HelenOS: State of the Union

Martin Děcký

DEPARTMENT OF DISTRIBUTED AND DEPENDABLE SYSTEMS
http://d3s.mff.cuni.cz/

CHARLES UNIVERSITY IN PRAGUE
FACULTY OF MATHEMATICS AND PHYSICS
A Few Figures

- **Physical lines of code:** 146,054
  - C: 132,738 (90.88 %)
  - Assembler: 8,278 (5.67 %)
  - Python: 2,948 (2.02 %)
  - Make: 1,437 (0.98 %)
  - Shell: 461 (0.32 %)
- **Ratio of comments:** 38 %
- "Extremely well-commented source code" (by Ohloh.net)
A Few Figures (2)

- **Number of files:** 1,688
  - Four-clause BSD license: 97.90 %
  - GNU GPL 2.0 license: 2.10 %
- **Number of commits:** 5,074
  - Subversion repo (since 18-02-2005): 4,759 (93.79 %)
  - Bazaar mainline (since 15-08-2009): 315 (6.21 %)
    - 2.74 commits/day on average
A Few Figures (3)

- **Number of commiters:** 16
  - MFF / unknown: 8
  - MFF / Sun: 3
  - Freelance: 2
  - MFF / DSRG: 1
  - Sun: 1

- **Number of contributors:** 29
  - Including derived works
Ave, Contributors!

Dmitry Bolkhovityanov  
Hewlett-Packard Co.  
Intel Corporation  
Jakub Jermar  
Jakub Vana  
Jan Hudecek  
Jiri Svoboda  
Josef Cejka  
Lubomir Bulej  
Lukas Mejdrech  
Martin Decky  
Martin Jelen  
Martin Mares  
Michal Kebrt  
Michal Konopa

Ondrej Palkovsky  
Ondrej Sery  
Pavel Jancik  
Pavel Rimsky  
Peter Majer  
Petr Stepan  
Petr Tuma  
Sergey Bondari  
The NetBSD Foundation, Inc.
Tim Post  
Tomas Bures  
Vineeth Pillai  
Vojtech Mencl  
Vrije Universiteit
1. Go to http://www.helenos.org/download
2. Fetch the ISO image for *amd64* or *ia32*
3. Optionally: Burn the image on a CD
4. Boot the image or CD

   Any decent PC or virtual machine should do
5. Play!
6. Go to http://trac.helenos.org/

   Get the sources from bzr://bzr.helenos.org/mainline
Let's Ride!

- Boot HelenOS in Qemu (preferably KVM)
  
  \texttt{qemu-kvm -cdrom HelenOS-0.4.2-ia32.iso}

  - You can observe the user test interface, there is a shell prompt (Brain Dead Shell)
  - The circles in top of the screen represent virtual consoles
    - Empty circle = detached console
    - Circle with number = attached console
    - Red circle = attached console with activity
  - You can switch between the virtual consoles using function keys (F1 – F11)
Let's Ride! (2)

- On virtual console 7 (press F7) you can see \textit{klog}
  - \textit{Klog} displays the contents of the kernel log
    - Kernel writes system messages to the kernel log
    - User space components write messages to the kernel log when the user space infrastructure is not yet ready
- Let's try some simple shell commands and run some tasks
  - \texttt{pwd}
  - \texttt{ls}
  - \texttt{ls /app}
  - \texttt{bdsh} (run a subshell)
  - \texttt{exit} (exit the subshell)
  - \texttt{cat readme}
  - \texttt{edit readme}
Let's Ride! (3)

- Unicode character set is universally supported (in both kernel and user space)
- You can use Ctrl+C and Ctrl+V to transfer texts between the editor and the shell or various instances of the editor and the shell
  - You can define the selection via Shift+Arrow Keys
- You can switch keyboard maps using Ctrl+Fn
  - Ctrl+F1 = US QWERTY
  - Ctrl+F2 = US Dvorak
  - Ctrl+F3 = Czech QWERTZ
Let's Ride! (4)

- Walk /dev filesystem

  
  ls /dev             (namespaces)
  ls /dev/bd         (block devices)
  ls /dev/null       (null devices, for distinction)
  ls /dev/char       (character devices, PS/2)
  ls /dev/hid_in     (input devices, keyboard and mouse)
  ls /dev/term       (virtual consoles)

- Mount point of devfs

  - Shows device nodes (registered to device mapper) to the user
  - Some of these device nodes actually implement file semantics (virtual consoles, etc.)

  getterm /dev/term/vc10 /app/bdsh
  (spawn a new shell instance on virtual console 11)
Let's Ride! (5)

- Stdin/stdout/stderr redirection

  tester (show available tests)
  tester thread1 (simple threading test)
  redir -e tester_err -- /app/tester thread1
  (redirect stderr of tester to a file tester_err)
  cat tester_err (display the contents)
Let's Ride! (6)

- Kernel console (press F12)
  - Only in debugging builds, not preset in release builds
  - For low-level debugging of the system
  - A jump-in replacement for lacking user space functionality (which is not essential)
  - Does not affect scheduling, user space tasks are still running
    threads (print list of threads)
    tasks (print list of tasks)
    continue (leave the kernel console, switch to user space)
    help (show other kernel console commands)
Let's Ride! (7)

- Debugging and tracing
  
  `taskdump -t 23`  (show stack trace of task 23)
  
  `taskdump -c core -t 23`  (dump core of task 23)
  
  `trace +t -t 23`  (trace thread events of task 23)
  
  `trace +s -t 23`  (trace syscalls of task 23)
  
  `trace +i -t 23`  (trace low-level IPC communication)
  
  `trace +p 23`  (trace high-level IPC protocols)
Let's Ride! (8)

- Access an ATA hard drive in Qemu/KVM
  - In Linux
    
    ```
    dd if=/dev/zero of=hd.img bs=1024 count=102400
    parted hd.img
    mklabel msdos
    mkpartfs primary fat16 0s -1s
    qemu-kvm -boot d -hda hd.img -cdrom HelenOS-0.4.2-ia32.iso
    ```
  
  - In HelenOS
    
    ```
    ata_bd
    ls /dev/bd
    mbr_part bd/disk0
    mkdir /mnt
    mount fat /mnt bd/disk0p0
    ```
Let's Ride! (9)

- Mount loopback filesystem

  ```
  mkfile --size 1048576 fat.img
  file_bd fat.img bd/loop0
  mkfat bd/loop0
  mkdir /mnt
  mount fat /mnt bd/loop0
  ```
Let's Ride! (10)

- Networking
  - In Linux
    - `contrib/conf/net-qe.sh` (Qemu with network device)
  - In HelenOS
    - `netstart` (initialize networking)
    - `ping 127.0.0.1`
    - `echo -t SOCK_DGRAM` (listen on UDP port 7)
  - In Linux
    - `nc -u localhost 8080`
    (UDP port 7 from Qemu virtual machine gets mapped to local UDP port 8080)
Ω ξεῖν', ἀγγέλλειν Λακεδαιμονίοις ὅτι τῆδε κείμεθα, τοῖς κείνων ῥήμασι πειθόμενοι.

Go tell the Spartans, thou who passest by, that here, obedient to their laws, we lie.