# Linux Storage, File Systems & Memory Management Summit: What is Coming Soon

Ric Wheeler
Senior Engineering Manager
File and Storage Team
Red Hat

# Overview

- What is the LSF/MM Summit?
- Performance Challenges
- Correctness and Error Handling
- Ease of Use and Management
- New Features & Updates

# What is the LSF/MM Summit?

# 2013 Linux Storage, File and Memory Management Summit

- Invitation only event run by the Linux kernel community
  - Three tracks – file, storage and memory management
  - 97 attendees in our San Francisco event in April, 2013
- Goal is to review and work out hot issues face to face
  - Not a conference with presentations
- Sponsors get a seat at the table
- Three wonderful LWN.net scribes covered the three tracks
- Each slide will point to the LWN details if it exists
- Overview of LSF/MM at large:
  - http://lwn.net/Articles/548089/

# Why Not Three Independent Events?

- Reflects the actual path of data from an application to real storage
- Joint sessions are critical to work out API's between the layers
- Highlights actual “use cases” instead of isolated kernel features

# Who Was There?

- 28 Companies represented:
  - Red Hat – 19
  - Google – 12
  - SUSE, Oracle  - 8
  - Intel, FusionIO – 5
- Track popularity:
  - File system – 42
  - Storage – 32
  - Memory management – 23
- The three LWN.net scribes keep us honest!

# Linux Storage & File & MM Summit 2013

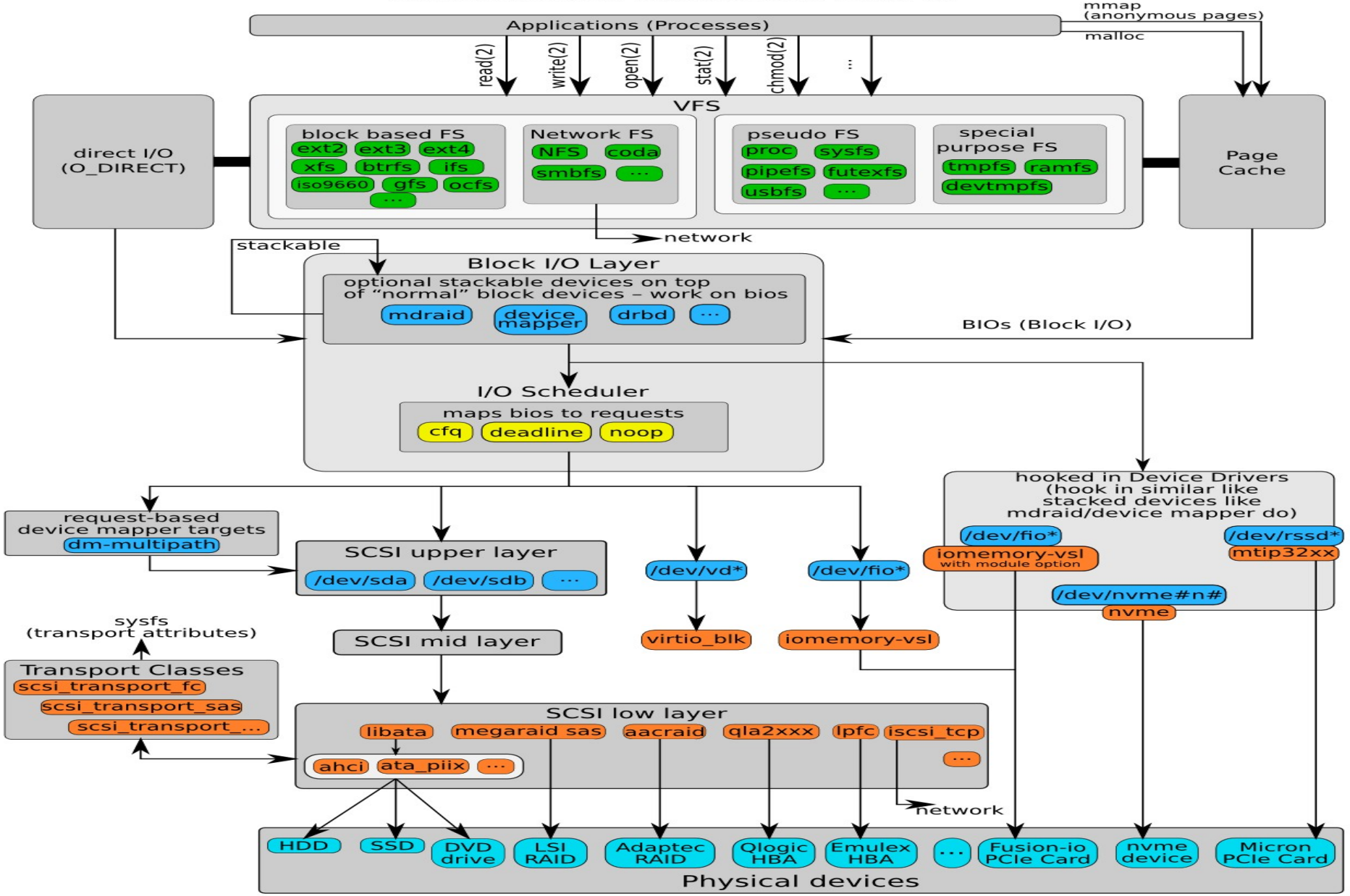
# Performance Challenges

# Storage Devices are too Fast for the Kernel!

- We are too slow for modern SSD devices
  - The Linux kernel did pretty well with just S-ATA SSD's
  - PCI-e SSD cards can sustain 500,000 or more IOPs and our stack is the bottleneck in performance
- A new generation of *persistent memory* is coming that will be
  - Roughly the same capacity, cost and performance as DRAM
  - The IO stack needs to go to millions of IOPs
  - http://lwn.net/Articles/547903

# The Linux I/O Stack Diagram

version 1.0, 2012-06-20
outlines the Linux I/O stack as of Kernel version 3.3



# Performance Work: SBC-4 Commands

- SBC-4 is a reduced & optimized SCSI disk command set proposed in T10
  - Current command set is too large and supports too many odd devices (tapes, USB, etc)
  - SBC-4 will support just disks and be optimized for low latency
  - Probably needs a new SCSI disk drive
- New atomic write command from T10
  - All of the change or nothing happens
  - Supported by some new devices like FusionIO already
- http://lwn.net/Articles/548116/

# More Performance Work

- Jens Axboe continues to work on a new, multi-queue IO subsystem
  - Learns from the lessons of the 10 gigabit ethernet world
  - Drives “opt in” to new IO stack by adding support for a new *make_request()* call
- New, standard drivers for PCIe class NVM devices
  - NVMe driver landed in the upstream kernel already from Intel
  - SCSI express PCIe driver from HP
    - https://github.com/HPSmartStorage/scsi-over-pcie

# Dueling Block Layer Caching Schemes

- With all classes of SSD's, the cost makes it difficult to have a purely SSD system at large capacity
  - Obvious extension is to have a block layer cache
- Two major upstream choices:
  - Device mapper team has a dm-cache target in 3.9
  - BCACHE queued up for 3.10 kernel
- Performance testing underway
  - BCACHE is finer grained cache
  - Dm-cache has a pluggable policy (similar to dm MPIO)
- https://lwn.net/Articles/548348

# IO Hints

- Storage device manufacturers want help from applications and the kernel
  - Tag data with hints about streaming vs random, boot versus run time, critical data
  - T10 standards body proposed SCSI versions which was voted down
- Suggestion raised to allow hints to be passed down via struct bio from file system to block layer
  - Support for expanding fadvise() hints for applications
  - No consensus on what hints to issue from the file or storage stack internally
- http://lwn.net/Articles/548296/

# Correctness and Error Handling

# Update on Data Protection Mechanism

- SCSI T10 supports
  - T10 DIF which adds bits between the HBA and the storage target
  - T10 DIX which cover the entire path from application down to hook into the DIF portion
- No standard user space API to allow DIX information to be set by applications
  - Oracle has a proprietary, shipping oracleasm driver with support
  - Darrick Wong working to evaluate sys_dio mechanism or a new kernel AIO
  - Existing DIX is very block oriented
- http://lwn.net/Articles/548294

# Time to Fix O_DIRECT?

- O_DIRECT allows an application to by pass the page cache
  - Used by high performance, sophisticated applications like databases
- The mechanism is very fragile and complicated
  - Cause of complicated hacks in btrfs to make it work
  - Very inefficient for XFS as well
- Proposed new version 2 of O_DIRECT will be investigated by Joel Becker
- http://lwn.net/Articles/548351

# Improving Error Return Codes?

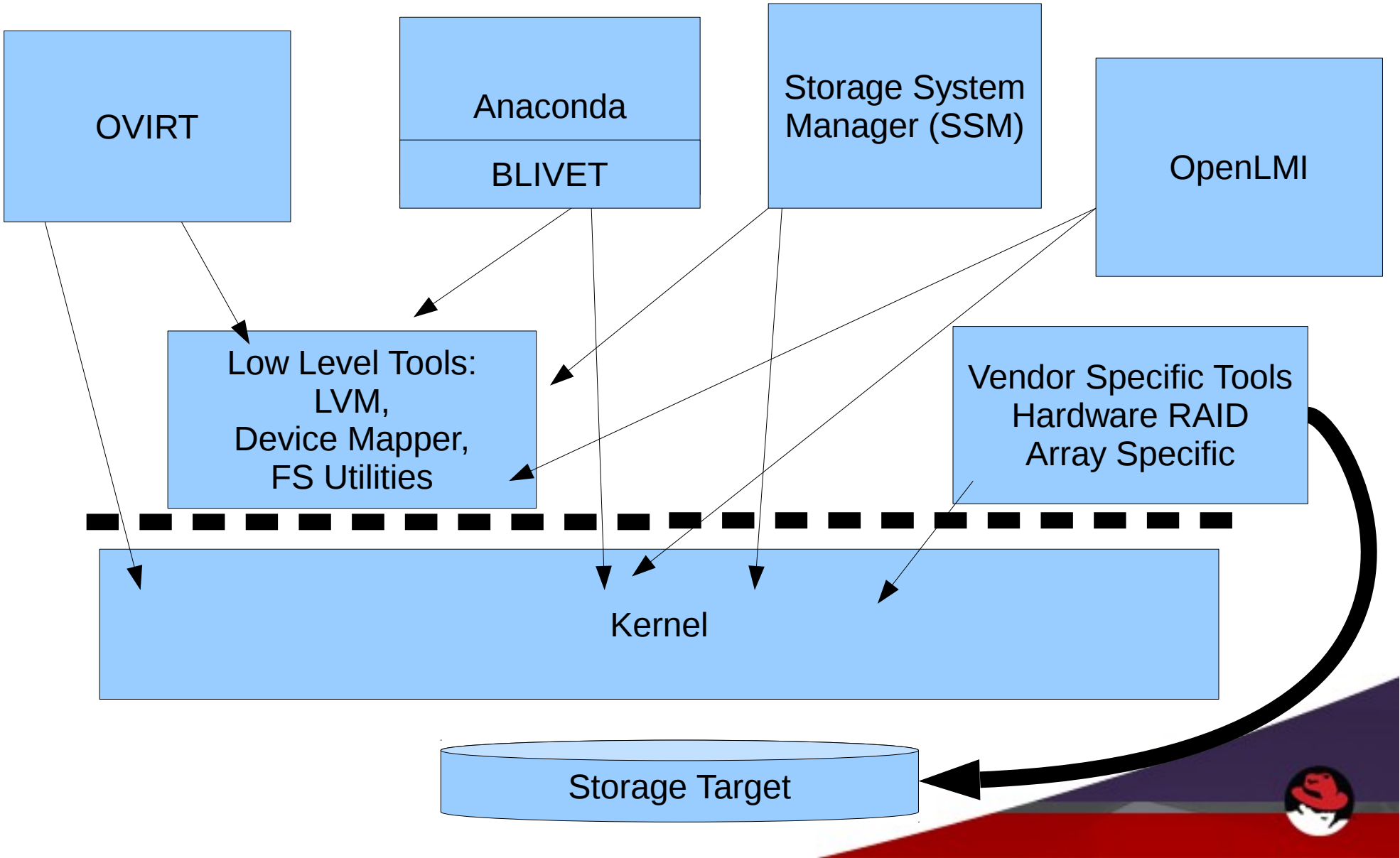
- The interface from the IO subsystem up to the file system is pretty basic
  - Low level device errors almost always propagate as EIO
  - Causes file system to go offline or read only
  - Makes it hard to do intelligent error handling at FS level
- Suggestion was to re-use select POSIX error codes to differentiate from temporary to permanent errors
  - File system might retry on temporary errors
  - Will know to give up immediately on others
  - Challenge is that IO layer itself cannot always tell!
- http://lwn.net/Articles/548353

# Ease of Use and Management

# Thinly Provisioned Storage & Alerts

- Thinly provisioned storage lies to users
  - Similar to DRAM versus virtual address space
  - Sys admin can give all users a virtual TB and only back it up with 100GB of real storage for each user
- Supported in arrays & by device mapper dm-thinp
- Trouble comes when physical storage nears its limit
  - Watermarks are set to trigger an alert
  - Debate is over where & how to log that
  - How much is done in kernel versus user space?
- User space policy agent was slightly more popular
- http://lwn.net/Articles/548295

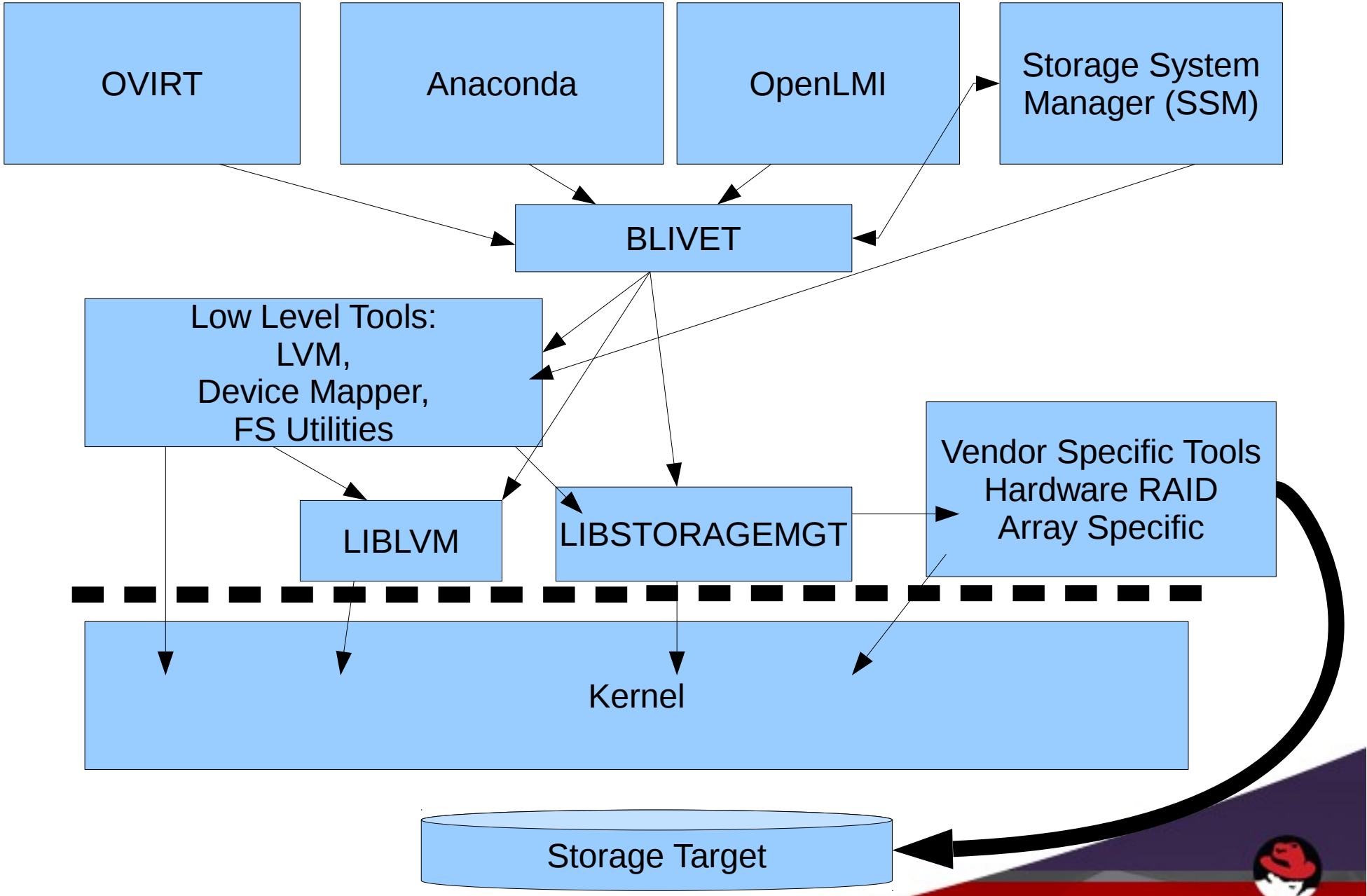
# Current Red Hat Storage Management Stack



# Ongoing Work on Storage Management

- Storage and file systems are difficult and complex to manage
  - Each file system and layer of the stack has its own tools
  - Different tools at installation time and during run time
  - No C or Python bindings
- Multiple projects have been ongoing
  - SUSE Snapper manages btrfs and LVM snapshots
  - libstoragemgt, liblvm,targetd libraries being developed
  - System Storage Manager
- http://lwn.net/Articles/548349

# Future Red Hat Stack Overview



# High Level Storage Management Projects

- Storage system manager project
  - CLI for file systems
  - http://storagemanager.sourceforge.net
- openlmi allows remote storage management
  - https://fedorahosted.org/openlmi/
  - http://events.linuxfoundation.org/images/stories/slides/lfcs2013_gallagher.pdf
- Ovirt project focuses on virt systems & their storage
  - http://www.ovirt.org/Home
- Installers like yast or anaconda

# Low Level Storage Management Projects

- Blivet library provides a single implementation of common tasks
  - Higher level routines and installers will invoke blivet
  - https://git.fedorahosted.org/git/blivet.git
  - Active but needs documentation!
- libstoragemgt provides C & Python bindings to manage external storage like SAN or NAS
  - http://sourceforge.net/p/libstoragemgmt/wiki/Home
  - Plans to manage local HBA's and RAID cards
- Liblvm provides C & Python bindings for device mapper and lvm
  - Project picking up after a few idle years

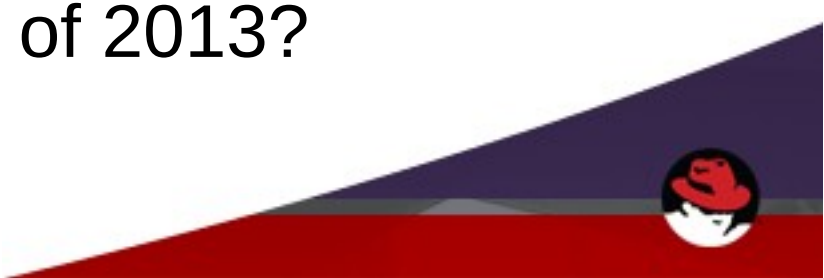
# New Features and Updates

# Copy Offload System Calls

- Upstream kernel community has debated “copy offload” for several years
  - Popular use case is VM guest image copy
- Proposal is to have one new system call
  - int copy_range(int fd_in, loff_t, upos_in, int fd_out, loff_t upos_out, int count)
  - Offload copy to SCSI devices, NFS or copy enabled file systems (like reflink in OCFS2 or btrfs)
- Patches for copy_range() posted by Zach Brown
  - https://lkml.org/lkml/2013/5/14/622
- http://lwn.net/Articles/548347

# State of BTRFS

- Distribution support for btrfs
  - SLES supports btrfs only as a system partition
  - SLES forbids use of btrfs RAID or multi-devices
  - Red Hat has btrfs in tech preview in RHEL6
- Upstream is focused mostly on bug fixing
  - RAID stripes are known to be not power failure safe
  - Still see a fair high volume of user bug reports
- Data duplication possibly in 3.11 & fix for RAID data integrity persistence
- Fully production ready by the end of 2013?
- https://lwn.net/Articles/548937



# New NFS Features

- Labeled NFS allow fine grain security labels for selinux
  - Server and client code should be upstream by 3.11
  - Patches have been tested over several years
- Support for NFS V4.1 largely finished
  - Client side support optional pNFS mode, server does not
- Starting to work on implementation of NFSv4.2
  - Support for the copy offload is in V4.2
  - Hole punching, support for fallocate over NFS
  - V4.2 features mostly optional
- https://lwn.net/Articles/548936

# Resources & Questions

- Resources
  - Linux Weekly News: http://lwn.net/
  - Mailing lists like linux-scsi, linux-ide, linux-fsdevel, etc
- SNIA NVM TWG
  - http://snia.org/forums/sssi/nvmp
- Storage & file system focused events
  - LSF/MM workshop
  - Linux Foundation & Linux Plumbers Events