Technology

- Java SE JDK (version 6 or newer)
- Eclipse
- SVN
  - SVN plugin for Eclipse
- PostgreSQL JDBC driver
- C3P0 connection pooling library
- SSH Tunneling
Java

• Java SE JDK (version 6 or newer)
    • *Be careful, the installer sometimes tries to install the Ask toolbar, which you likely do NOT want*
  – Java is already installed on the lab machines
Java

• Java works well as a back-end server-side language
• Browser plugins have significant security issues
  – *Be sure to disable the Java browser plugins*
• Desktop use never caught on
Eclipse

• Standard version preinstalled on the lab machines

• If you are installing on your own machine then get the Eclipse IDE for Java EE Developers version
Eclipse

• Allows you to configure multiple workspaces
  – Suggest using separate workspace for CSE 530A to avoid naming conflicts with other classes

• Multiple projects within a workspace
  – Separate programs should be organized as separate projects
  – Each lab and studio will be a separate project
Eclipse

- Different "perspectives" for different purposes
  - Java
    - for working on Java projects
  - Java EE (Enterprise Edition)
    - adds additional tools for web development and database access (and others)
  - SVN Repository Exploring
    - for browsing SVN repositories and checking out projects
    - should only be used for initial checkouts
  - Team Synchronizing
    - for comparing, updating, and committing changes to repositories
SVN

• Subversion is a software versioning and revision control system

• *Centralized* system
  – Means there is a master repository
  – Newer systems such as git and mercurial are distributed

• Typical workflow
  1. Update working copy with changes from repository
  2. Make changes to working copy
  3. Commit changes in working copy back to repository
SVN

- **Do not check out entire repository into workspace**
  - Eclipse will treat it as a single project and not know what to do with it

- Check out each project individually
  - studio-2 and lab-1 (and future studios and labs) are configured as individual Eclipse projects
SSH

• Secure Shell
  – allows execution of network services over a secure channel
    • remote shell, file copy, port forwarding and tunneling
  – standard client/server architecture
    • both client and server included on almost all OSes (except Windows)
Tunneling

• From outside the CEC network, the postgres machine is unreachable
Tunneling

• ssh allows us to forward connections to a local port through shell to postgres
Tunneling

• Internal machines can also tunnel through shell – (Why would we want to do this?)
Tunneling

• Mac OS and Linux
  – "ssh -L 5432:postgres:5432 user@shell.cec.wustl.edu"

Diagram:
- Forward local port
- Local port 5432
- Target port 5432
- Target of tunnel (as viewed from ssh server)
Tunneling

• Windows
  – Need third-party client
    • PuTTY is my favorite
Tunneling

• Note that if local port 5432 is already in use then setting up a tunnel using local port 5432 will *quietly fail*
  – In other words, if you are running PostgreSQL locally then binding an outgoing ssh tunnel to port 5432 will not work