

Containers: Pagecache accounting and control subsystem (v1)

This patch adds pagecache accounting and control on top of
Paul's container subsystem v7 posted at
http://lkml.org/lkml/2007/2/12/88

and Balbir's RSS controller posted at
http://lkml.org/lkml/2007/2/19/10

This patchset depends on Balbir's RSS controller and cannot
work independent of it. The page reclaim code has been merged
with container RSS controller. However compile time options
can individually enable/disable memory controller and/or
pagecache controller.

Comments, suggestions and criticisms are welcome.

Features:
---
* New subsystem called 'pagecache_acct' is registered with containers
* Container pointer is added to struct address_space to keep track of
  associated container
* In filemap.c and swap_state.c, the corresponding container's
  pagecache_acct subsystem is charged and uncharged whenever a new
  page is added or removed from pagecache
* The accounting number include pages in swap cache and filesystem
  buffer pages apart from pagecache, basically everything under
  NR_FILE_PAGES is counted as pagecache. However this excluded
  mapped and anonymous pages
* Limits on pagecache can be set by echo 100000 > pagecache_limit on
  the /container file system. The unit is in kilobytes
* If the pagecache utilisation limit is exceeded, pagecache reclaim
  code is invoked to recover dirty and clean pagecache pages only.

Advantages:
---
* Does not add container pointers in struct page

Limitations:
---
* Code is not safe for container deletion/task migration
* Pagecache page reclaim needs performance improvements

* Global LRU is churned in search of pagecache pages

Usage:
---

* Add patch on top of Paul container (v7) at kernel version 2.6.20
* Enable CONFIG_CONTAINER_PAGECACHE_ACCT in 'General Setup'
* Boot new kernel
* Mount container filesystem
 mount -t container /container
 cd /container
* Create new container
 mkdir mybox
 cd /container/mybox
* Add current shell to container
 echo $$ > tasks
* There are two files pagecache_usage and pagecache_limit
* In order to set limit, echo value in kilobytes to pagecache_limit
 echo 100000 > pagecache_limit
 #This would set 100MB limit on pagecache usage
* Trash the system from current shell using scp/cp/dd/tar etc
* Watch pagecache_usage and /proc/meminfo to verify behavior

* Only unmapped pagecache data will be accounted and controlled.
  These are memory used by cp, scp, tar etc. While file mmap will
  be controlled by Balbir's RSS controller.

ToDo:
----

* Merge with container RSS controller and eliminate redundant code
* Support and test task migration and container deletion
* Review reclaim performance
* Optimise page reclaim

--