



# CS 2001

## Department Computing Resources

Wencan Luo  
Borrowed from Marian K. Iskander

<http://people.cs.pitt.edu/~tech/news/faqs.html>

# Agenda

---



- Computing resources in the department
- OpenAFS
- How to:
  - Get AFS tokens (Windows/Linux)
  - Control access to AFS files
  - Manage group access in AFS
  - Manage Subversions (svn) repositories
- Using the *Condor* cluster
  - How to submit batch jobs



# Agenda

---

- Students' webpages – why ??
- From where to get licensed software ?



# Department computing resources



## *The Elements Cluster*

Hostname	Processors	Memory	Architecture	OS Rev	Kernel
antimony	Dual Dual-Core 3.8GHz Xeons	12GB RAM	32-bit Linux	RHEL 4	2.6
arsenic	Dual Dual-Core 3.8GHz Xeons	12GB RAM	64-bit Linux	RHEL 4	2.6
aluminum	Dual Quad-Core 2.33GHz Xeons	16GB RAM	64-bit Linux	RHEL 4	2.6
selenium	Dual Quad-Core 2.33GHz Xeons	16GB RAM	64-bit Linux	CentOS 4.8	2.6
hydrogen	Dual Dual-Core 3.6GHz Xeons	12GB RAM	64-bit Linux	RHEL 4	2.6
oxygen	Dual Dual-Core 3.6GHz Xeons	12GB RAM	64-bit Linux	RHEL 4	2.6
nitrogen	Dual Dual-Core 3.6GHz Xeons	12GB RAM	32-bit Linux	RHEL 3	2.4
rhenium	Dual Hyper-Threaded Quad-Core 2.93GHz Xeons	96GB RAM	64-bit Linux	CentOS 5.5	2.6
nickel	Dual Hyper-Threaded Six-Core 3.33GHz Xeons	96GB RAM	64-bit Linux	CentOS 5.5	2.6
neodymium	Dual Hyper-Threaded Six-Core 3.33GHz Xeons	96GB RAM	64-bit Linux	CentOS 5.5	2.6
neptunium	Dual Hyper-Threaded Six-Core 3.33GHz Xeons	96GB RAM	64-bit Linux	CentOS 5.5	2.6



# Department computing resources

---



## *SPARC/Solaris Systems*

Hostname	Model	Processors	Memory	OS Rev
blitz	Sun Enterprise 450	Dual 300MHz UltraSPARC IIs	512MB RAM	Solaris 9
hydra	Sun Enterprise 4500	Ten 250MHz UltraSPARC IIs	2.5GB RAM	Solaris 10



# Remotely accessing the machines



---

## • Linux

- 1- Open the Linux terminal
- 2- Use Secure Shell (SSH) to connect to the remote machine
  - If you don't want to specify a machine just ssh elements and you will be directed to any available machine in the dept.

```
ssh <username>@elements.cs.pitt.edu
```

- If you want a specific machine

```
ssh <username>@neptunium.cs.pitt.edu
```

- 3- Enter your CS account password

## • Windows

- 1- Install any ssh application (PuTTY, Secure ssh)
  - 2- Specify the machine (or just elements)
  - 3- Enter username and password
- 



# OpenAFS



- AFS (originally *Andrew File System*) is a distributed file and authentication service designed to be scalable to many client computers, using secure authentication and with flexible access control.
- Developed by CMU, commercialized by Transarc, bought by IBM, branched into DFS/DCE, made open source  
[www.openafs.org](http://www.openafs.org)
- Students are highly encouraged to use AFS to store their important files and projects since it is highly reliable, and secure. **Don't** rely solely on your local storage devices to store your important stuff ..



# OpenAFS



- Where does your account reside ?
  - /afs/cs.pitt.edu/usr0/**username**/ (Linux)
  - \\afs\cs.pitt.edu\usr0\**username**\ (Windows)
- How to access your AFS directory using Windows
  - 1- Download the appropriate openAFS client distribution from [www.openafs.org](http://www.openafs.org) and install
  - 2- From your computer explorer window, select -> Map Network drive, select a drive letter, then enter your afs paths as follows : //afs/cs.pitt.edu/usr0/username
  - 3- Using the AFS client authentication, obtain your token to be able to access your directories





# OpenAFS



- How to access your AFS directory using Linux
  - 1- You need to find the exact steps for your Linux distribution online, the process diverges from one Linux distribution to the other ... just Google this !!
  - 2- Once you have configured your afs client. From the Linux terminal you can make use of the following utilities:

Command	why ?
<b>klog</b>	Identify the user to the system (requests password) and generate a token. Usually integrated with the system login procedure.
<b>tokens</b>	Display currently held tokens and their expiration times
<b>fs</b>	System file queries <ul style="list-style-type: none"><li>• fs quota</li><li>• fs listacl</li><li>• fs examine</li></ul>
<b>unlog</b>	explicitly discards one or more tokens. It is usually performed automatically when logging out

# Control access to AFS files (ACLs)



- ACLs are maintained for each directory in your volume and are under your control.

Access Control Rights	Shorthand Notation
<b>r:</b> read	<b>read</b> = rl
<b>l:</b> lookup	<b>write</b> = rlidwk
<b>i:</b> insert	<b>all</b> = rlidwka
<b>a:</b> administer	<b>none</b> = removes entry
<b>d:</b> delete	
<b>w:</b> write	
<b>k:</b> lock	



# Control access to AFS files (ACLs)



- The AFS command used to set and modify ACLs is the "fs" command. To know more about fs use **fs -help**

fs commands	why ?
<code>fs listacl</code>	List ACL for current working directory
<code>fs listacl <i>dir</i></code>	List ACL for the directory <i>dir</i>
<code>fs setacl <i>dir</i> &lt;user&gt; <i>all</i></code>	Give <i>user</i> all rights to <i>dir</i>
<code>fs setacl <i>dir</i> <i>system:anyuser</i> <i>none</i></code>	Revoke all rights to group <i>system:anyuser</i>
<code>fs setacl . john <i>write</i> jane <i>read</i></code> or <code>fs setacl . john <i>rlidwk</i> jane <i>rl</i></code>	Provide user john with write access and user jane with read access to all files in the current working directory



set recursive permissions

```
find ./your_working_dir -type d -exec fs sa -dir \{\} -  
acl <user> all \;
```



# Managing group access in AFS

---



- How to create and manage AFS groups

Note: typing **pts help** will list the various pts commands.

**Create:** pts creategroup *username:<identifier>*

**Add user:** pts adduser jsmith *username:<identifier>*

**Remove user:** pts removeuser jsmith *username:<identifier>*

**Listing group members:** pts membership *username:<identifier>*

**Examine** group membership and change flags

pts examine *username:<identifier>*

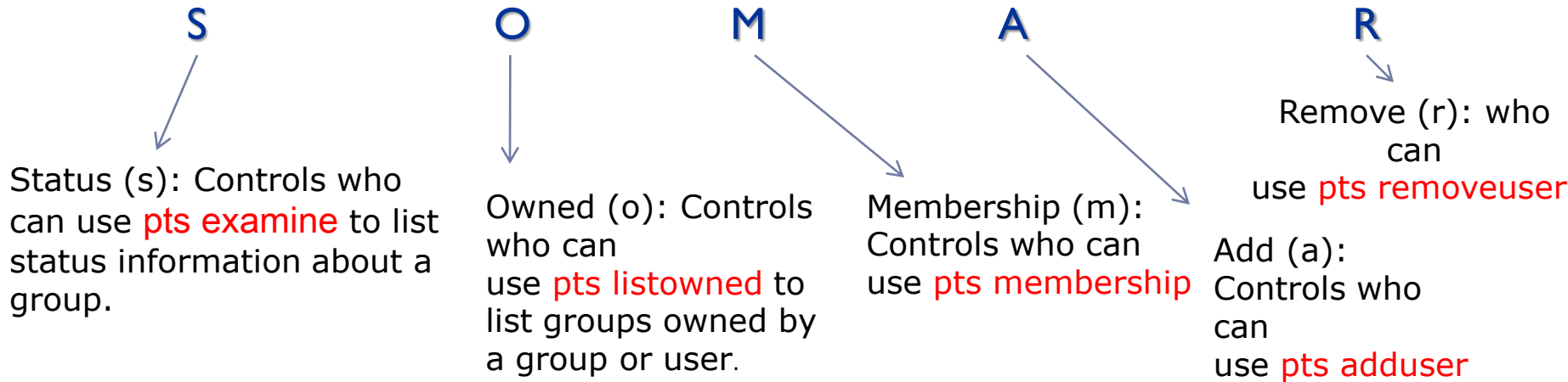
---



# Managing group access in AFS



- Flags



**Each one of the flags, somar , has three possible values:**

- A hyphen, "-", gives rights only to the group's owner
- A lowercase version of the flag (eg a lowercase "s") gives rights to members of the group, in addition to those who have "hyphen" rights.
- An uppercase version of the flag gives rights to anyone.



# Private Directories and DropBox



- **How to make an AFS directory private ?**

- To make an AFS directory so that only you can read & administer the contents, you should remove all entries ACLs except one for you. A quick way to do this is with the command:

```
fs setacl <directory> your-username all -clear
```

For example:

```
fs setacl /afs/cs.pitt.edu/user0/wencan/private wencan all -clear
```

- **How to make a "drop box" directory ?**

- A "drop box" directory is a directory that people can copy files to but not read, delete, or write to files that are already in the directory. **Good way for submitting assignments.**
- To create such a directory, create a private directory and then add "**il**" only rights for the pts group or user that should have rights to create files in that directory.



# Subversion



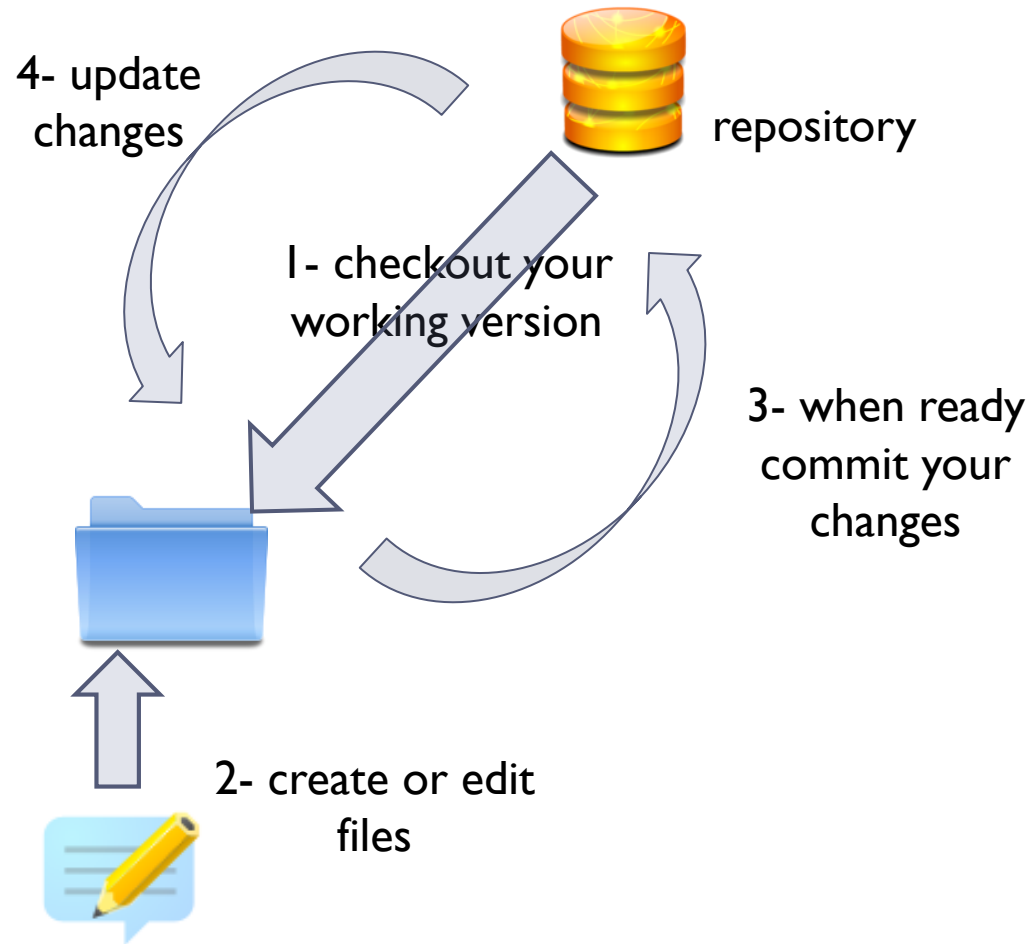
- CollabNet, Inc, developed Subversion starting in 2000 as part of a collaboration software suite.
- SVN tracks changes to files and also directories and directory hierarchies.
- Uses a client-server model.
  - local access to Subversion Repository
  - `svn://` protocol for client-server operation (can be SSH tunneled)
  - `http(s)://` protocol, layered on top of WebDAV (uses Apache modules)
- All data are stored in a server database repository



# Subversion



- The essential **Subversion lifecycle** is the following:





# Subversion



- **Creating** svn repository (in your directory):

```
svnadmin create <svn_identifier>
```

- **NOTE :** Subversion uses this directory to store information about your projects, like file revisions.
- You won't need to directly deal with this directory, so I suggest keeping it in a *safe place* and not messing with its contents unless you know what you're doing 😊



# Subversion



- **Checkout** your working copy

```
svn checkout  
file:///afs/cs.pitt.edu/usr0/<username>/<svn_identifer>
```

- **Create** a directory for your project in your repository to import your project files to later

```
svn mkdir  
file:///afs/cs.pitt.edu/usr0/<username>/<svn_identifer>/<myproj>
```

- **Importing**

```
cd <your_project_dir>  
svn import  
file:///afs/cs.pitt.edu/usr0/<username>/<svn_identifer>/<myproj>
```



# Subversion



- **Committing** your changes

while in your working directory:

```
svn commit -m "some_message"
```

- **Working with revisions**

Check status:

```
svn status <file_name>
```

- U:** File was updated.
- A:** File was added.
- D:** File was deleted.
- R:** File was replaced.
- G:** File was merged.
- ?:** Resource is not under version control.
- !:** Resource is missing or incomplete (removed by another tool than Subversion).



# Subversion



- **Adding file**

while in your working directory:

```
svn add <file/dir_name>
```

Then you need to commit the added file/dir

- **Finding differences**

```
svn diff
```

```
svn diff <file/dir_name>
```

- **Comparing revisions**

```
svn compare -r R1:R2 <filename>
```

- **Revert local edits**

```
svn revert <filename>
```

- **Revert to Previous Revisions**

```
svn update -r R
```



# Another option: GIT

---

- ▶ Decentralized and Faster
- ▶ Two web-based hosting service
  - ▶ <https://github.com>
  - ▶ <https://bitbucket.org/>
- ▶ GIT cheat sheet
  - ▶ <https://training.github.com/kit/downloads/github-git-cheat-sheet.pdf>



# The *Maté* Cluster

---



- The department operates a cluster of 16 SunFire X2100 compute servers, each with dual 2.4GHz 64-bit AMD Opteron processors and 4GB RAM.
- This cluster runs the **Condor** distributed processing manager on Red Hat Enterprise Linux 4.
- The control machine for this cluster is  
[s1.mate.cs.pitt.edu](http://s1.mate.cs.pitt.edu)
- For a complete reference guide visit  
<http://www.cs.wisc.edu/condor/manual/v7.4/>



# The *Maté* Cluster



- 1- To use the cluster, you need to give the condor cluster read/write access rights to your working directory (recursively).

```
find ./your_working_directory -type d -exec fs sa -dir {} -acl condor rlidwk \;
```

- 2- Create your jobs file (*filename.sub*)

- 3- Connect to [s1.mate.cs.pitt.edu](http://s1.mate.cs.pitt.edu)

- 4- **Submit** jobs

```
condor_submit <filename.sub>
```



# The *Maté* Cluster

---



5- **Check** jobs status

```
condor_status
```

6- **Monitor** your job(s)

```
condor_q
```

7- **Kill** your job(s)

```
condor_rm xxx
```

where xxx is your job ID that will be displayed to you when you issue the command in (6)

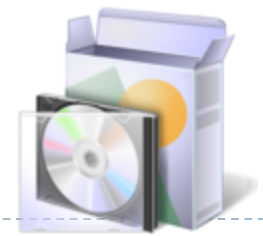
---





# Software licenses

---



- Go to <http://software.pitt.edu/>

Log in with your pitt account, and you will find a selection of licensed software titles at no cost 😊

- Free student software licenses and low cost software can be obtained from the university (windows, office, Matlab.... ). Visit

<http://technology.pitt.edu/software.html>

For more details on the available software and where to get them from.

---



# Your homepage

---



- Why ?
    - It is your front image in the department and worldwide
    - You need to show your academic progress and publications .. right?
    - If your teaching, then this is a good place to put recitations/lab material for your students to download
  - Where to put your files ?
    - All your html pages, scripts .. etc should go under your afs space in public/html directory.
    - Make sure if you add downloadable material to put it in the public/html to be accessible from your homepage.
- 



---

# Thank you



For any technical questions or to report any equipment issues in the department  
Email [tech@cs.pitt.edu](mailto:tech@cs.pitt.edu) or file a ticket at <https://ticket.cs.pitt.edu/>

