Introduction to the Programming Assignments

Philipp Kupferschmied

October 22, 2008
Philipp Kupferschmied
in charge of tutorials and programming assignments
contact
pkupfer@ira.uka.de
Wed, 15:30–17:00h
Bdg. 50.34, R.163
Longish Roadmap

1 Motivation
   - Overview
   - Bonus

2 OS/161
   - Overview
   - Requirements & Tips
   - Linux

3 Mercurial
   - Purpose
   - Concept
   - Commands
   - Remote

4 SSH
   - Passwordless Remote Login

5 GDB
   - Debugging Applications
   - Hooking GDB up with System/161
System Architecture Course Structure

1. lecture
   - broad overview
   - abstract examples

2. tutorials
   - in-depth examination
   - isolated aspects

3. programming assignments
   - implement algorithms in OS
   - experience component interaction
Tutorials

- every week
  - Wednesday, Thursday, and Friday
  - started today . . .

- online registration required

- attend the registered tutorial only
  - capacity of the rooms is quite limited
  - except on public holidays

- learn to describe and reason about abstract ideas

- discuss solutions to prepublished questions

- prepare and participate!
  - questions online on Friday
  - solutions online 1 or 2 weeks afterwards
Programming Assignments

- 4 programming assignments
  1. setup the system
  2. synchronization
  3. system calls and address spaces
  4. virtual memory subsystem
     - page-tables
     - page replacement strategies
     - swapping
- OS/161 as starting point
  - 20,000 LOC
- 3 assessable parts per assignment
  1. code reading
  2. analysis and design
  3. implementation
- 3 or 4 work weeks per assignment
Participation and Rewards

- programming assignments are voluntary
- bonus on final exam available
  - up to 4 bonus points
  - first bonus point
    - $\geq 50\%$ of code reading and
    - $\geq 50\%$ of practical parts
- groups of 2 students only
  - use the forum to find a partner
  - register your group on-line (OS/161 Groups)
- register with one of your tutors
  - via email to os161@ira.uka.de
  - your email must contain:
    - your group id ($sap\text{w}X$)
    - name of both team members
    - desired tutor
- first assignment is already out
Preliminary Marks

1. setup
   - 20 points code reading
   - 10 points implementation

2. synchronization
   - 20 points code reading
   - 15 points implementation

3. syscalls and address spaces
   - 20 points code reading
   - 15 + 5 points implementation

4. virtual memory
   - 20 points code reading
   - 20 points implementation

Assignment Submission

- code reading parts via web-form
- implementation via email to os161@ira.uka.de
Preliminary Timeline

- Programming assignment 0 is out since Monday
  - Deadline for theoretical part: 7.11.2008
  - Deadline for practical part: 14.11.2008
- Programming assignment 1 will be out on 14.11.2008
  - Deadline for theoretical part: 21.11.2008
- Programming assignment 2 will be out on 12.12.2008
  - Deadline for practical part: 16.01.2009
- Programming assignment 3 will be out on 16.01.2009
  - Deadline for theoretical part: 23.01.2009
  - Deadline for practical part: 13.02.2009
OS/161 Software Family

- **OS/161**
  - rudimentary, functional OS
  - targets MIPS
  - (most) HW dependent parts done
  - stubs or holes at interesting places

- **System/161**
  - simulates MIPS processor and
  - simplified devices
  - GDB debugging hooks

- **Cross-Compiler Toolchain**
  - cs161-gcc, -as, -ld, ...
  - cs161-gdb

- **Mercurial**
  - manage the code
  - allow concurrent development

- **SSH**
Component Interaction

Motivation OS/161 Mercurial SSH GDB
Overview Requirements & Tips Linux

Component Interaction Diagram:
- main.c
- MS/161
- cs161-gcc
- cs161-binutils
- System/161
- cs161-gdb
- GNU/Linux on x86
- .hg
- Sources

System Architecture Group
Introduction to the Programming Assignments
~/sysarch/root $ ./sys161 kernel
OS/161 base system version 1.10
Copyright (c) 2000, 2001, 2002, 2003
President and Fellows of Harvard College. All rights reserved.

Put-your-group-name-here’s system version 0 (ASST0 #1)

Cpu is MIPS r2000/r3000
512k physical memory available
Device probe...
lamebus0 (system main bus)
emu0 at lamebus0
ltrace0 at lamebus0
 :
lser0 at lamebus0
con0 at lser0
pseudorand0 (virtual)

OS/161 kernel [? for menu]: q
System Requirements

- primary platform: Linux on x86
  - e.g., ATIS machines
- reportedly works on *BSD, Mac OS X
  - requires development packages (build-essentials, gcc, binutils)
- Windows via cygwin
  - works, but unsupported
  - gcc, binutils, wget, tar, bzip2, make, bash, sed
  - probably more . . .
  - use putty to work on i08fs1 instead?
- 500 MB hard disk space
  - while building cross-compilers
  - 100 MB afterwards
  - 50 MB on ATIS
    - cross-compilers already installed!
    - see /opt/cs161/bin
Proposed Filesystem Structure

- **$HOME/**
  - **sysarch/**
    - **sharedrepos/**\(^1\)
    - **bin/**\(^2\)
    - **tools/**\(^2\)
    - **root/**
    - **asst0-src/**
      - ...
    - **asst3-src/**
    - **(all downloaded archives).tbz2**
  - simplifies clean-up half a year from now

---

\(^1\) only one of each team
\(^2\) not on ATIS machines, see /opt/cs161
General Remarks (1/2)

- start early
  - read carefully
    - assignment
    - code
  - think
    - what do you need?
    - what is available? → read again
  - implement

- work continuously
  - avoid long breaks

- communicate with your partner
  - discuss problems
  - divide the implementation tasks
test early
- not (only) 1h before deadline

write dummy stubs
- simple functionality
- support early testing
- but more work

test again
- changes may break test cases
directories
- mkdir -p relative/path/name
- cd relative/path/name
- cd ..
- ls -l

archives (tarballs)
- .bz2: tar -xjf archive.tbz2
- .gz: tar -xzf archive.tgz

build process
- ./configure [options]
- make target or simply make
- make install
  - not required for OS/161 sources
environment

- HOME: absolute path to your home
- PATH: colon separated list of paths to search for executables
  - must not contain ".
  - add /path/to/cs161-gcc

inspecting environment variables

- echo $VAR

defining environment variables

- bash: export VAR="value"
- other shells: see respective man page
login shell standard environment

- in `~/.bashrc` or `~/.profile` (or `~/.bash_login`):
  
  ```
  export PATH=/opt/cs161/bin:$HOME/sysarch/bin:$PATH
  ```
Problem Statement

- two developers edit a single file
  - same file, the one who saves last prevails
  - distinct copies, must be merged sometime

- solutions
  - editor-specific file locking
  - policy
  - source code management systems
Purpose of Source Code Management

- allow concurrent development
  - no file locking
  - support merging of work
- track project history
  - who? when? what? how?
  - why?
- restore previous file revisions
  - after disk crash
  - drop development in a dead end
  - how was it done before?
Basic Notions of Mercurial

Mercurial distributed revision control tool
HG chemical symbol of mercury, “short name” of Mercurial
repository working directory + project history
revision retrievable state of a file
changeset collection of changes that lead to a new revision
tag symbolic name for a project state
working directory contains files on which development takes place
tip most recent changeset in repository
phipp@host:~/hg-demo$ ls -la

```
total 12
-dwrxr-xr-x 3 philipp philipp 4096 2008-09-18 17:38 .
-dwrxr-xr-x 53 philipp philipp 4096 2008-09-18 17:43 ..
-dwrxr-xr-x 2 philipp philipp 4096 2008-09-18 17:38 sources
```
philipp@host:~/hg-demo$ cd sources
philipp@host:~/hg-demo/sources$ ls -la

total 12
-d-rw-r-xr-x 2 philipp philipp 4096 2008-09-18 17:38 .
-d-rw-r-xr-x 3 philipp philipp 4096 2008-09-18 17:38 ..
-rw-r--r-- 1 philipp philipp 104 2008-09-18 17:38 main.c
philipp@host:~/hg-demo/sources$ hg init
philipp@host:~/hg-demo/sources$ ls -la

```
total 16
-rw-r--r-- 1 philipp philipp 104 2008-09-18 17:38 main.c
```

Mercurial Sample Session

philipp@host:~/hg-demo/sources$ hg add
adding main.c
philipp@host:~/hg-demo/sources$ hg commit -m "Initial commit"
Mercurial Sample Session

philipp@host:~/hg-demo/sources$ hg log
changeset: 0:0391671a98ef
tag: tip
user: Philipp Kupferschmied <pk@ibds.uka.de>
date: Thu Sep 18 17:45:52 2008 +0200
summary: Initial commit
Mercurial Sample Session

```
philipp@host:~/hg-demo/sources$ vim main.c
```
Mercurial Sample Session

philipp@host:~/hg-demo/sources$ hg diff
diff -r 0391671a98ef main.c
--- a/main.c Thu Sep 18 17:45:52 2008 +0200
+++ b/main.c Thu Sep 18 17:47:31 2008 +0200
@@ -2,7 +2,8 @@
        int main( int argc, char **argv )
        {
-           printf( "Hallo Welt!\n" );
+           //printf( "Hallo Welt!\n" );
+           printf( "Hello World!\n" );

             return 0;
        }
Mercurial Sample Session

philipp@host:~/hg-demo/sources$ hg commit -m "Changed output language"
philipp@host:~/hg-demo/sources$ hg log
changeset: 1:a32fd2a82371
tag: tip
user: Philipp Kupferschmied <pk@ibds.uka.de>
date: Thu Sep 18 17:48:16 2008 +0200
summary: Changed output language

changeset: 0:0391671a98ef
user: Philipp Kupferschmied <pk@ibds.uka.de>
date: Thu Sep 18 17:45:52 2008 +0200
summary: Initial commit
phipp@host:~/hg-demo/sources$ cd ..
phipp@host:~/hg-demo$ ls -la
total 12
    drwxr-xr-x  3 philipp philipp  4096 2008-09-18 17:38 .
    drwxr-xr-x  53 philipp philipp  4096 2008-09-18 17:48 ..
    drwxr-xr-x  3 philipp philipp  4096 2008-09-18 17:48 sources
phipp@host:~/hg-demo$ hg clone sources sources2
updating working directory
1 files updated, 0 files merged, 0 files removed, 0 files unresolved
phipp@host:~/hg-demo$ ls -la
total 16
    drwxr-xr-x  4 philipp philipp  4096 2008-09-18 17:49 .
    drwxr-xr-x  53 philipp philipp  4096 2008-09-18 17:48 ..
    drwxr-xr-x  3 philipp philipp  4096 2008-09-18 17:48 sources
    drwxr-xr-x  3 philipp philipp  4096 2008-09-18 17:49 sources2
Mercurial Sample Session

philipp@host:~/hg-demo$ cd sources2
philipp@host:~/hg-demo/sources2$ hg log
changeset: 1:a32fd2a82371
tag: tip
user: Philipp Kupferschmied <pk@ibds.uka.de>
date: Thu Sep 18 17:48:16 2008 +0200
summary: Changed output language

changeset: 0:0391671a98ef
user: Philipp Kupferschmied <pk@ibds.uka.de>
date: Thu Sep 18 17:45:52 2008 +0200
summary: Initial commit
Mercurial Sample Session

philipp@host:~/hg-demo/sources2$ echo > foo.c
philipp@host:~/hg-demo/sources2$ hg add
adding foo.c
Mercurial Sample Session

```
philipp@host:~/hg-demo/sources2$ hg commit -m "Added foo.c"
philipp@host:~/hg-demo/sources2$ hg log
changeset: 2:06d64ee049a9
tag: tip
user: Philipp Kupferschmied <pk@ibds.uka.de>
date: Thu Sep 18 17:51:11 2008 +0200
summary: Added foo.c

changeset: 1:a32fd2a82371
user: Philipp Kupferschmied <pk@ibds.uka.de>
date: Thu Sep 18 17:48:16 2008 +0200
summary: Changed output language

changeset: 0:0391671a98ef
user: Philipp Kupferschmied <pk@ibds.uka.de>
date: Thu Sep 18 17:45:52 2008 +0200
summary: Initial commit
```
philipp@host:~/hg-demo/sources2$ cd ../sources
philipp@host:~/hg-demo/sources$ ls -la
  total 16
  drwxr-xr-x 3 philipp philipp 4096 2008-09-18 17:48 .
  drwxr-xr-x 4 philipp philipp 4096 2008-09-18 17:49 ..
  drwxr-xr-x 3 philipp philipp 4096 2008-09-18 17:48 .hg
  -rw-r--r-- 1 philipp philipp 138 2008-09-18 17:47 main.c

philipp@host:~/hg-demo/sources$ hg log
  changeset: 1:a32fd2a82371
  tag:        tip
  user:       Philipp Kupferschmied <pk@ibds.uka.de>
  date:       Thu Sep 18 17:48:16 2008 +0200
  summary:    Changed output language

  changeset: 0:0391671a98ef
  user:       Philipp Kupferschmied <pk@ibds.uka.de>
  date:       Thu Sep 18 17:45:52 2008 +0200
  summary:    Initial commit
Mercurial Sample Session

philipp@host:~/.hg-demo/sources$ cd ../sources2
philipp@host:~/.hg-demo/sources2$ hg push ../sources
pushing to ../sources
searching for changes
adding changesets
adding manifests
adding file changes
added 1 changesets with 1 changes to 1 files
philipp@host:~/hg-demo/sources2$ cd ../sources
philipp@host:~/hg-demo/sources$ ls -la

```
total 16
drwxr-xr-x 3 philipp philipp 4096 2008-09-18 17:48 .
drwxr-xr-x 4 philipp philipp 4096 2008-09-18 17:49 ..
drwxr-xr-x 3 philipp philipp 4096 2008-09-18 17:52 .hg
-rw-r--r-- 1 philipp philipp 138 2008-09-18 17:47 main.c
```
philipp@host:~/hg-demo/sources$ hg log
changeset: 2:06d64ee049a9
  tag: tip
  user: Philipp Kupferschmied <pk@ibds.uka.de>
  date: Thu Sep 18 17:51:11 2008 +0200
  summary: Added foo.c

changeset: 1:a32fd2a82371
  user: Philipp Kupferschmied <pk@ibds.uka.de>
  date: Thu Sep 18 17:48:16 2008 +0200
  summary: Changed output language

changeset: 0:0391671a98ef
  user: Philipp Kupferschmied <pk@ibds.uka.de>
  date: Thu Sep 18 17:45:52 2008 +0200
  summary: Initial commit
philipp@host:~/hg-demo/sources$ hg update
1 files updated, 0 files merged, 0 files removed, 0 files unresolved

philipp@host:~/hg-demo/sources$ ls -la
total 20
drwxr-xr-x 3 philipp philipp 4096 2008-09-18 17:53 .
drwxr-xr-x 4 philipp philipp 4096 2008-09-18 17:49 ..
-rw-r--r-- 1 philipp philipp 1 2008-09-18 17:53 foo.c
drwxr-xr-x 3 philipp philipp 4096 2008-09-18 17:53 .hg
-rw-r--r-- 1 philipp philipp 138 2008-09-18 17:47 main.c
Our Scenario

- set up folder accessible by both team members
  - hosted on one member’s ATIS account
- folder contains a Mercurial repository per programming assignment
- both team members
  - clone the repository
  - work/commit on the local copy
  - push local changes to shared repository
  - pull remote changes from shared repository
- detailed instructions can be found on assignment 0
Merge Conflicts

```
- foo.c rev 1
  Line A
  Line B

- foo.c rev 2
  Line A
  Line B
  Line C

- foo.c rev 2*
  Line A
  Line B
  line C
```

- clone
- clone
- push
- push

?
When Pushing of Changes Fails

- Mercurial error: *abort: push creates new remote heads!*
- What to do?
  - pull remote changes via *hg pull*
  - merge conflicts using *hg merge*

Output of *hg merge*

merging foo.c
warning: conflicts during merge.
merging foo.c failed!
0 files updated, 0 files merged, 0 files removed, 1 files unresolved

```
foo.c
Line A
Line B
<<<<<<<
local
Line C
=======
line C
>>>>>>>
other
```
Frequently Used Mercurial Commands

- **hg init**
  - creates new repository in given directory

- **hg add [files]**
  - schedules files for addition to the repository

- **hg remove [files]**
  - schedules files for removal from the repository

- **hg commit [files]**
  - commits the changes as new HEAD to the repository
  - requests short comment on changes

- **hg update revision**
  - updates the working directory to the specified revision (or tip if no number is given)
Using Multiple Repositories

- `hg clone src dst`
  - creates a copy of repository `src` in directory `dst`
- `hg push dest`
  - pushes changes from local repository to `dest`
- `hg pull src`
  - pulls changes from repository at `src`
Reviewing Your Changes

- `hg diff [files]`
  - show differences between
    - working copy and
    - corresponding repository revision (not tip revision)
Naming Milestones

- `hg tag tagname`
  - gives the current revision a name (tag) with name `tagname`
Some Hints

- commit frequently
  - use tags
  - only commit compiling code
- write meaningful commit messages
  - the first line of a commit message should contain a summary of your changes
  - use additional lines for more precise explanations
  - avoid meaningless messages like “made it work” or “bugfix”
  - there is no need to mention the files you modified, that’s recorded anyway
- feel free to work with multiple, local repositories
- additional documentation is available on the Mercurial website, a good starting point is
  http://www.selenic.com/mercurial/wiki/index.cgi/UnderstandingMercurial
Accessibility

- Mercurial solves many problems
  - record project history
  - distributed development
  - time machine
- but
  - there has to be at least one repository used for “merging” the work of all group members
  - give write access to one repository?
  - not feasible in ATIS
    - no real groups
    - do not make your $HOME group/world writable!
Remote Mercurial Repositories

- set up SSH to allow logins w/o password
- restrict access to Mercurial repositories
- our suggestion
  - one member `s_hghost` creates repositories that are accessible for other group member `s_member`
  - local changes are pushed to “global” repository from time to time
Public Key Authentication

- mkdir -p ~/.ssh
- cd ~/.ssh
- ssh-keygen -t dsa -P "" -f sysarch_s_member
- cat >> config
  Host sysarch_hg
  Hostname i08fs1.ira.uka.de
  User s_hghost
  IdentityFile ~/.ssh/sysarch_s_member

Key Binding

Tell SSH to log in to i08fs1.ira.uka.de as user “s_hghost” when requested to log in to “sysarch_hg”, authenticate via key pair “~/.ssh/sysarch_s_member{.pub}”.

Granting Restricted Access To Your Account

- as user `s_hghost` do
  - `mkdir -p ~/.ssh`
  - `cd ~/.ssh`
  - add to `authorized_keys` lines such as
      - command to no-pty must be one line
      - there must be no newline after “no-pty”
  - append s_member’s public key to `authorized_keys`

Authorizing Keys

Whoever logs in to your account using the pasted key executes “hg-ssh”, pretending to be `s_hghost`. With this configuration, s_member can only invoke the “hg-ssh” script, which allows access to the repositories in ~/sysarch/sharedrepos.
GDB Command Overview

- **gdb myapp**
  - press [Ctrl] + [c] to enter the debugger
- **b** set breakpoint
- **bt, up, down** backtrace, inspect call-stack
- **p sym** print symbol, *sym* may be any C cast
  - **s** execute line of code
  - **c** continue execution
  - **l** list source code at current location
- **q** leave GDB
Debugging OS/161

Vain Attempts

- `gdb kernel`
  - cannot execute MIPS binary on x86

- `cs161-gdb kernel`
  - cannot execute MIPS binary on x86

- `gdb ./sys161 kernel`
- `cs161-gdb ./sys161 kernel`
  - debug System/161?!?

- `./sys161 cs161-gdb kernel`
  - cs161-gdb is not a MIPS binary
Debugging OS/161

Vain Attempts

- **gdb** kernel
  - cannot execute MIPS binary on x86

- **cs161-gdb** kernel
  - cannot execute MIPS binary on x86

- **gdb** ./sys161 kernel
- **cs161-gdb** ./sys161 kernel
  - debug System/161?!!

- ./sys161 cs161-gdb kernel
  - cs161-gdb is not a MIPS binary
Vain Attempts

- `gdb kernel`
  - cannot execute MIPS binary on x86

- `cs161-gdb kernel`
  - cannot execute MIPS binary on x86

- `gdb ./sys161 kernel`
- `cs161-gdb ./sys161 kernel`
  - debug System/161?!

- `./sys161 cs161-gdb kernel`
  - cs161-gdb is not a MIPS binary
Debugging OS/161

Vain Attempts

- gdb kernel
  - cannot execute MIPS binary on x86
- cs161-gdb kernel
  - cannot execute MIPS binary on x86
- gdb ./sys161 kernel
- cs161-gdb ./sys161 kernel
  - debug System/161?!?
- ./sys161 cs161-gdb kernel
  - cs161-gdb is not a MIPS binary
Correct Solution

- ./sys161 -w kernel
  - start System/161
  - immediately wait for debugger to connect
    - or press [Ctrl] + [g] to wait for debugger
- cs161-gdb kernel
  - start debugger (in different shell!)
  - communicate with System/161 via sockets
Configuring cs161-gdb: .gdbinit

- add to `~/sysarch/root/.gdbinit`:
  - target remote unix:.sockets/gdb
    - talk to System/161 via named socket
  - dir `../asst0-src/kern/compile/ASST0`
    - use object files in this directory
    - update this for future assignments!
  - set print array on
  - set print pretty on
    - beautify output
  - b panic
    - break when panic() is called

**Hint**
Strictly speaking, only the first line is required . . .
That’s all, folks!

Thanks for your attention.

Good luck!

… and may the forum be with you …