

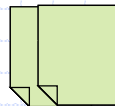
# Using Subversion:

Nirav Dave  
Computer Science & Artificial Intelligence Lab  
Massachusetts Institute of Technology

1

## John Q. Hacker is working on a large project

- ◆ Add some stuff
- ◆ Changes some stuff
- ◆ And some more stuff
- ◆ Realizes that he wanted to look at something in the blue version
- ◆ But... he's already changed the file. The data is gone.



John should have made a history of snapshots

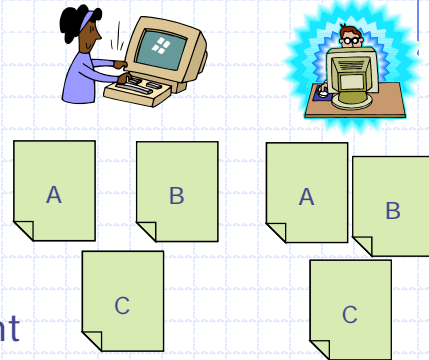
June 3, 2008

2

## Alice and Bob are working together

- ◆ Split up tasks

- Alice: widget A
- Bob: widget B



- ◆ What's the newest version?

- ◆ Do we have coherent understanding of C?

June 3, 2008

3

## Okay, so what do we want?

- ◆ Some central place (a repository) to store the newest version of our design

- ◆ A record of the history to see:

- When bugs were introduced (and who added them)
- What did we have before (work out John's problem)

- ◆ A way of resolving conflicts

- Make it easier for us to get a sane widget C

- ◆ Almost a Distributed Memory w/ Cache coherences

- Makes explicit when things are "final/stable" and can be seen by others

June 3, 2008

4

## Enter Subversion

- ◆ A version control system
  - Well suited for programming/text files
- ◆ Like CVS (but with global tags)
- ◆ A fairly good choice for small projects
  - A little awkward for long-term branches

June 3, 2008

5

## A day in the life of a subversion checkout

- ◆ Start: A version of the repository

**svn update**

This synchronizes your local copy  
with the global repository

June 3, 2008

6

## A day in the life of a subversion checkout

- ◆ Changes are made
  - Just like normal
- ◆ New files are added to the system

```
svn add <filename>
```

Now your local repository has the changes you wanted to have

June 3, 2008

7

## Verifying your changes

- ◆ It's good to review what changes you've made:

```
svn status
```

- ◆ Gives you a list of files that are modified (M), added (A), etc.

June 3, 2008

8

## A day in the life of a subversion checkout

- ◆ Attempt to Commit, leaving a helpful message for later use

```
svn commit
```

- ◆ Two Cases:
  - The repository sees noone has changed it since you got a copy of it's world.
  - Otherwise, it asks that you get an updated version of the repository.

## Dealing with Merge

- ◆ Merging changes can be tricky.
  - Sometimes it leaves the file in a conflicted state (a unified diff of the two version)
  - User must manually fix the error and explicitly say the file is fixed using

```
svn resolved <filename>
```

## More information

Tutorial:

`http://artis.imag.fr/~Xavier.Decoret/  
resources/svn/index.html`