GNU parted

GNU parted
Version 2.0 Development Plans
David Cantrell <dcantrell@redhat.com>
FOSDEM 2007

David Cantrell

- Software engineer at Red Hat working on Fedora and Red Hat Enterprise Linux
- Development work includes anaconda, GNU parted, and DHCP software
- Former developer on the Slackware Linux project
- Project co-maintainer for GNU parted
- Usually write C and Python code

GNU parted

- Project overview and scope
- Storage methodologies
- Current features
- Current limitations
- Goals for version 2.0
- Plans for version 1.x
- Contributors

Project Overview and Scope

- GPL library for partition and filesystem management
- Command line program provided as a replacement for fdisk(8)
- Support as many filesystems, disk labels, and operating systems as possible
- Support creating, destroying, copying, moving, and resizing partitions and filesystems

Storage Methodologies

- **Disks** - physical devices
- **Labels** - describe the disk layout
- **Partitions** - disk divisions
- **Filesystems** - organization system
- 'slice' == 'partition'

- Seagate, Maxtor, IBM, Western Digital
- DOS, Macintosh, Sun, BSD
- Primary, Extended, Swap, Whole Disk
- DOS, FAT, ext3, NTFS, FFS, HFS+

Storage Methodologies (cont.)

- RAID and LVM present new challenges
- RAID devices translate to Disks for the most part
- RAID mirrors may need special treatment since breaking the mirror still preserves the disk label
- LVM volume groups == disks
- LVM logical volumes == partitions
- LVM volume group vs. normal block device
- Removable media

Current Features

- Multiple architectures supported
- Wide range of label types supported
  - Examples: DOS, GPT, Mac, Sun, BSD
- Wide range of filesystems supported
  - Examples: Ext2, FAT, HFS
- Same commands and library calls regardless of platform
- Functionality contained in libparted rather than a userspace program

Current Limitations

- Limited filesystem operation support
- e.g., Lacking ext3 creation or resize
- No direct NTFS creation support
- No direct HFS+ creation support
- Library bindings only available for C
- Poor exception handling in libparted
- Lack of good API documentation
- Parted(8) lacks certain features in fdisk(8)

Goals for Version 2.0

- Support non-size boundary specifications (CHS)
- API overhaul
- Improve VFS layer to support more fs features
- Remove legacy fs code and use existing libraries
- Unit testing framework
- Add library bindings for other languages
  - Obsolete pyparted
  - Allow library usage in more applications
  - Possibly autogenerate direct C bindings, then build on that

Goals for Version 2.0 (cont.)

- Improve API documentation
- Improve error handling throughout libparted
- Side projects
  - Exposing LVM devices as virtual block devices
  - Collaboration with other projects using libparted

Plans for Version 1.x

- Version 1.8 will continue in maintenance mode
- No major features introduced in 1.x
- Bug fixes backported from mainline tree
- Help developers ease transition to using libparted 2.0
- Allow commercial distributions to continue with 1.x branch as needed

Contributors

- Translators needed for documentation
- Coders for languages other than C needed for library binding development
- Developers experienced with non-Linux operating systems needed
- Developers using libparted in other projects; we'd like to hear what you like/dislike about the API
- Parted package maintainers; send bug reports and patches upstream!

References

- GNU parted Home Page
  - http://www.gnu.org/software/parted/
- Mailing List
  - parted-devel@lists.alioth.debian.org
- IRC
  - #parted on FreeNode