Lab: Version Control System and SVN

NetDB

CS, NTHU,

Fall, 2012

Outline

- Why do we need version control?
- Introduction to SVN
- Using SVN Client in Eclipse
- Exercise and Homework Submission

Outline

- Why do we need version control?
- Introduction to SVN
- Using SVN Client in Eclipse
- Exercise and Homework Submission

Why Do We Need Version Control?

- A place to store the projects
- Synchronization between modifications made by different developers
- Even when you are developing a project along, SVN still helps in showing your revision history

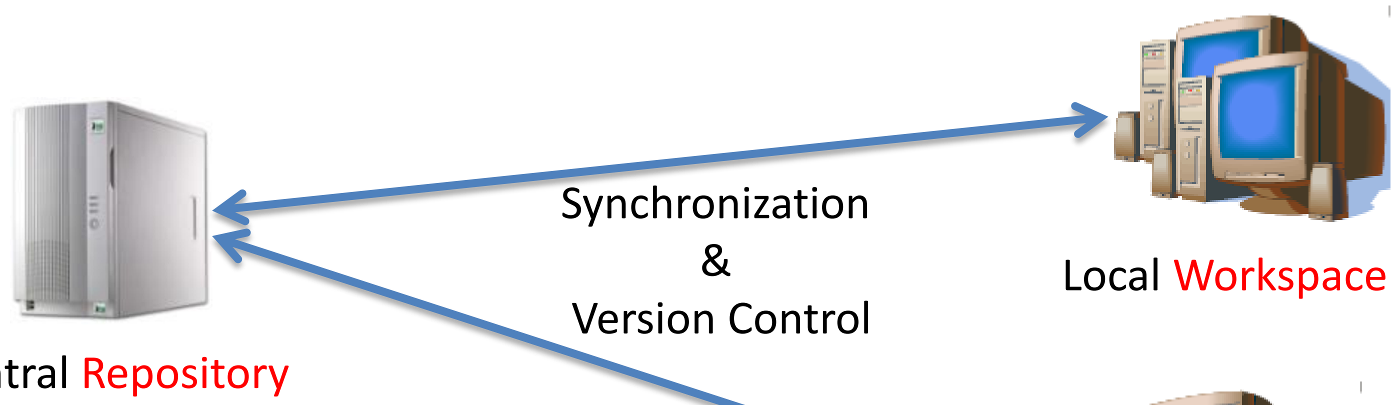
Outline

- Why do we need version control?
- Introduction to SVN
- Using SVN Client in Eclipse
- Exercise and Homework Submission

Introduction to SVN

- “Subversion” in short
- An open-source version control system
  – To record every changes made to files (data/code) and directories
    - Who made a change?
    - What has been changed?
    - When did they make it?
    - Why did they make it?
  – To allow the recovery of older versions
  – To trace the history of how your data/code changed
  – To collaboratively edit and share data/code

Repository and Workspace



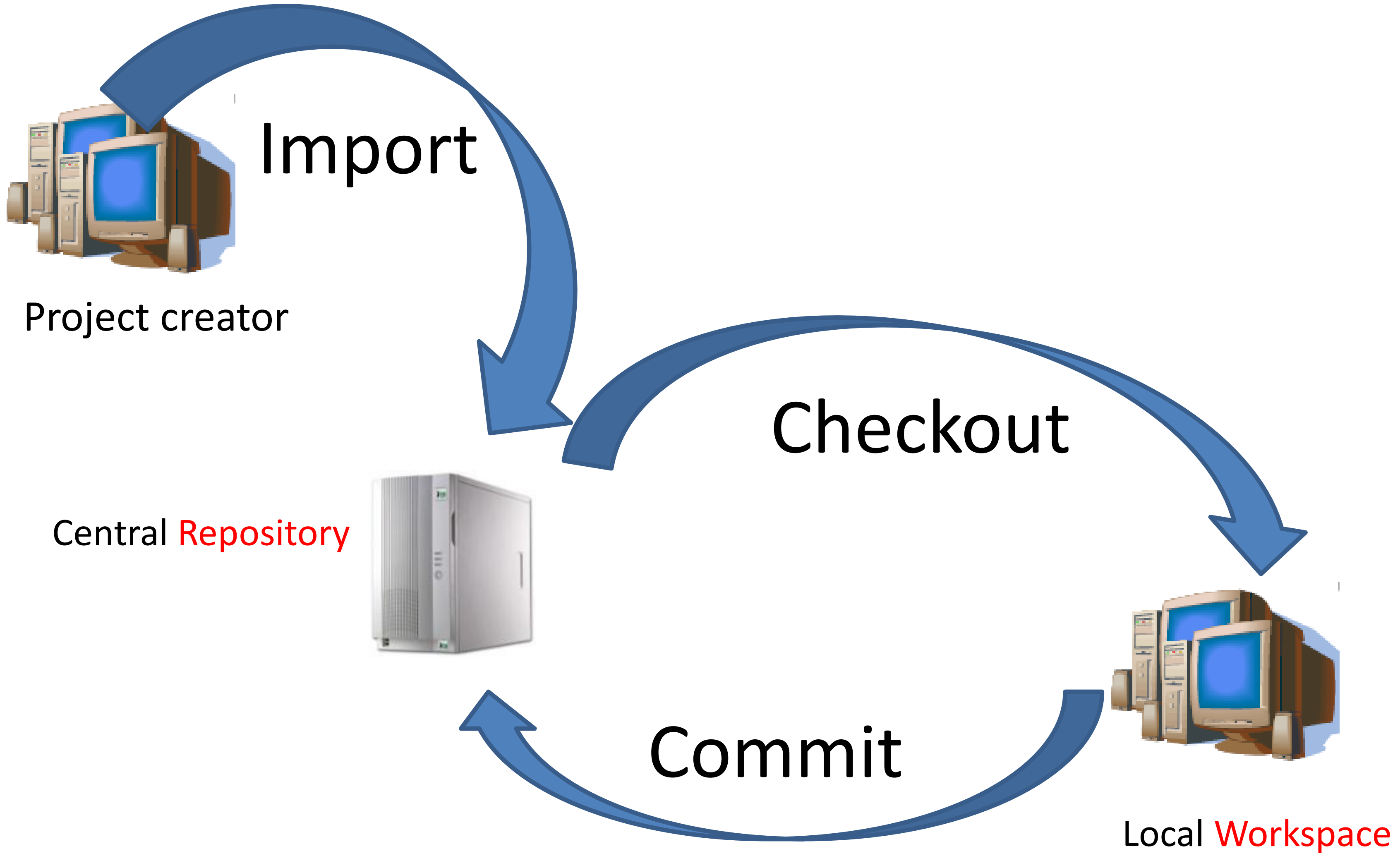
Store data/code and all change history in the central repository.

Load data/code, and develop and fix them at local workspace.

Repository

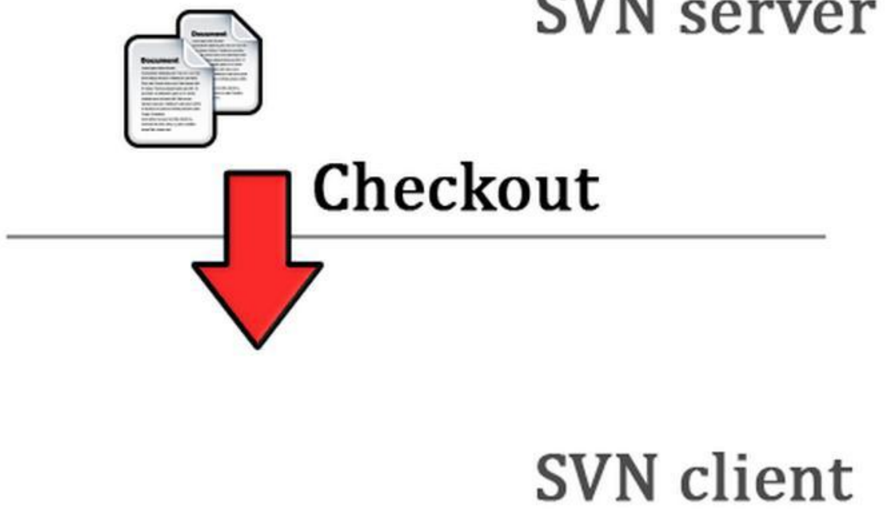
- SVN repository can be built on different protocols (e.g. svn, http, https)
- Each directory or file has a unique URL
  - **E.g.** https://netdb.cs.nthu.edu.tw/svn/SoftwareStudio_prj/
- Some SVN clients to access the repository:
  - Web browser
  - TortoiseSVN
  - Subclipse

How to Interact with the Repository?



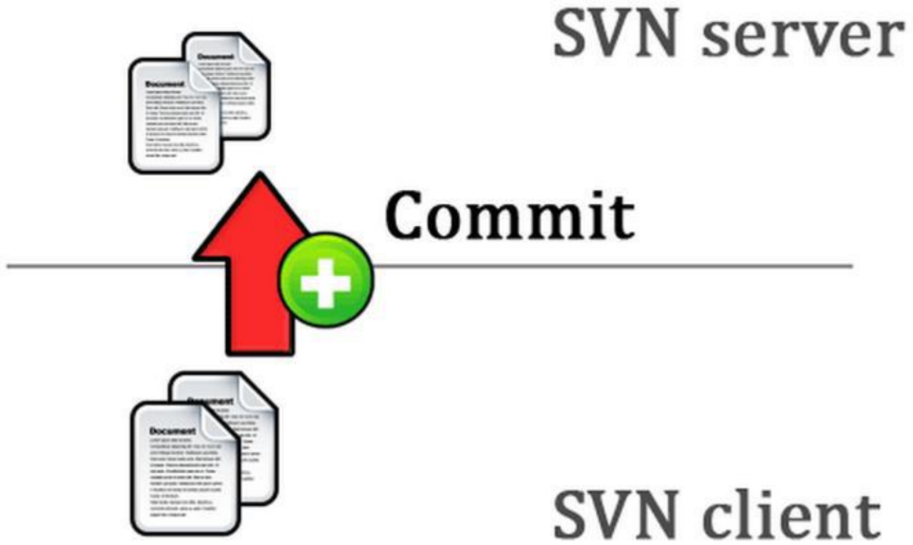
Checkout

- Checkout repository to create a working copy in local workspace
- Checked out files appear to be a directory with program's source



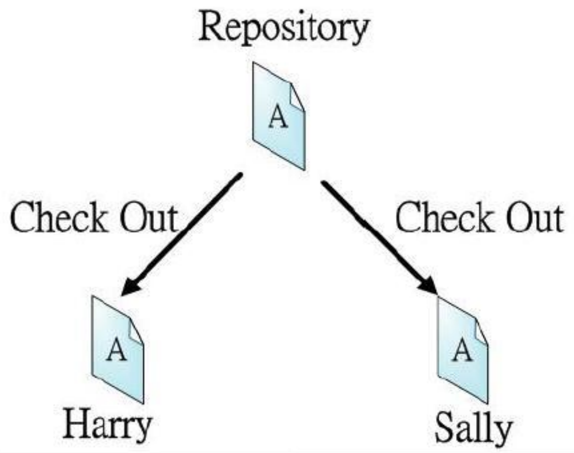
Commit

- Find changes made since checkout and commits them into the repository
- Creates a new revision number in the repository

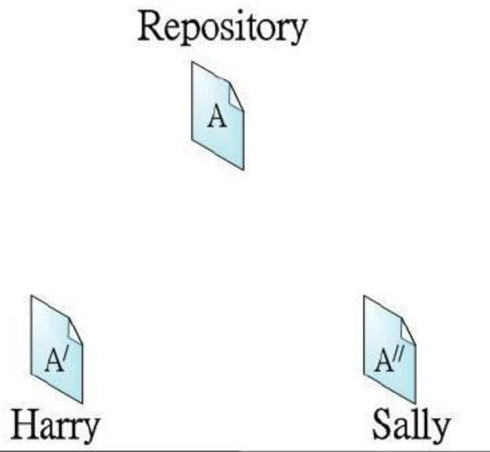


Conflict and Merge

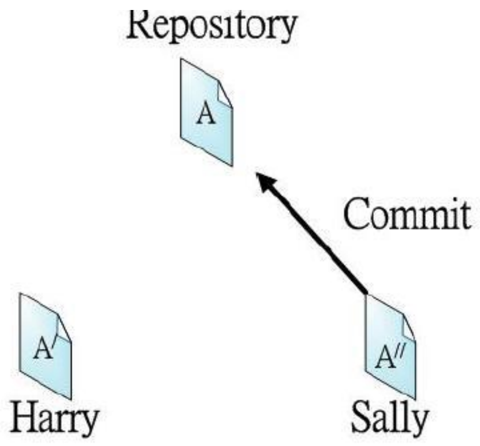
1. Checkout at the same time



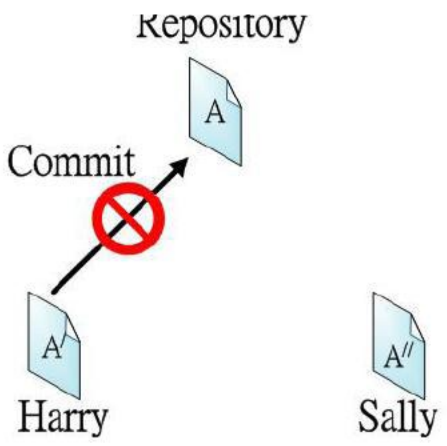
2. Each modifies his/her file



3. Sally commit first



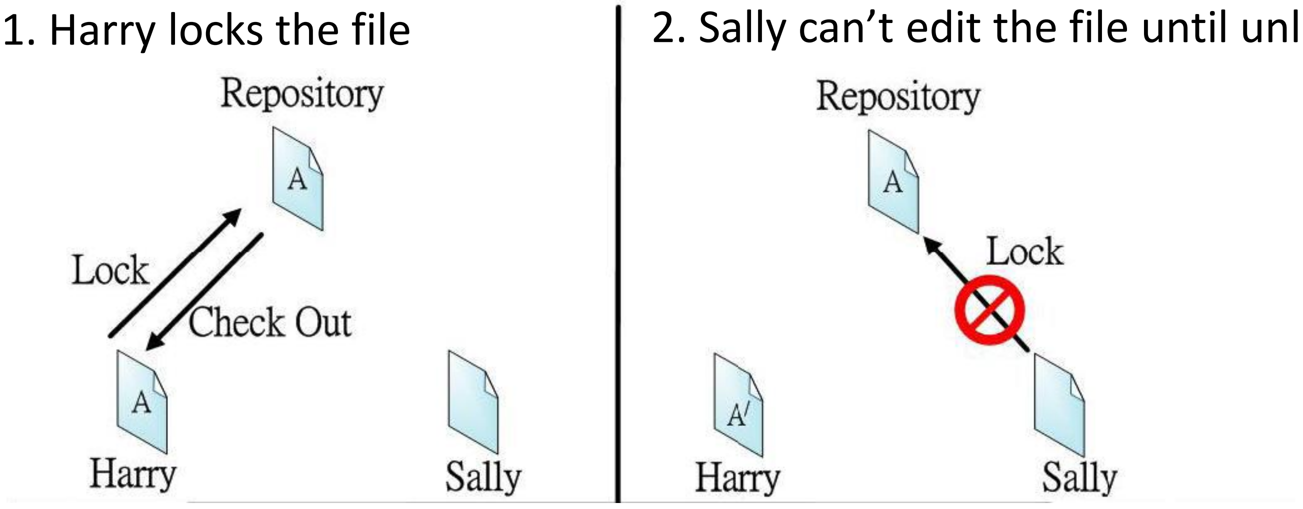
4. Out-Of-Date when Harry commit



Conflict and Merge (Sol.1)

1. Harry locks the file

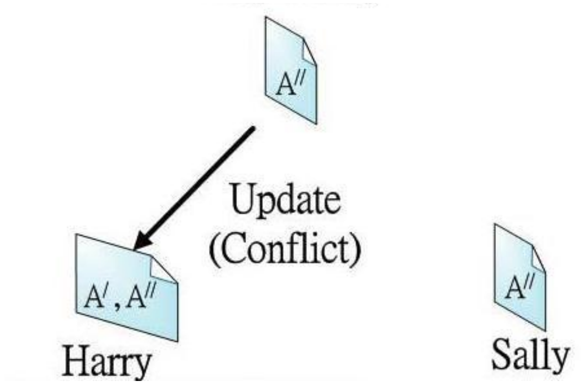
2. Sally can't edit the file until unlocking



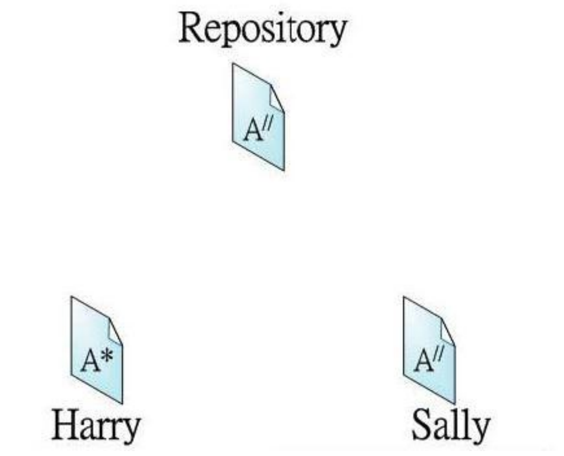
Conflict and Merge (Sol.2)

- This is the way SVN does

1. Harry commits and finds the file conflicting



2. Harry merges the differences between repository-file and his file and then re-commits



Outline

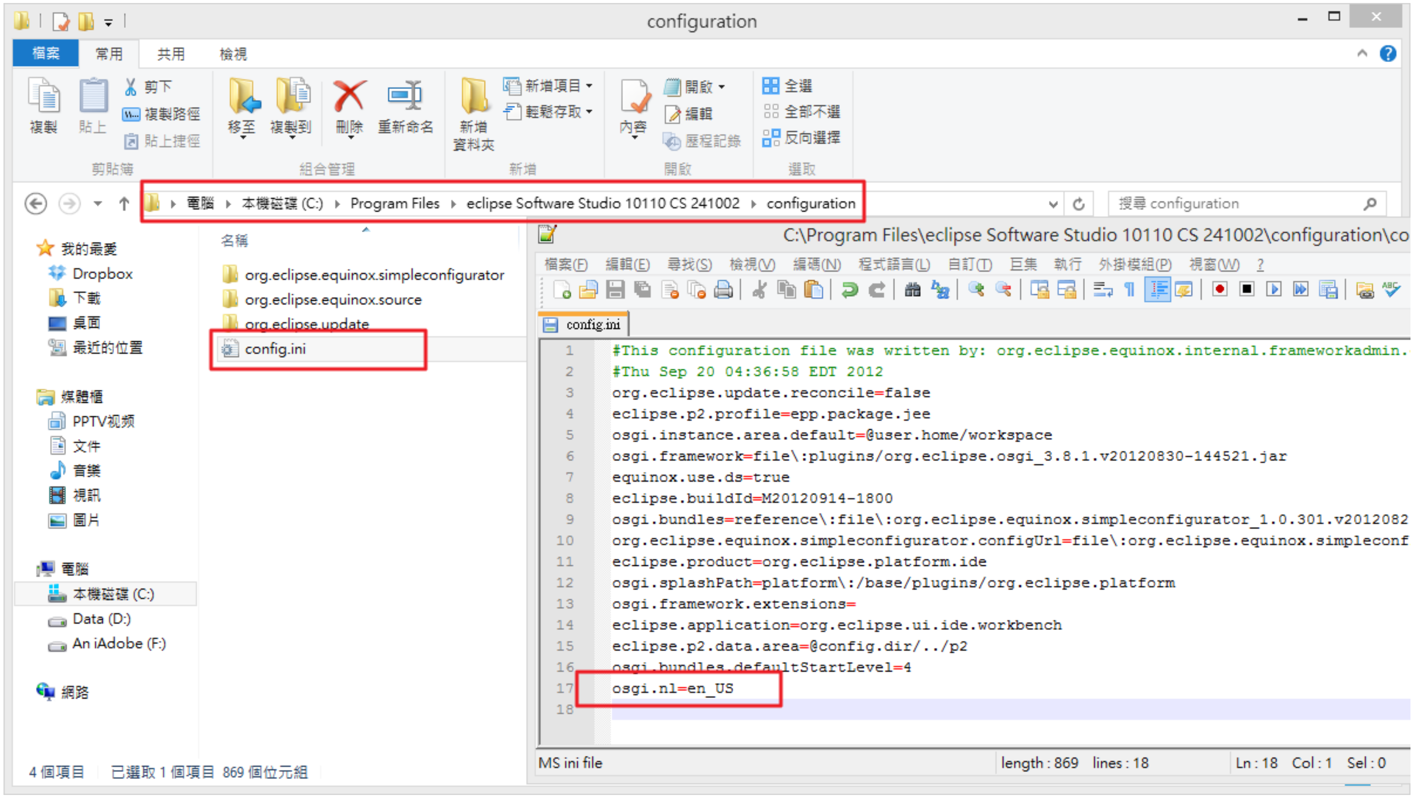
- Why do we need version control?
- Introduction to SVN
- Using SVN Client in Eclipse
- Exercise and Homework Submission

Using SVN in Eclipse

- Installing the Subclipse plugin
  - A built-in SVN client in eclipse
  - See Appendices B on the course website
- Subclipse operations in eclipse
  - “Checkout” a project from repository
  - “Commit” your code from eclipse
  - “Import” project to repository

Set Your Subclipse GUI to English

- Add a line "osgi.nl=en_US" into eclipse/configuration/config.ini



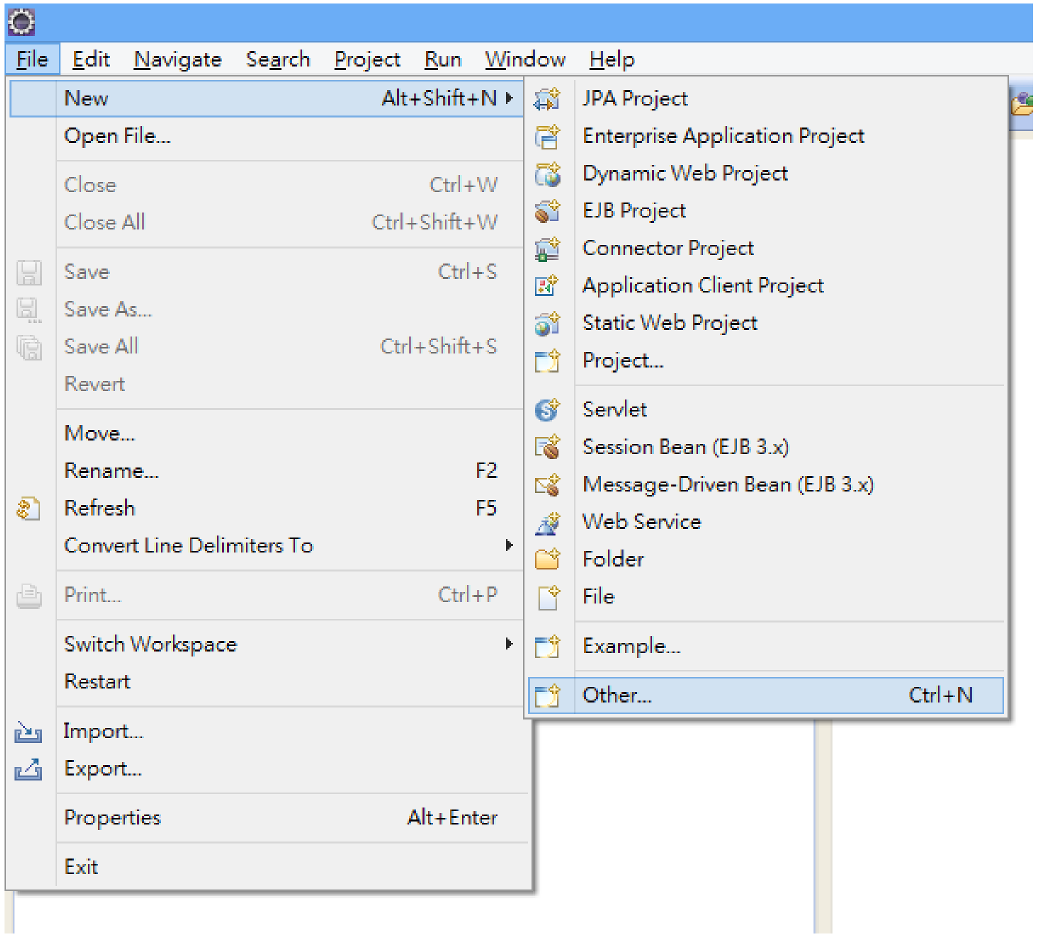
```
#This configuration file was written by: org.eclipse.equinox.internal.frameworkadmin.
#Thu Sep 20 04:36:58 EDT 2012
org.eclipse.update.reconcile=false
eclipse.p2.profile=epp.package.jee
osgi.instance.area.default=@user.home/workspace
osgi.framework=file\:plugins/org.eclipse.osgi_3.8.1.v20120830-144521.jar
equinox.use.ds=true
eclipse.buildId=M20120914-1800
osgi.bundles=reference\:file\:org.eclipse.equinox.simpleconfigurator_1.0.301.v2012082
org.eclipse.equinox.simpleconfigurator.configUrl=file\:org.eclipse.equinox.simpleconf
eclipse.product=org.eclipse.platform.ide
osgi.splashPath=platform\:/base/plugins/org.eclipse.platform
osgi.framework.extensions=
eclipse.application=org.eclipse.ui.ide.workbench
eclipse.p2.data.area=@config.dir/../p2
osgi.bundles.defaultStartLevel=4
osgi.nl=en_US

```

Authentication Required

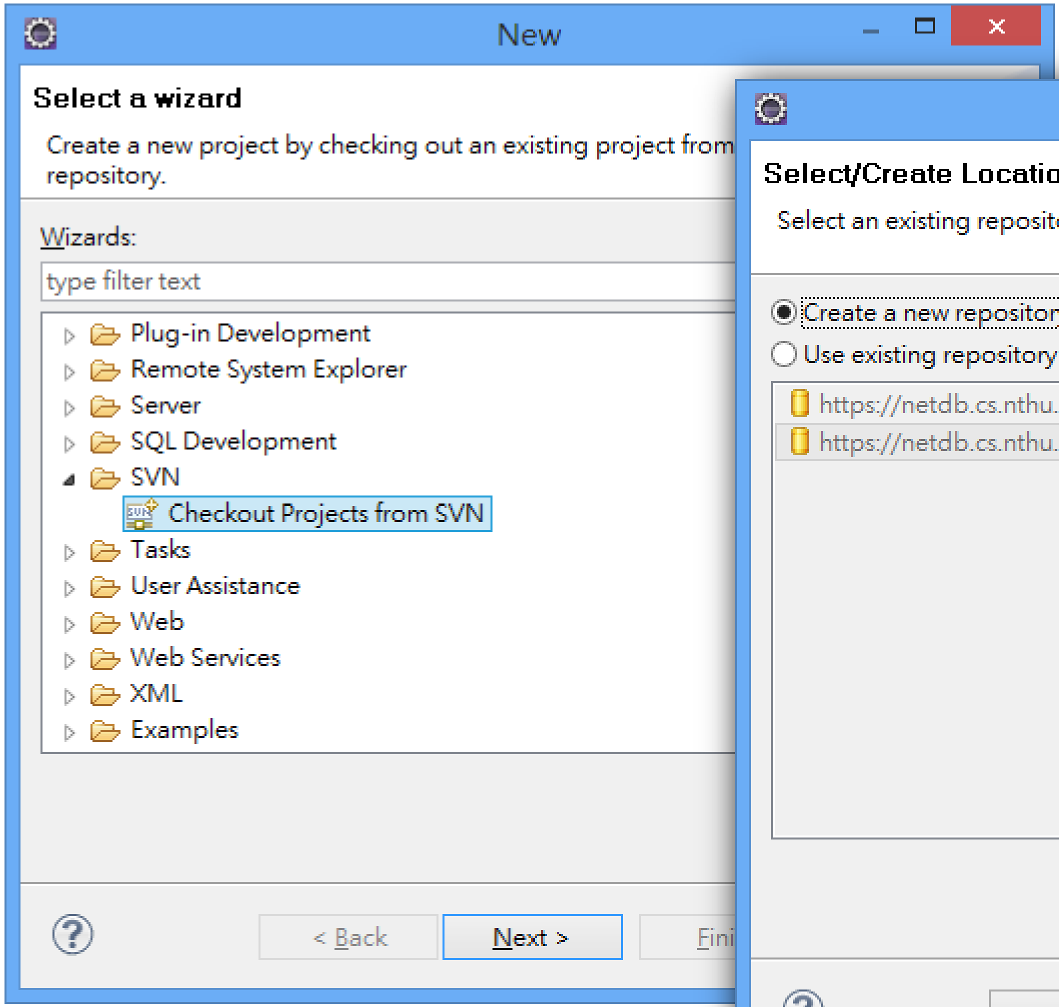
- Account:
  - Your student ID
- Password:
  - Your student ID by default
  - Change it using this page: https://netdb.cs.nthu.edu.tw/svntools/passwd/index.php

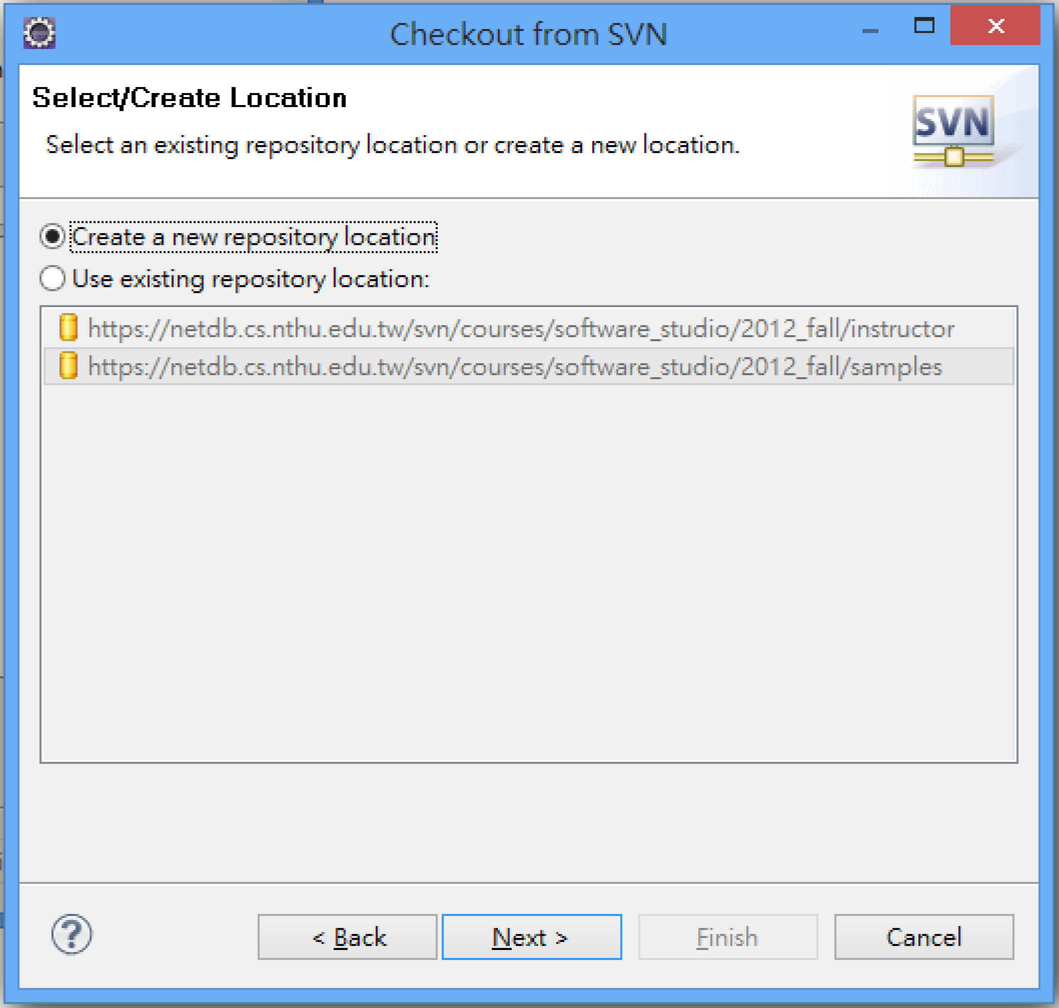
Checkout Project from Repository

- File -> New -> Other



Checkout Project from Repository

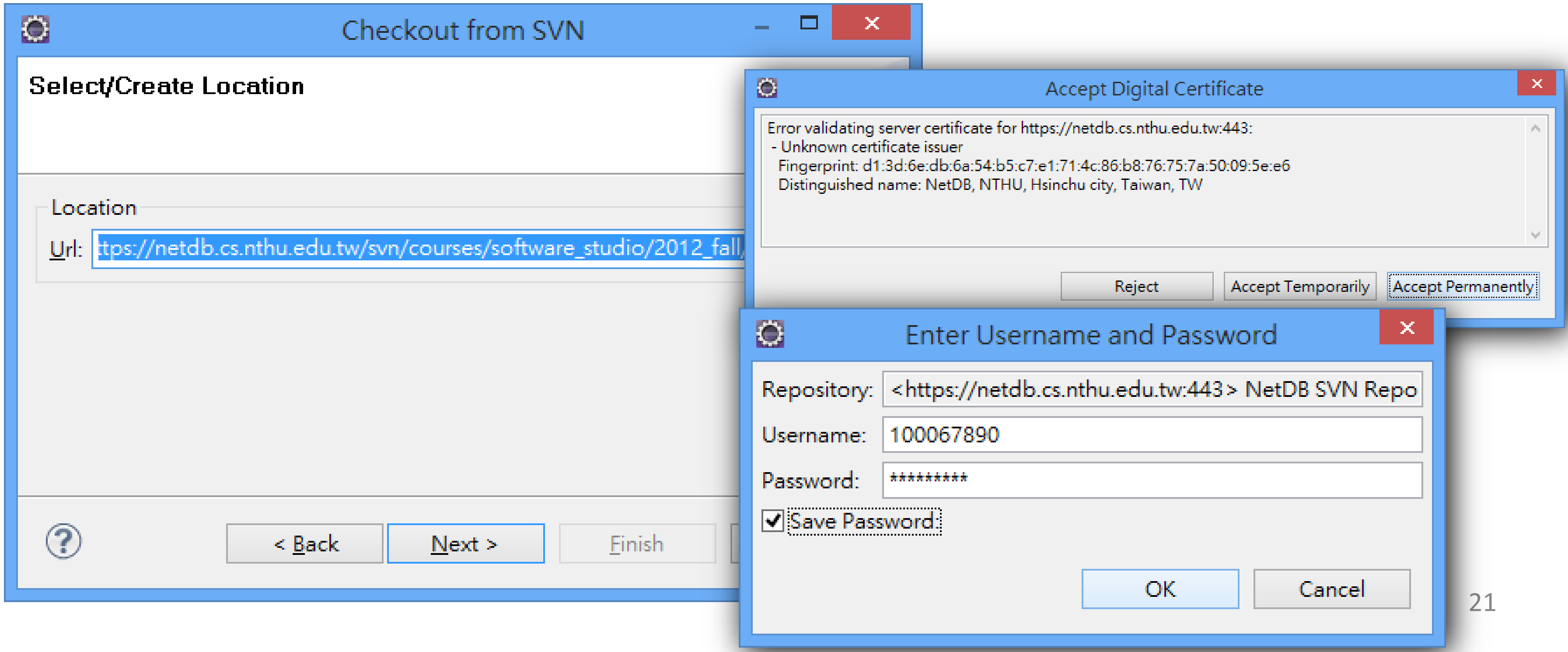



Checkout from SVN
Select/Create Location
Select an existing repository location or create a new location.
SVN
Create a new repository location
Use existing repository location:
< Back
Next >
Finish
Cancel

https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/instructor
https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/samples

Checkout Project from Repository

- SVN Repository URL for sample projects https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/samples/


Checkout from SVN

Select/Create Location

Location

Url: ttps://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/

Accept Digital Certificate

```
Error validating server certificate for https://netdb.cs.nthu.edu.tw:443:
 - Unknown certificate issuer
   Fingerprint: d1:3d:6e:db:6a:54:b5:c7:e1:71:4c:86:b8:76:75:7a:50:09:5e:e6
   Distinguished name: NetDB, NTHU, Hsinchu city, Taiwan, TW
```

Enter Username and Password

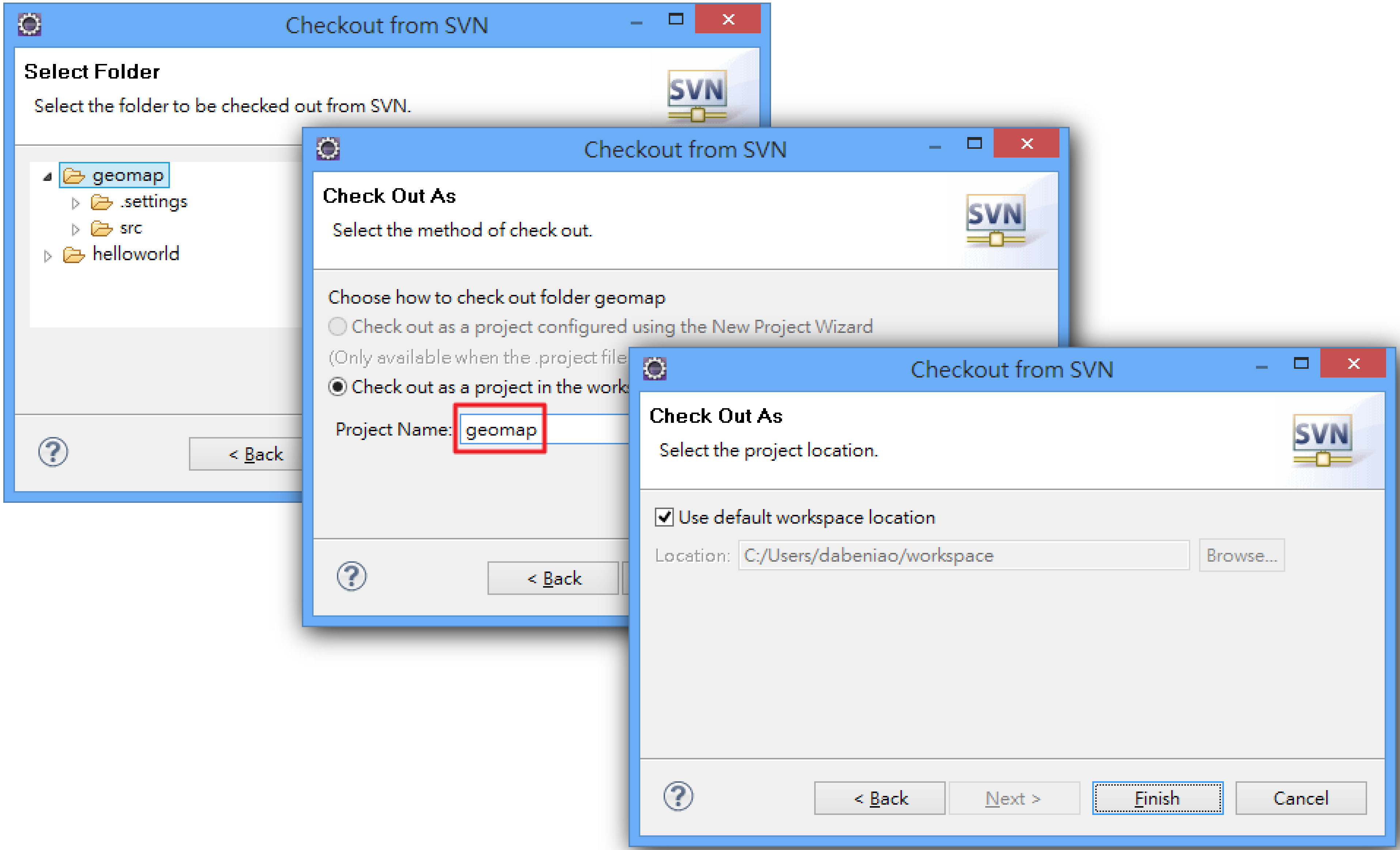
Repository: <https://netdb.cs.nthu.edu.tw:443> NetDB SVN Repo

Username: 100067890

Password: *********

[✓] Save Password

Checkout Project from Repository


Checkout from SVN

Select Folder

Select the folder to be checked out from SVN.

- geomap
  - .settings
  - src
- helloworld

Checkout from SVN

Check Out As

Select the method of check out.

Choose how to check out folder geomap

- [ ] Check out as a project configured using the New Project Wizard
- (Only available when the .project file
- [✓] Check out as a project in the works

Project Name: geomap

Checkout from SVN

Check Out As

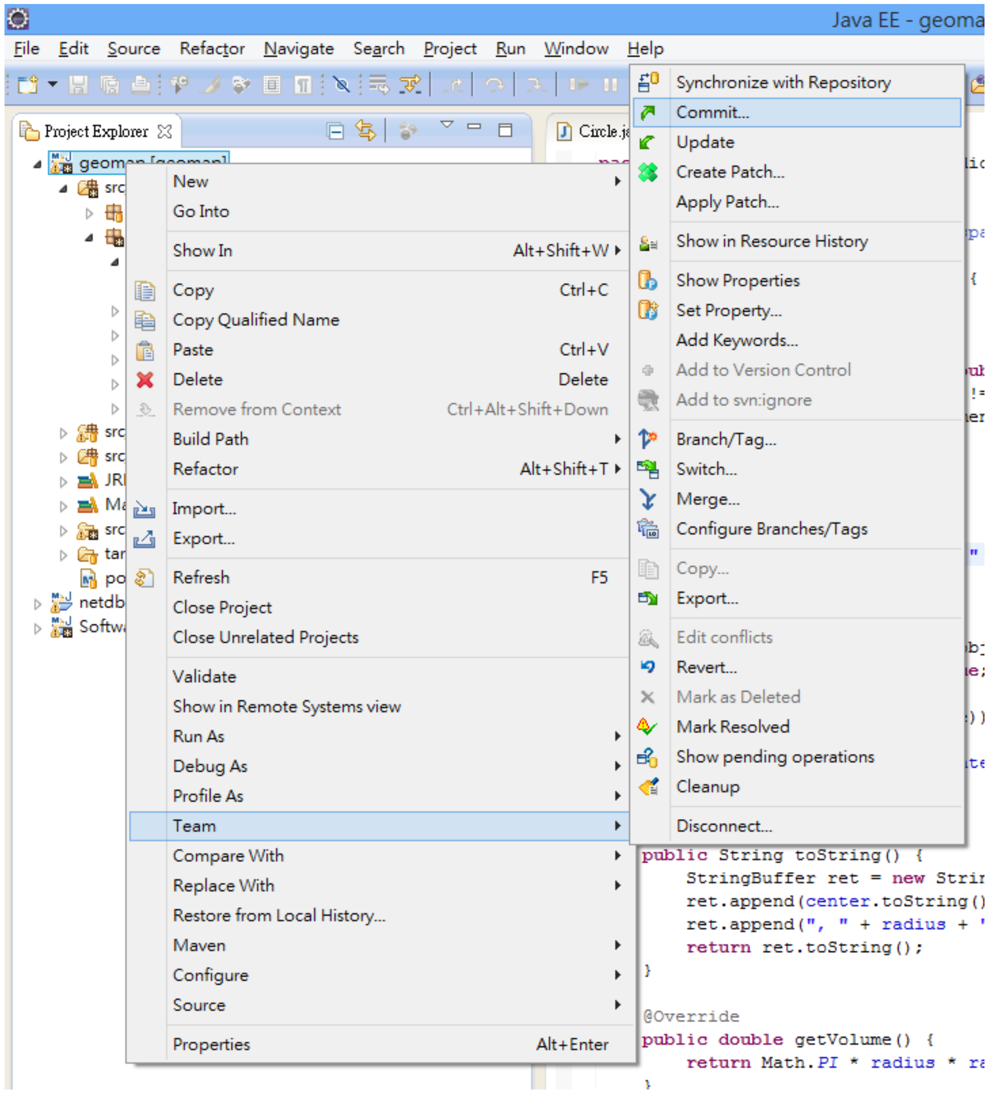
Select the project location.

- [✓] Use default workspace location

Location: C:/Users/dabeniao/workspace

Commit Changes to Repository

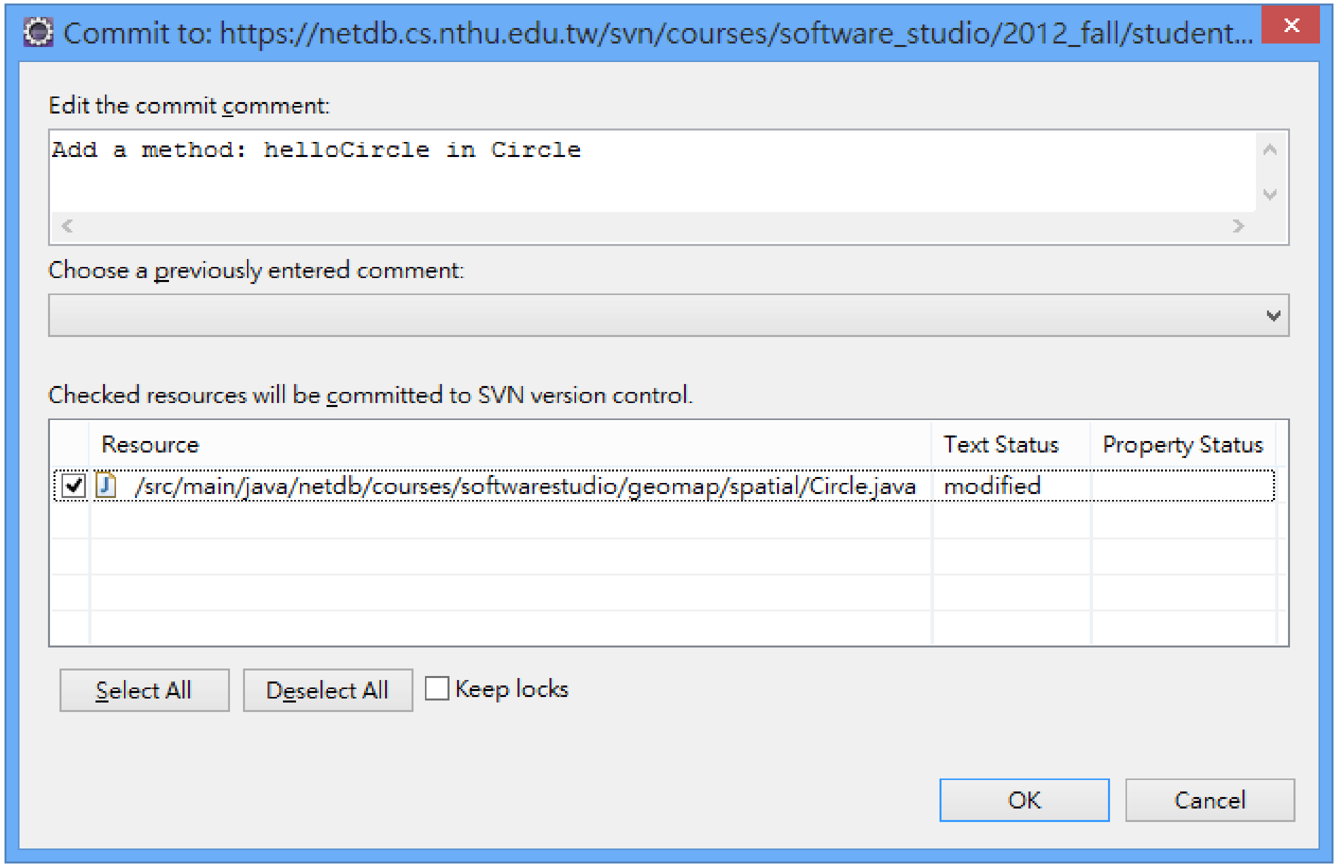
- Checked-out projects are connected to SVN
- Changes can be committed back
  - Right click -> Team -> Commit
- You can commit either files or directories



```
public String toString() {
    StringBuffer ret = new Strin
    ret.append(center.toString()
    ret.append(", " + radius + '
    return ret.toString();
}

@Override
public double getVolume() {
    return Math.PI * radius * ra
}
```

Commit Changes to Repository

- It is a good practice to always comment on the changes you made



Commit to: https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/student...

Edit the commit comment:

```
Add a method: helloCircle in Circle
```

Choose a previously entered comment:

Checked resources will be committed to SVN version control.

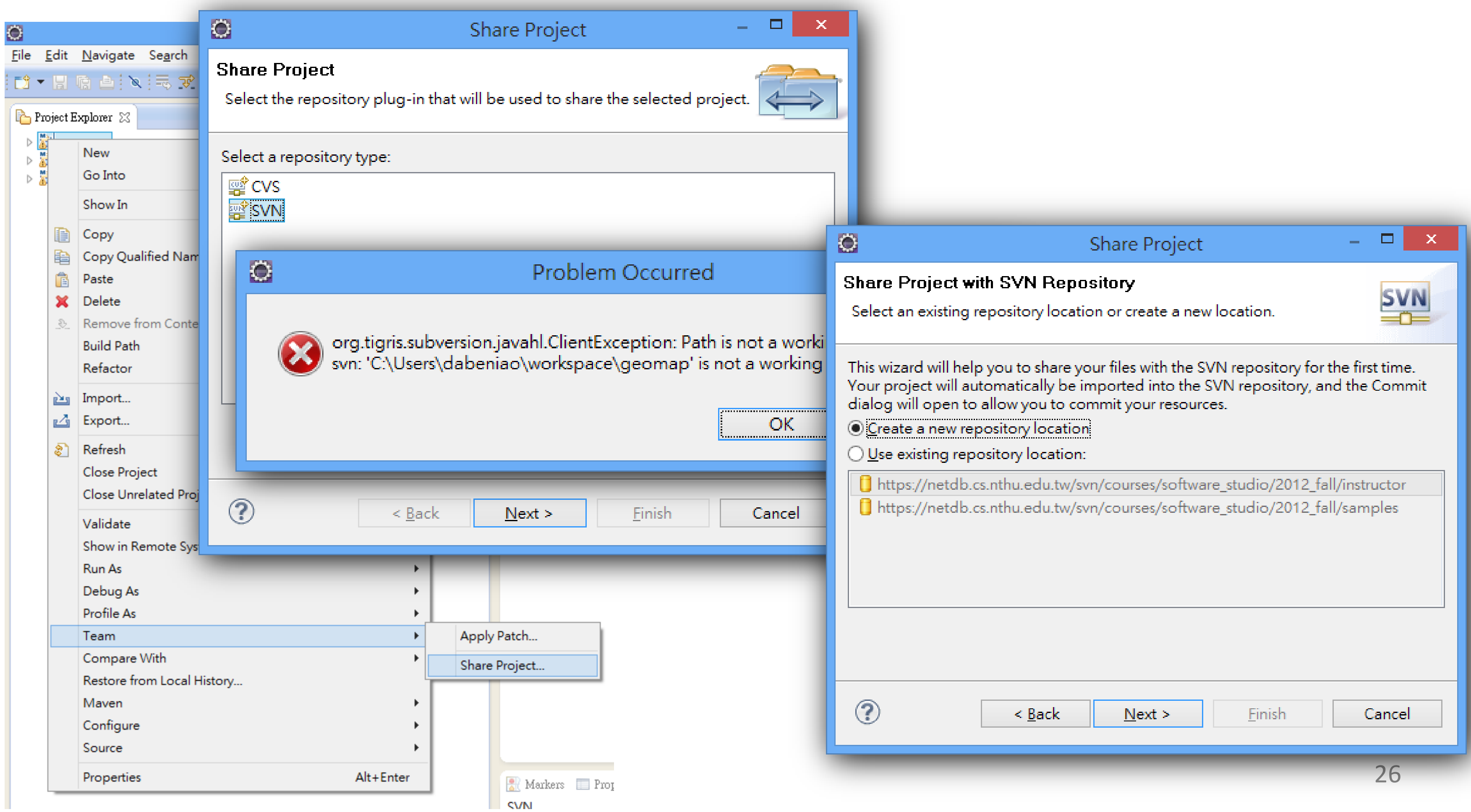
	Resource	Text Status	Property Status
✓	/src/main/java/netdb/courses/softwarestudio/geomap/spatial/Circle.java	modified	

Import Project to Repository

- How to create projects on the SVN server?
  - Create a local project (e.g., using Maven as before)
  - "Import" it into SVN

Import Project to Repository

- Right click the project -> Team -> Share Project



Share Project

Share Project

Select the repository plug-in that will be used to share the selected project.

Select a repository type:

CVS

SVN

Problem Occurred

org.tigris.subversion.javahl.ClientException: Path is not a worki
svn: 'C:\Users\dabeniao\workspace\geomap' is not a working

Share Project

Share Project with SVN Repository

Select an existing repository location or create a new location.

This wizard will help you to share your files with the SVN repository for the first time. Your project will automatically be imported into the SVN repository, and the Commit dialog will open to allow you to commit your resources.

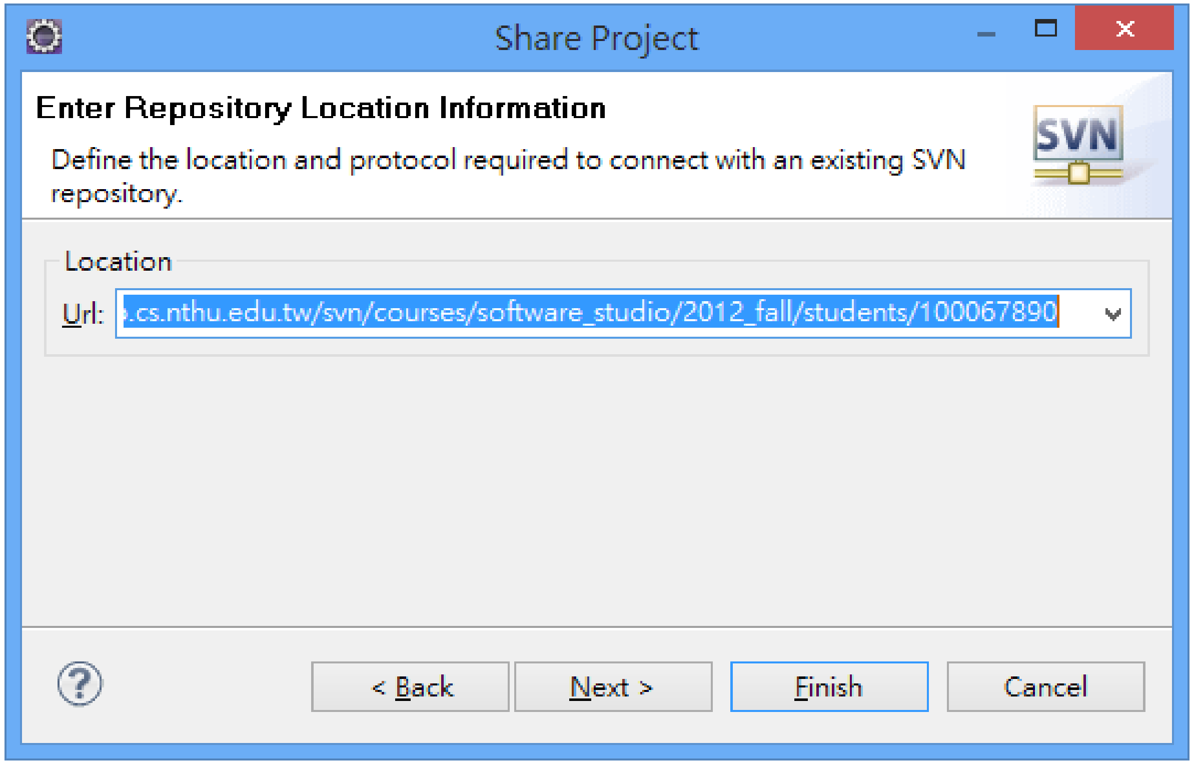
[✓ Create a new repository location]

Use existing repository location:

https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/instructor
https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/samples

Import Project to Repository

- https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/students/${your-id}



Enter Repository Location Information

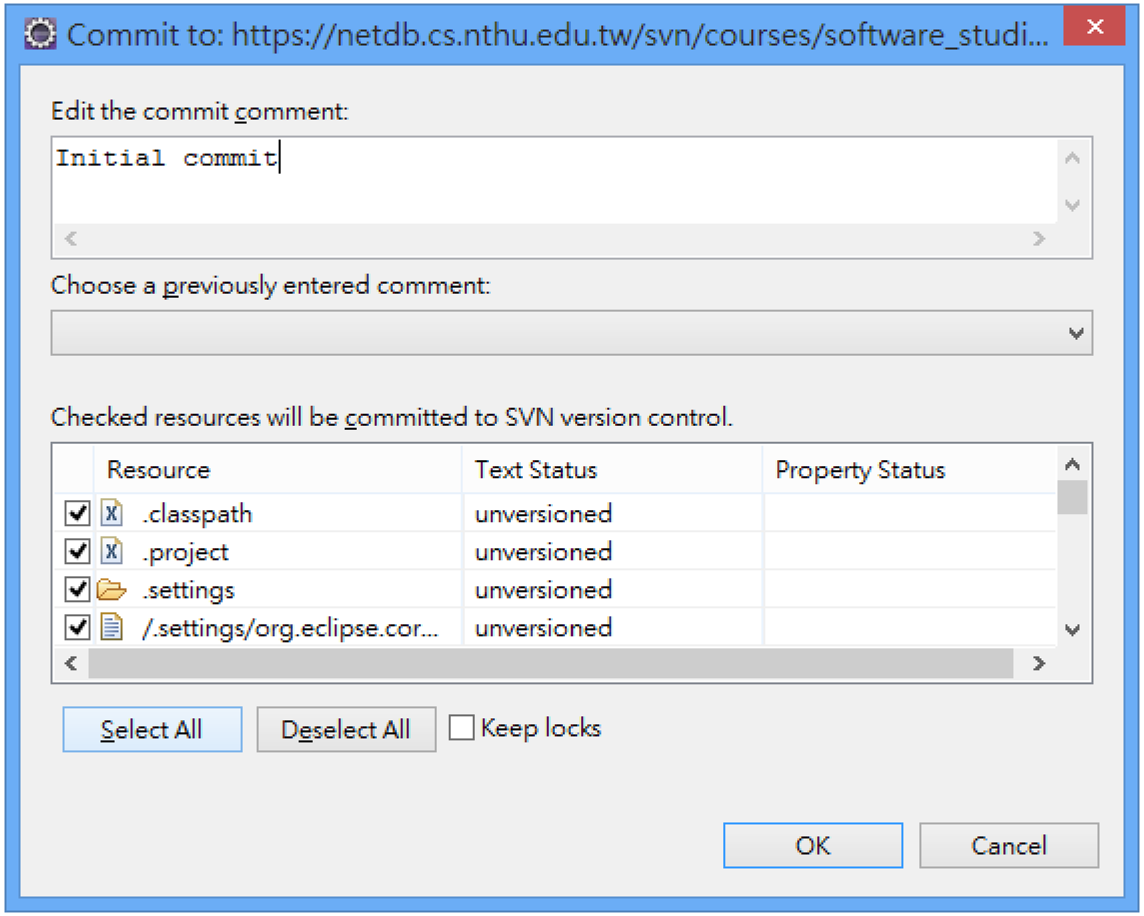
Define the location and protocol required to connect with an existing SVN repository.

Location

Url: .cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/students/100067890

Import Project to Repository

- Click "Select All" to import all resources



Commit to: https://netdb.cs.nthu.edu.tw/svn/courses/software_studi...

Edit the commit comment:

```
Initial commit
```

Choose a previously entered comment:

Checked resources will be committed to SVN version control.

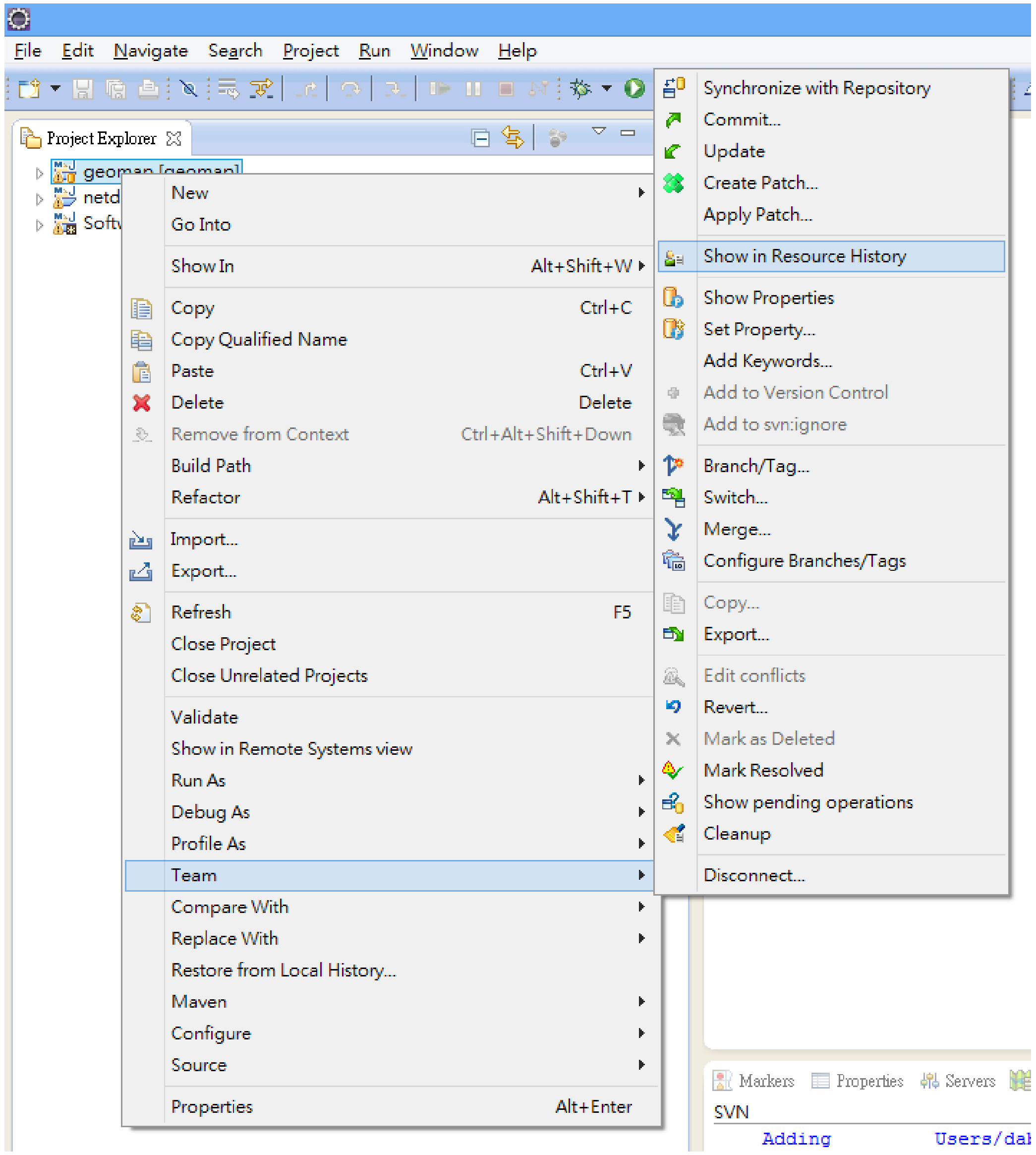
	Resource	Text Status	Property Status
✓	.classpath	unversioned	
✓	.project	unversioned	
✓	.settings	unversioned	
✓	/.settings/org.eclipse.cor...	unversioned	

What's Next?

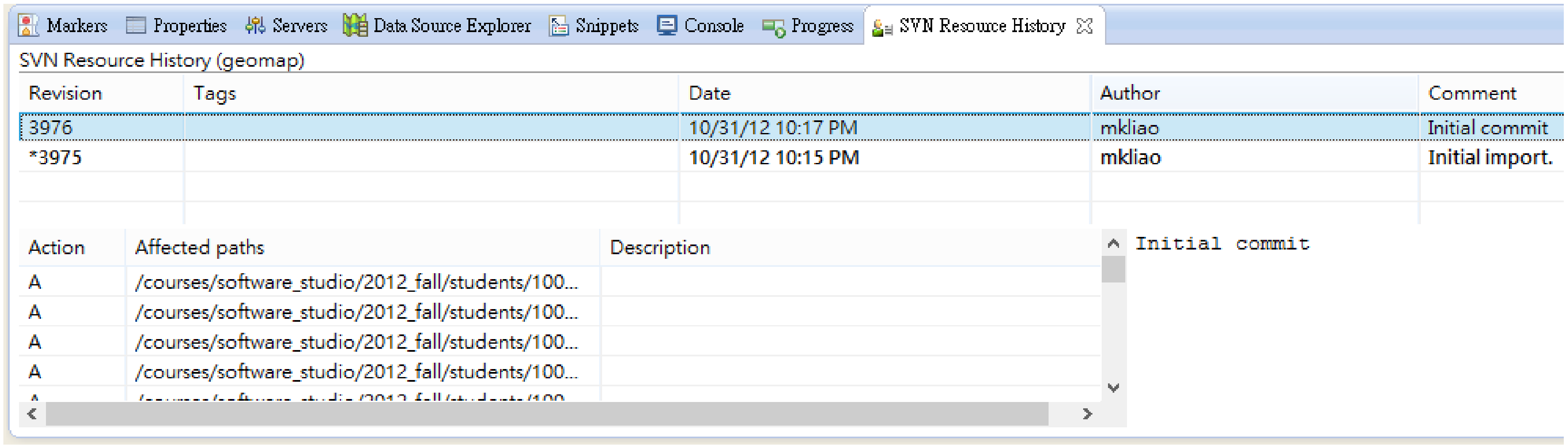
- View the revision history
- Switch to older versions
- Lookup differences while conflicting

Revision History

- Right click -> Team -> Show history



Revision History



Markers | Properties | Servers | Data Source Explorer | Snippets | Console | Progress | SVN Resource History

SVN Resource History (geomap)

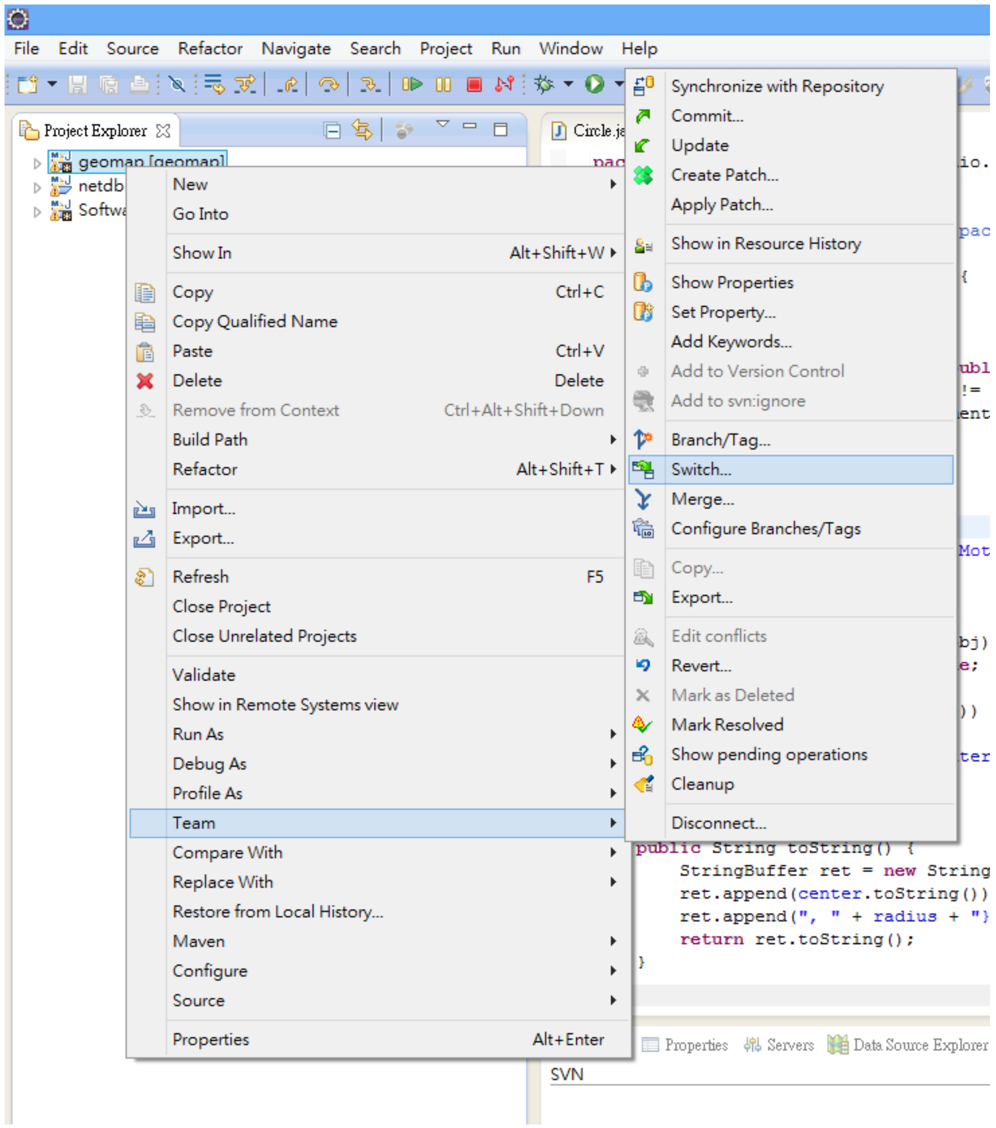
Revision	Tags	Date	Author	Comment
3976		10/31/12 10:17 PM	mkliao	Initial commit
*3975		10/31/12 10:15 PM	mkliao	Initial import.

Action	Affected paths	Description
A	/courses/software_studio/2012_fall/students/100...	
A	/courses/software_studio/2012_fall/students/100...	
A	/courses/software_studio/2012_fall/students/100...	
A	/courses/software_studio/2012_fall/students/100...	

```
Initial commit
```

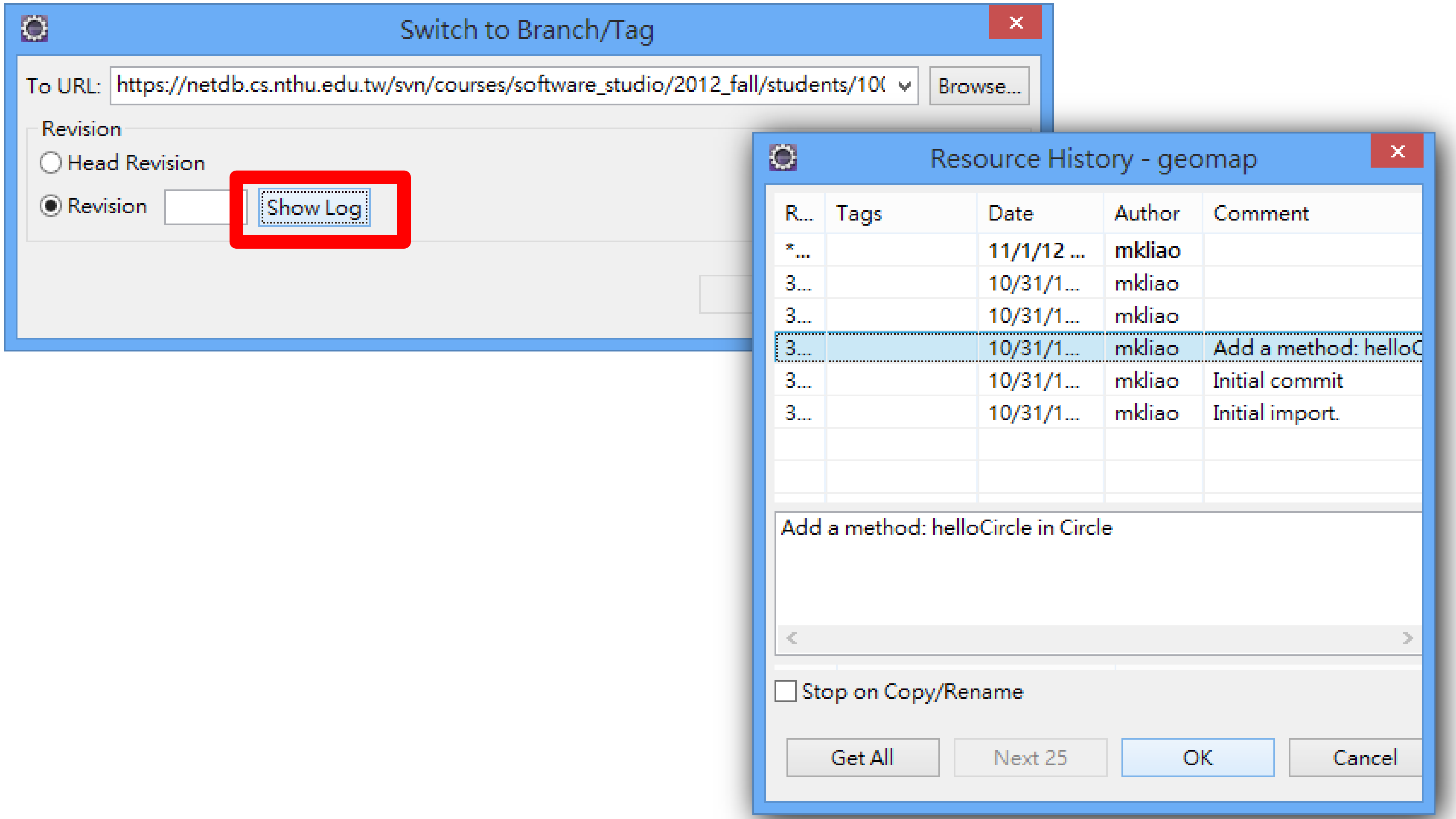
Switch to Older Versions

- Right click -> Team -> Switch



```
public String toString() {
    StringBuffer ret = new String
    ret.append(center.toString())
    ret.append(", " + radius + "}
    return ret.toString();
}
```

Switch to Older Versions



Switch to Branch/Tag

To URL: https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/students/10(

Browse...

Revision

( ) Head Revision

(●) Revision

Show Log

Resource History - geomap

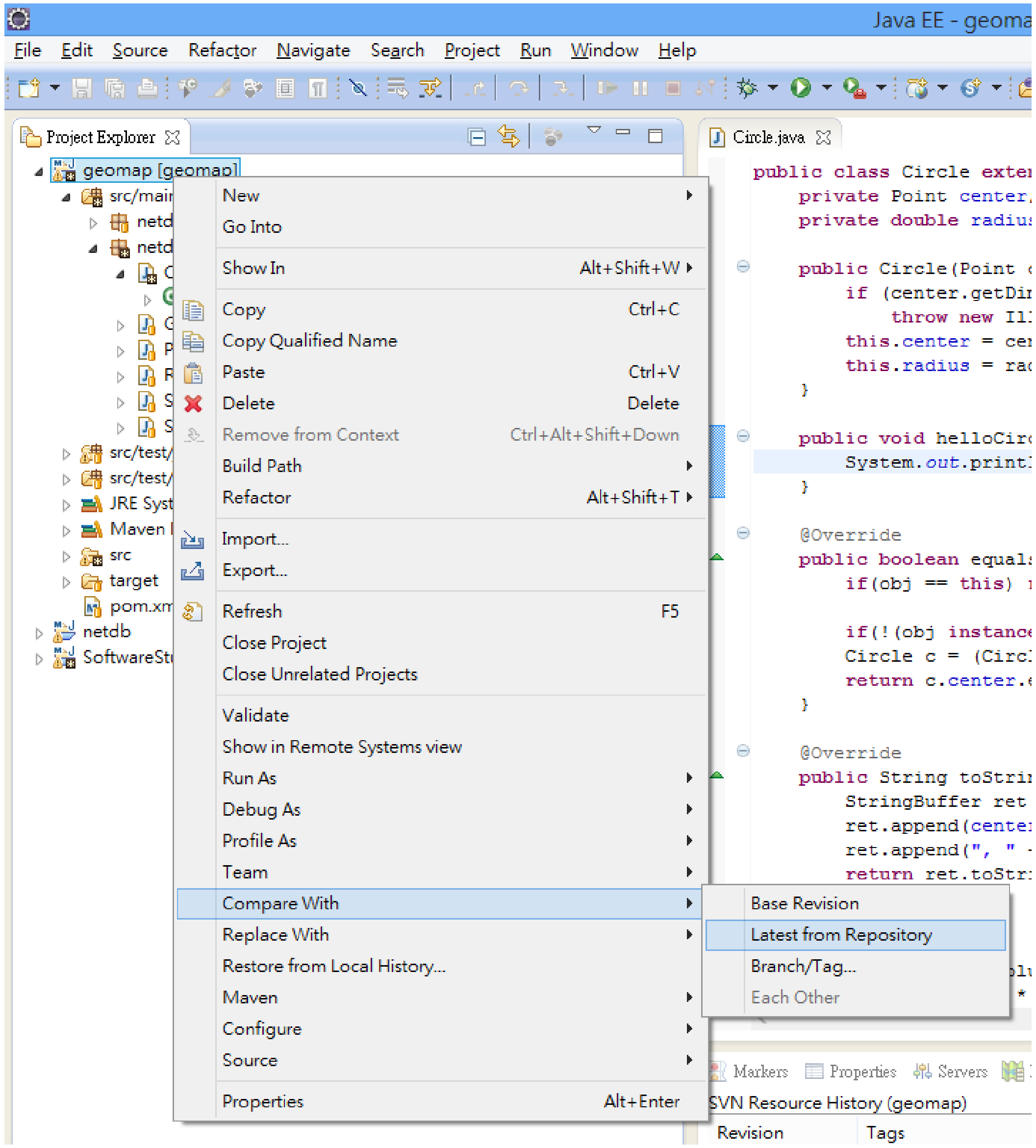
R...	Tags	Date	Author	Comment
*...		11/1/12 ...	mkliao	
3...		10/31/1...	mkliao	
3...		10/31/1...	mkliao	
3...		10/31/1...	mkliao	Add a method: helloC
3...		10/31/1...	mkliao	Initial commit
3...		10/31/1...	mkliao	Initial import.

Add a method: helloCircle in Circle

☐ Stop on Copy/Rename

Get All | Next 25 | OK | Cancel

Lookup Difference

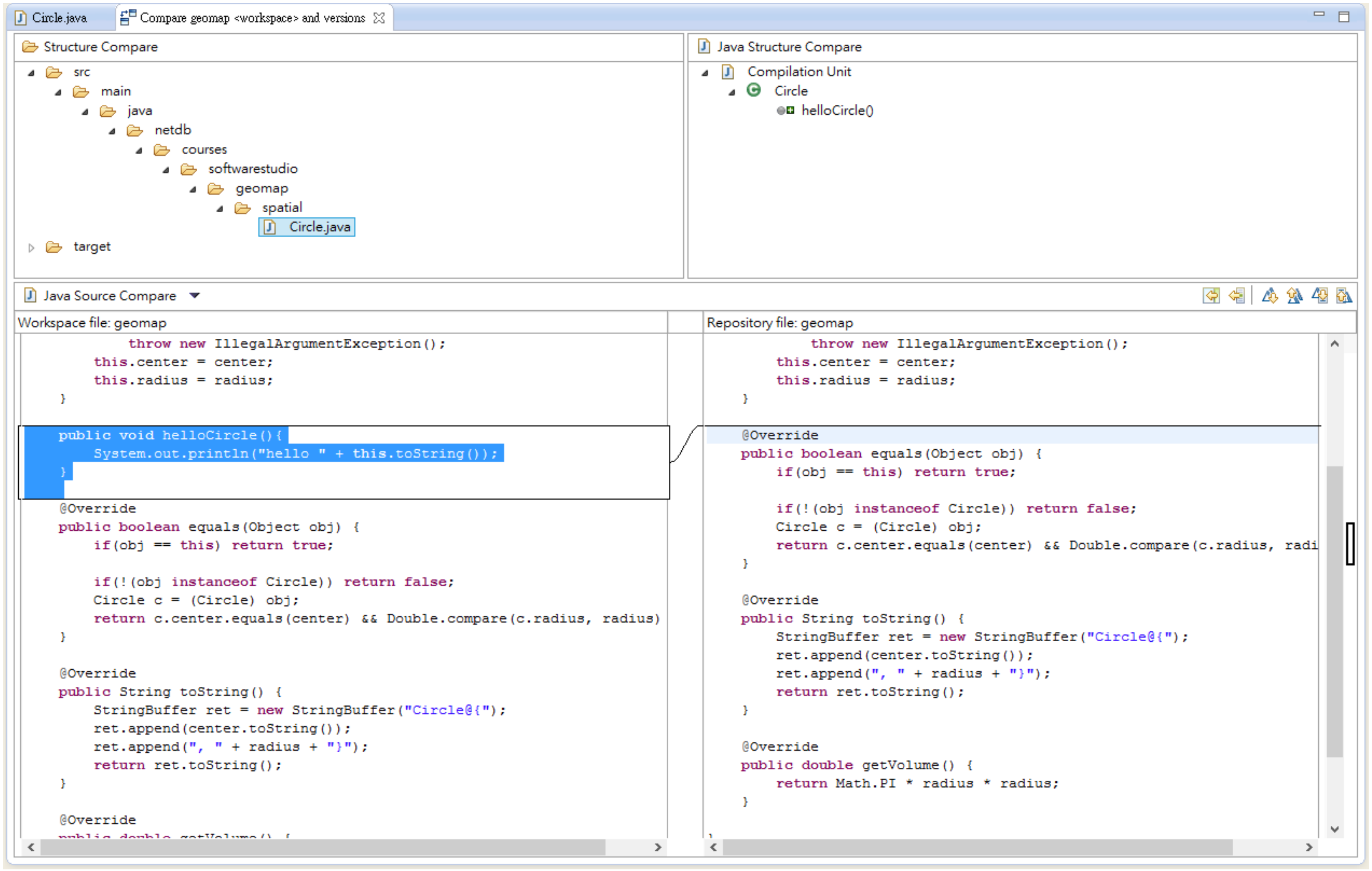


```
public class Circle exte
    private Point center,
    private double radiu

    public Circle(Point
        if (center.getDi
            throw new Il
        this.center = ce
        this.radius = ra
    }

    public void helloCir
        System.out.print
    }

    @Override
    public boolean equal
        if(obj == this)

        if(!(obj instanc
        Circle c = (Circ
        return c.center.
    }

    @Override
    public String toStri
        StringBuffer ret
        ret.append(cente
        ret.append(", "
        return ret.toStr
```

- Right click -> Compare with -> ...

- We can lookup the differences between
  1. Local files
  2. Local file and file in repository

Lookup Difference



Circle.java

Compare geomap <workspace> and versions

Structure Compare

- src
  - main
    - java
      - netdb
        - courses
          - softwarestudio
            - geomap
              - spatial
                - Circle.java
- target

Java Structure Compare

- Compilation Unit
  - Circle
    - helloCircle()

Java Source Compare

Workspace file: geomap

```
            throw new IllegalArgumentException();
        this.center = center;
        this.radius = radius;
    }

    public void helloCircle(){
        System.out.println("hello " + this.toString());
    }

    @Override
    public boolean equals(Object obj) {
        if(obj == this) return true;

        if(!(obj instanceof Circle)) return false;
        Circle c = (Circle) obj;
        return c.center.equals(center) && Double.compare(c.radius, radius)
    }

    @Override
    public String toString() {
        StringBuffer ret = new StringBuffer("Circle@{");
        ret.append(center.toString());
        ret.append(", " + radius + "}");
        return ret.toString();
    }

    @Override
```

Repository file: geomap

```
            throw new IllegalArgumentException();
        this.center = center;
        this.radius = radius;
    }

    @Override
    public boolean equals(Object obj) {
        if(obj == this) return true;

        if(!(obj instanceof Circle)) return false;
        Circle c = (Circle) obj;
        return c.center.equals(center) && Double.compare(c.radius, radi
    }

    @Override
    public String toString() {
        StringBuffer ret = new StringBuffer("Circle@{");
        ret.append(center.toString());
        ret.append(", " + radius + "}");
        return ret.toString();
    }

    @Override
    public double getVolume() {
        return Math.PI * radius * radius;
    }
```

Local file

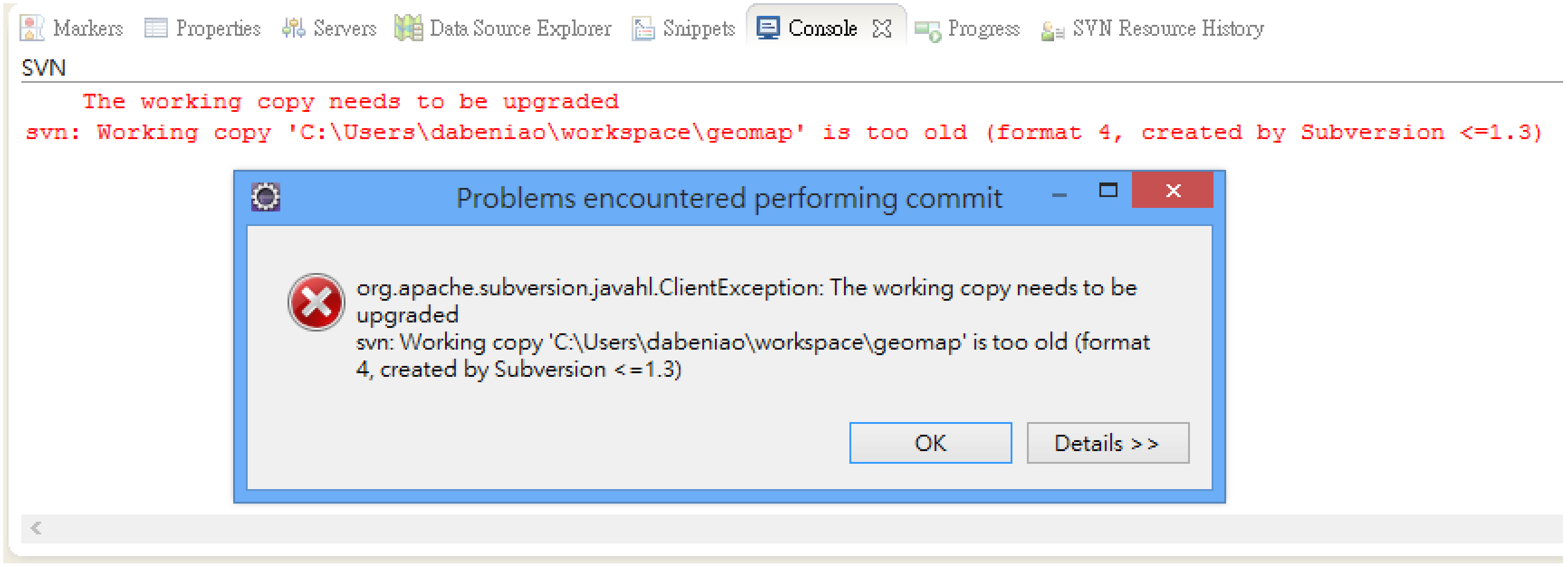
File in the repository

Merging Conflicts

- What to do when conflicts occur at commit time?
  - You must resolve conflicts before proceeding
- First, lookup the differences
- Then you have options:
  - Modify your code locally, merge the changes, and commit merged version to SVN
  - Discard your changes by reverting your code to an older version
  - Contact other authors to modify/revert their code

Merging Conflicts

- Conflict when commit



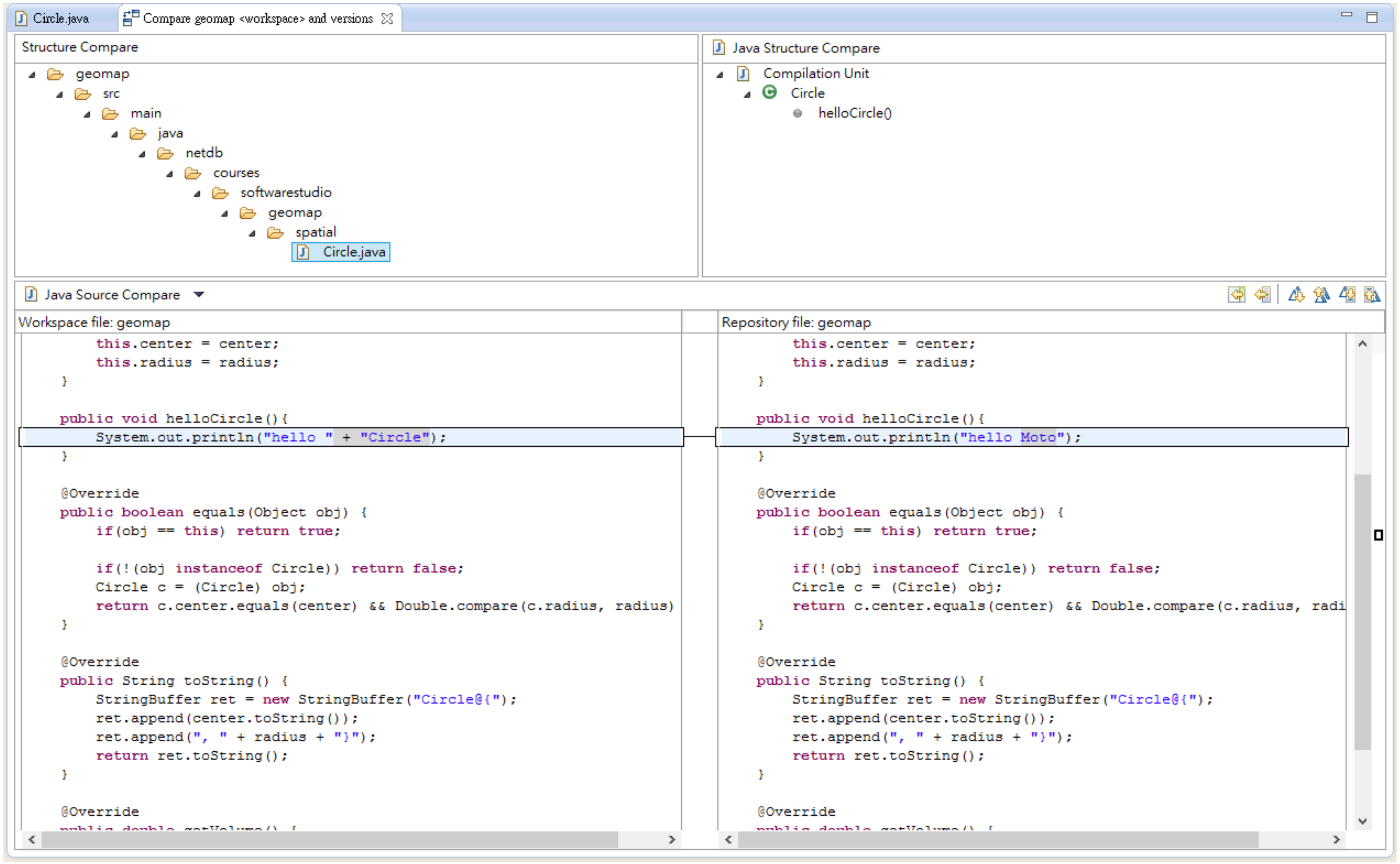
```
    The working copy needs to be upgraded
svn: Working copy 'C:\Users\dabeniao\workspace\geomap' is too old (format 4, created by Subversion <=1.3)
```

Problems encountered performing commit

org.apache.subversion.javahl.ClientException: The working copy needs to be upgraded
svn: Working copy 'C:\Users\dabeniao\workspace\geomap' is too old (format 4, created by Subversion <=1.3)

Merging Conflicts

- Lookup Difference



Circle.java

Compare geomap <workspace> and versions

Structure Compare

```
geomap
  src
    main
      java
        netdb
          courses
            softwarestudio
              geomap
                spatial
                  Circle.java
```

Java Structure Compare

```
Compilation Unit
  Circle
    helloCircle()
```

Java Source Compare

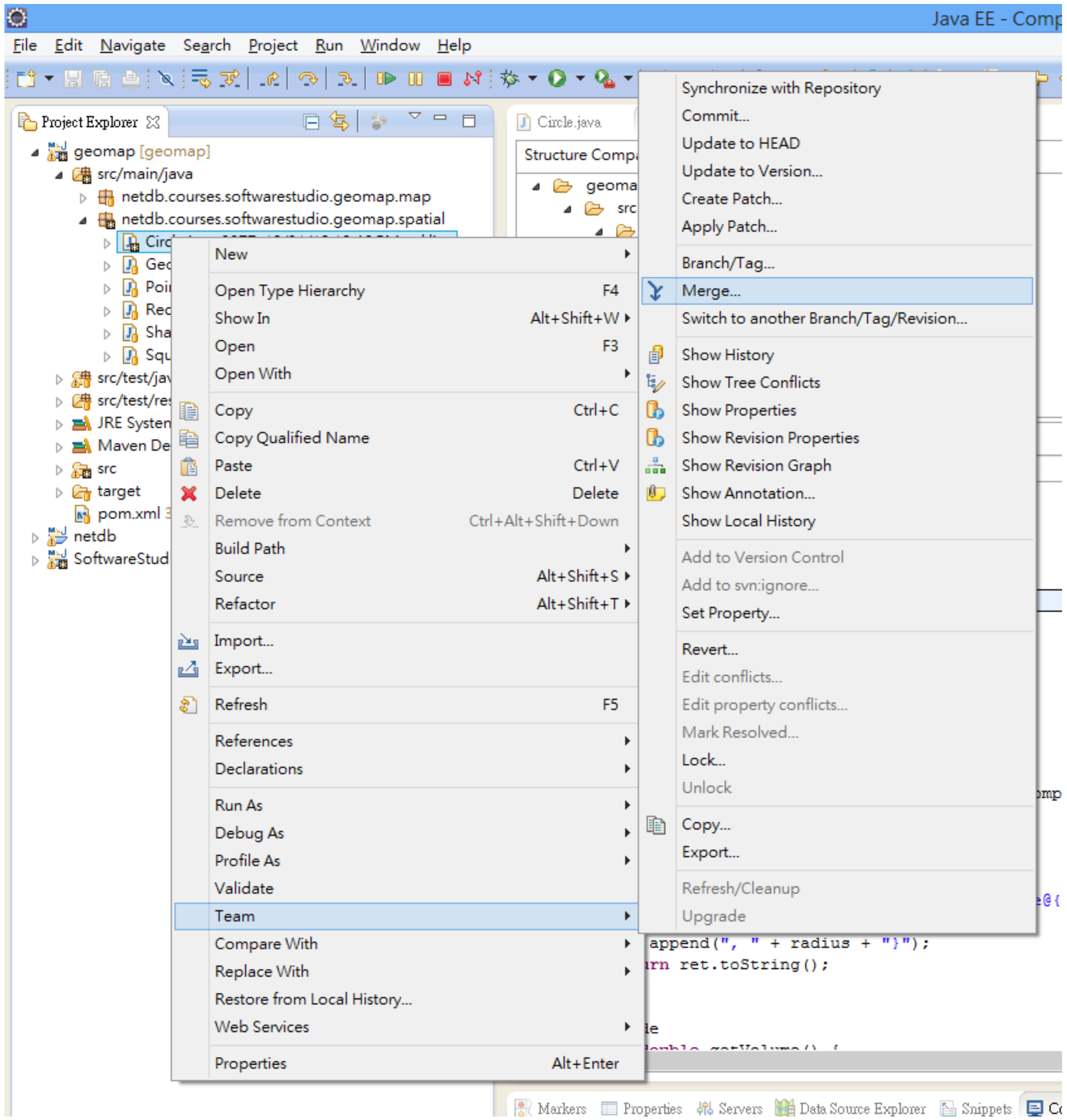
Workspace file: geomap

```
        this.center = center;
        this.radius = radius;
    }

    public void helloCircle(){
        System.out.println("hello " + "Circle");
    }

    @Override
    public boolean equals(Object obj) {
        if(obj == this) return true;

        if(!(obj instanceof Circle)) return false;
        Circle c = (Circle) obj;
        return c.center.equals(center) && Double.compare(c.radius, radius)
    }

    @Override
    public String toString() {
        StringBuffer ret = new StringBuffer("Circle@{");
        ret.append(center.toString());
        ret.append(", " + radius + "}");
        return ret.toString();
    }

    @Override
```

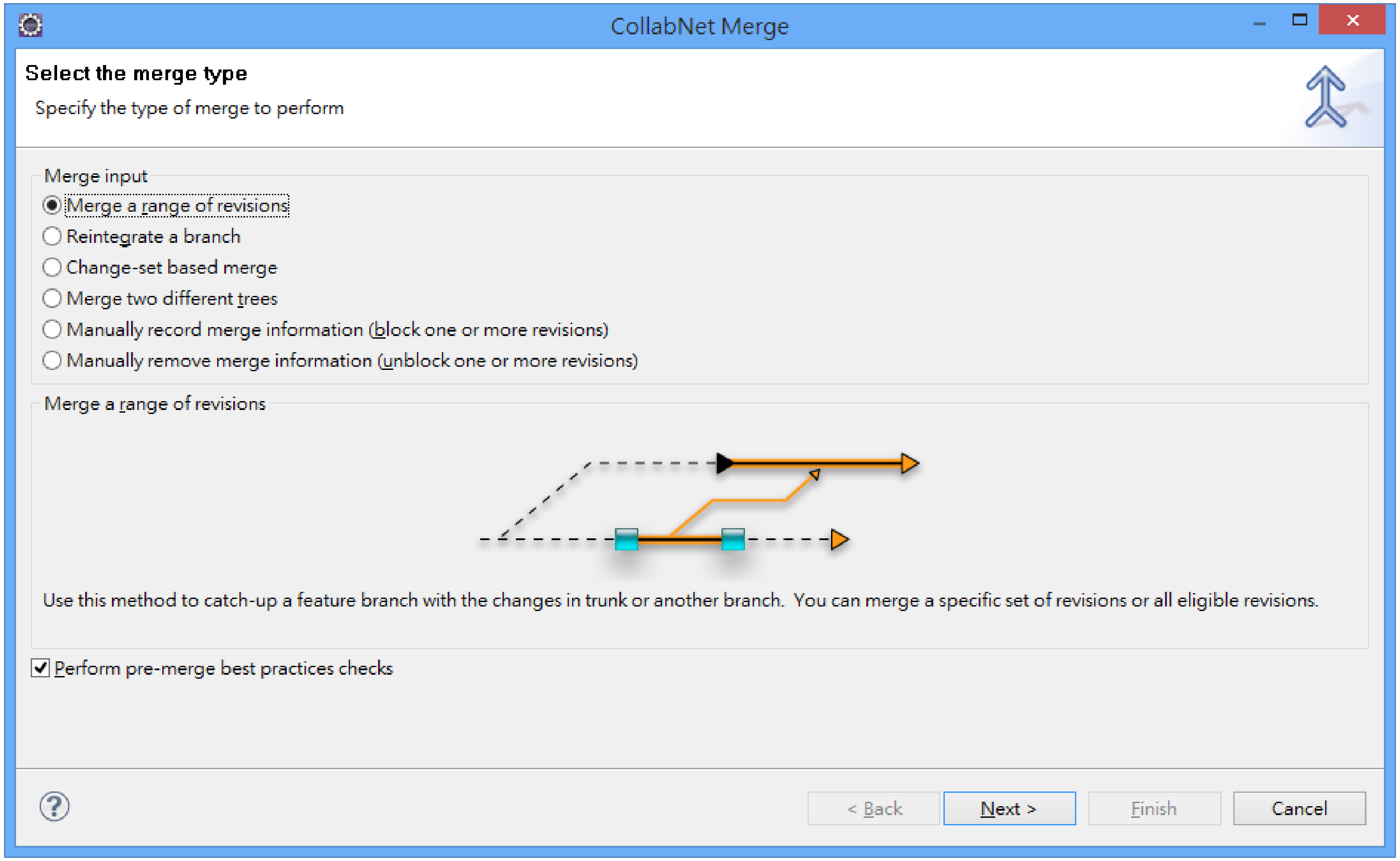
Repository file: geomap

```
        this.center = center;
        this.radius = radius;
    }

    public void helloCircle(){
        System.out.println("hello Moto");
    }

    @Override
    public boolean equals(Object obj) {
        if(obj == this) return true;

        if(!(obj instanceof Circle)) return false;
        Circle c = (Circle) obj;
        return c.center.equals(center) && Double.compare(c.radius, radi
    }

    @Override
    public String toString() {
        StringBuffer ret = new StringBuffer("Circle@{");
        ret.append(center.toString());
        ret.append(", " + radius + "}");
        return ret.toString();
    }

    @Override
```

Merging Conflicts



Merging Conflicts


CollabNet Merge

Select the merge type

Specify the type of merge to perform

Merge input

- [✓] Merge a range of revisions
- [ ] Reintegrate a branch
- [ ] Change-set based merge
- [ ] Merge two different trees
- [ ] Manually record merge information (block one or more revisions)
- [ ] Manually remove merge information (unblock one or more revisions)

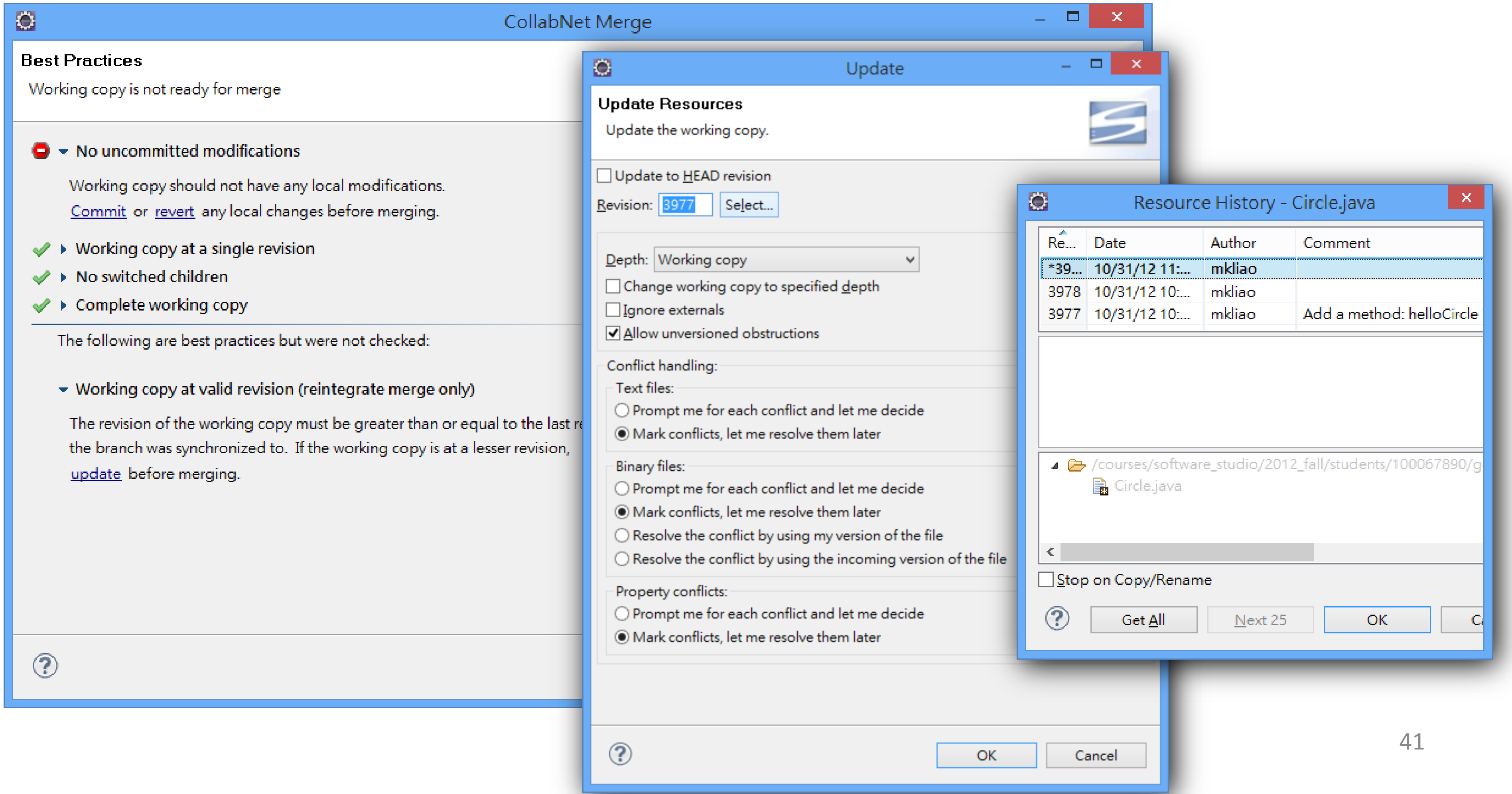
Merge a range of revisions

Use this method to catch-up a feature branch with the changes in trunk or another branch. You can merge a specific set of revisions or all eligible revisions.

[✓] Perform pre-merge best practices checks

Merging Conflicts

- There are options to process the merge.



CollabNet Merge

Best Practices

Working copy is not ready for merge

- No uncommitted modifications

  Working copy should not have any local modifications.
  Commit or revert any local changes before merging.
- ✓ Working copy at a single revision
- ✓ No switched children
- ✓ Complete working copy

The following are best practices but were not checked:

- Working copy at valid revision (reintegrate merge only)

  The revision of the working copy must be greater than or equal to the last re
  the branch was synchronized to. If the working copy is at a lesser revision,
  update before merging.

Update

Update Resources

Update the working copy.

- [ ] Update to HEAD revision

Revision: 3977 Select...

Depth: Working copy

- [ ] Change working copy to specified depth
- [ ] Ignore externals
- [✓] Allow unversioned obstructions

Conflict handling:

Text files:
- ○ Prompt me for each conflict and let me decide
- ● Mark conflicts, let me resolve them later

Binary files:
- ○ Prompt me for each conflict and let me decide
- ● Mark conflicts, let me resolve them later
- ○ Resolve the conflict by using my version of the file
- ○ Resolve the conflict by using the incoming version of the file

Property conflicts:
- ○ Prompt me for each conflict and let me decide
- ● Mark conflicts, let me resolve them later

Resource History - Circle.java

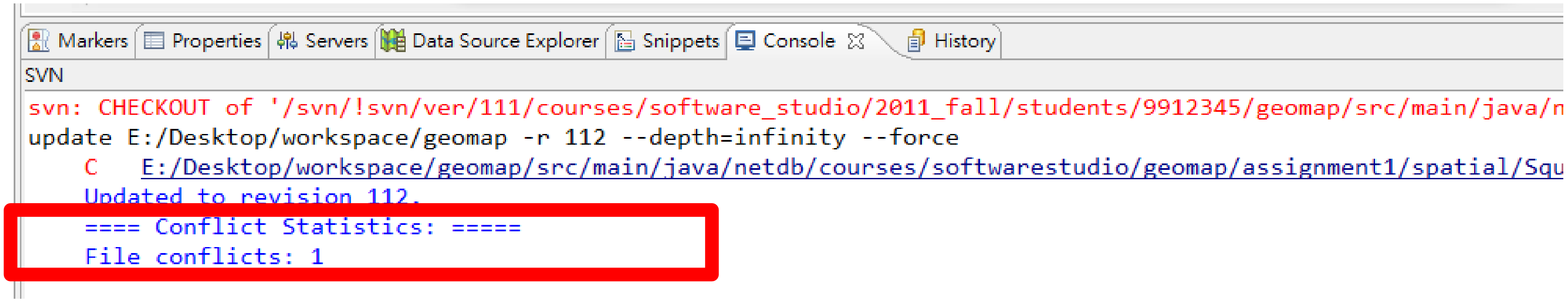
Re...	Date	Author	Comment
*39...	10/31/12 11:...	mkliao	
3978	10/31/12 10:...	mkliao	
3977	10/31/12 10:...	mkliao	Add a method: helloCircle

/courses/software_studio/2012_fall/students/100067890/g
Circle.java

- [ ] Stop on Copy/Rename

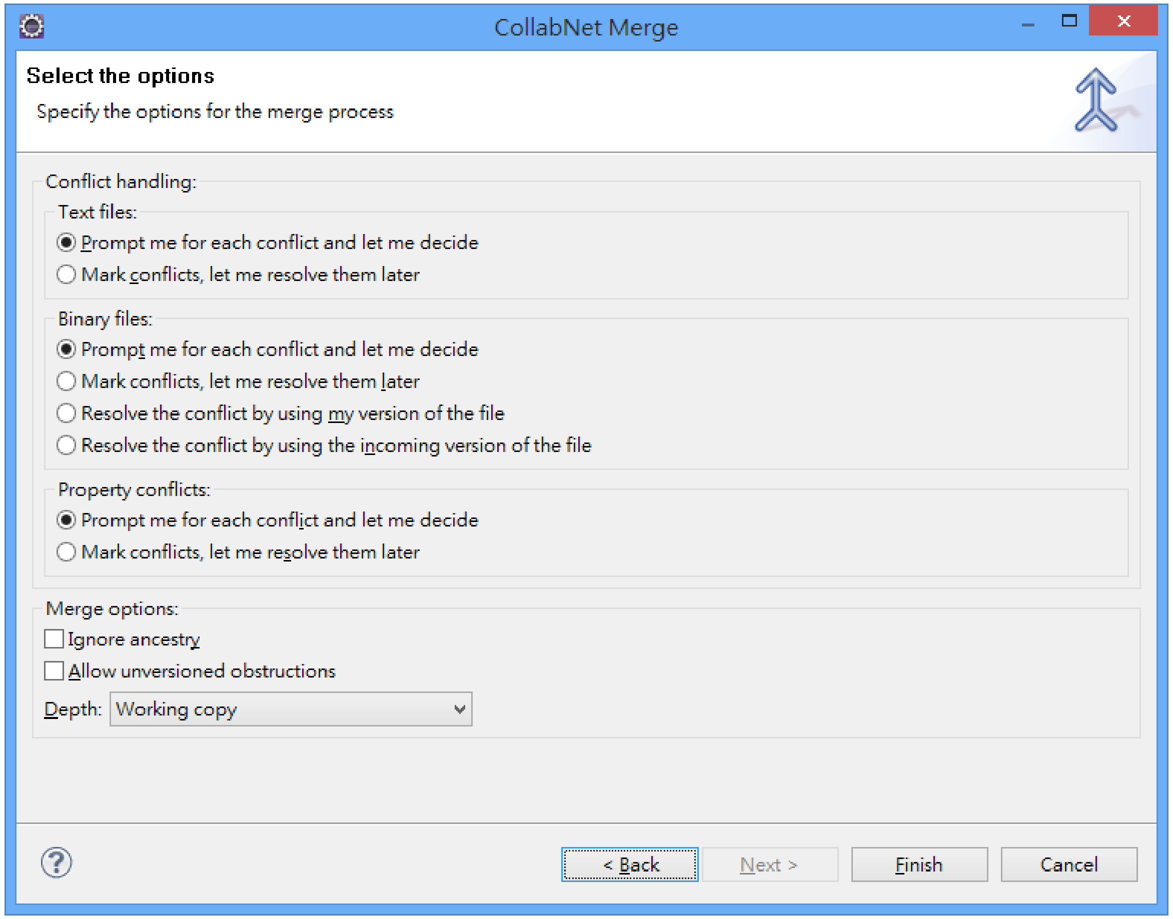
Merging Conflicts

- The console will show the conflicts



```
svn: CHECKOUT of '/svn/!svn/ver/111/courses/software_studio/2011_fall/students/9912345/geomap/src/main/java/n
update E:/Desktop/workspace/geomap -r 112 --depth=infinity --force
    C   E:/Desktop/workspace/geomap/src/main/java/netdb/courses/softwarestudio/geomap/assignment1/spatial/Squ
    Updated to revision 112.
    ==== Conflict Statistics: =====
    File conflicts: 1
```

Merging Conflicts



Select the options

Specify the options for the merge process

Conflict handling:

Text files:

- (●) Prompt me for each conflict and let me decide
- ( ) Mark conflicts, let me resolve them later

Binary files:

- (●) Prompt me for each conflict and let me decide
- ( ) Mark conflicts, let me resolve them later
- ( ) Resolve the conflict by using my version of the file
- ( ) Resolve the conflict by using the incoming version of the file

Property conflicts:

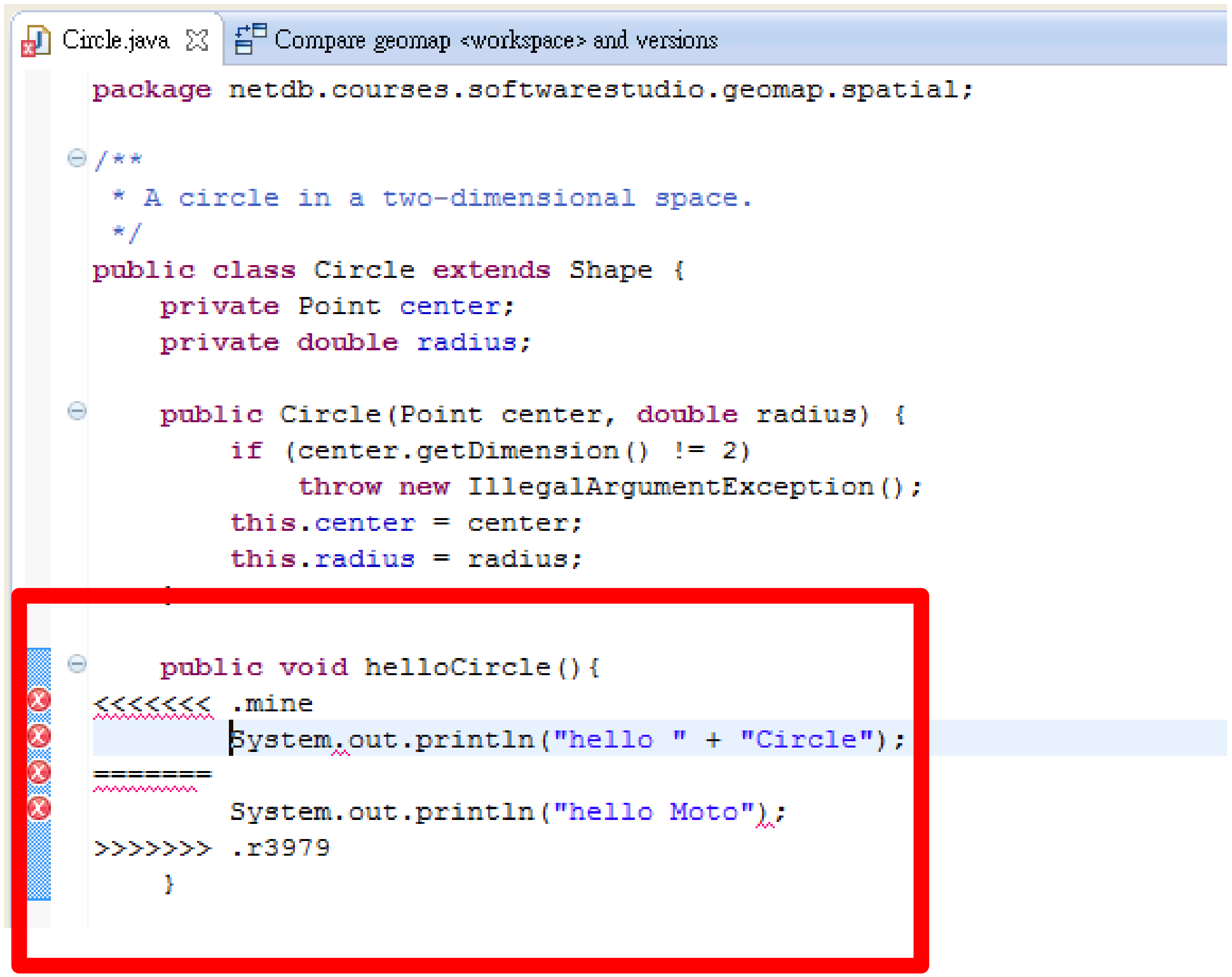
- (●) Prompt me for each conflict and let me decide
- ( ) Mark conflicts, let me resolve them later

Merge options:

- [ ] Ignore ancestry
- [ ] Allow unversioned obstructions

Depth: Working copy

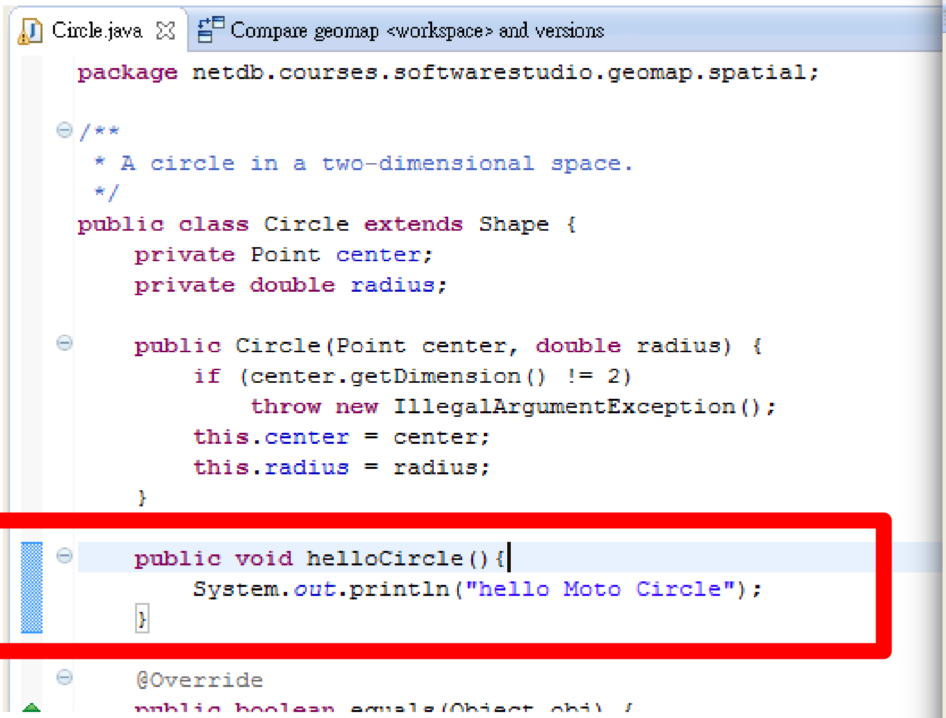
Merging Conflicts



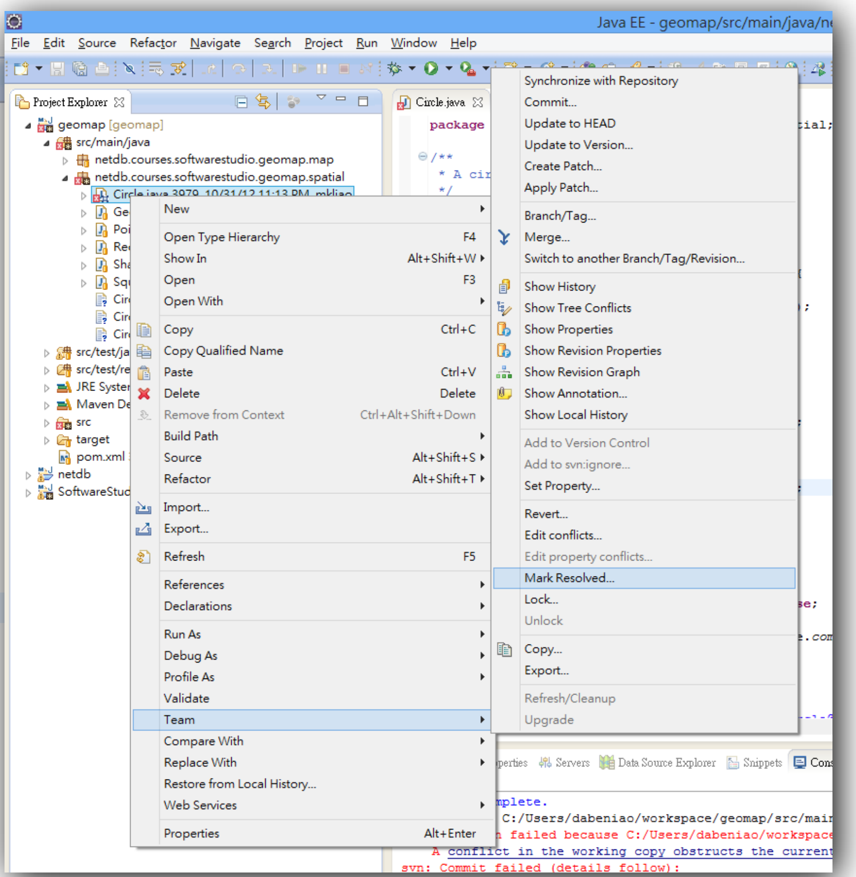
```
package netdb.courses.softwarestudio.geomap.spatial;

/**
 * A circle in a two-dimensional space.
 */
public class Circle extends Shape {
    private Point center;
    private double radius;

    public Circle(Point center, double radius) {
        if (center.getDimension() != 2)
            throw new IllegalArgumentException();
        this.center = center;
        this.radius = radius;
    }

    public void helloCircle(){
<<<<<<< .mine
        System.out.println("hello " + "Circle");
=======
        System.out.println("hello Moto");
>>>>>>> .r3979
    }
```

Merging Conflicts

- Commit it after the conflicts are resolved



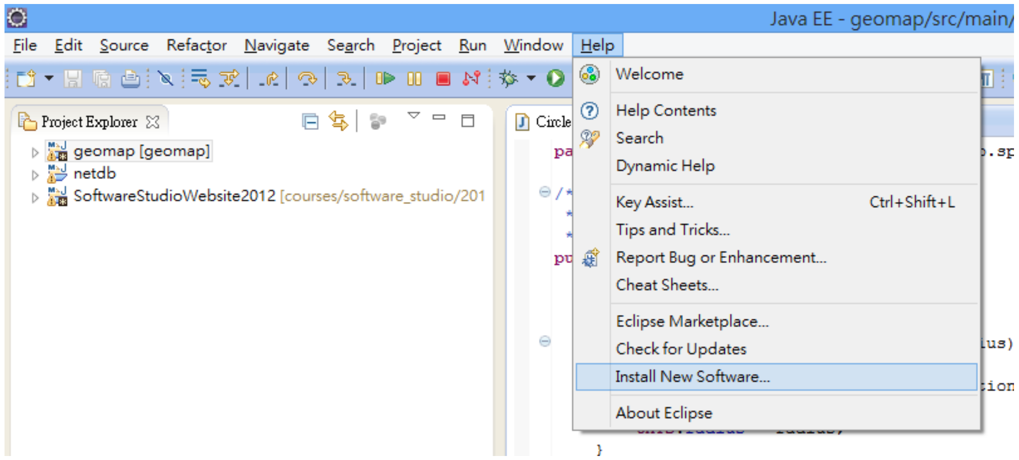
```
package netdb.courses.softwarestudio.geomap.spatial;

/**
 * A circle in a two-dimensional space.
 */
public class Circle extends Shape {
    private Point center;
    private double radius;

    public Circle(Point center, double radius) {
        if (center.getDimension() != 2)
            throw new IllegalArgumentException();
        this.center = center;
        this.radius = radius;
    }

    public void helloCircle(){
        System.out.println("hello Moto Circle");
    }

    @Override
    public boolean equals(Object obj) {
```

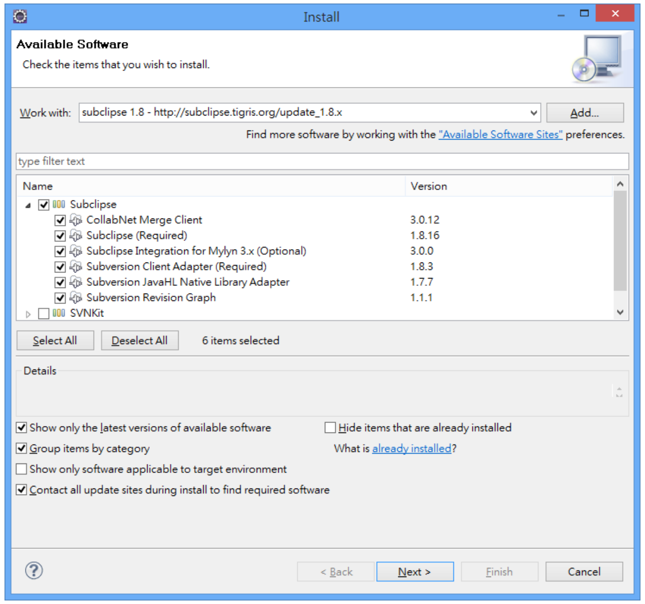


```
mplete.
 C:/Users/dabeniao/workspace/geomap/src/main
n failed because C:/Users/dabeniao/workspace
   A conflict in the working copy obstructs the current
svn: Commit failed (details follow):
```

Merging Conflicts

- The merge tool mentioned here is only available in subclipse 1.6.x or later
- You can install version 1.8.x in eclipse using the following URL
  - http://subclipse.tigris.org/update_1.8.x


Java EE - geomap/src/main/
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
geomap [geomap]
netdb
SoftwareStudioWebsite2012 [courses/software_studio/201
Circle
Welcome
Help Contents
Search
Dynamic Help
Key Assist... Ctrl+Shift+L
Tips and Tricks...
Report Bug or Enhancement...
Cheat Sheets...
Eclipse Marketplace...
Check for Updates
Install New Software...
About Eclipse


Install

Available Software

Check the items that you wish to install.

Work with: subclipse 1.8 - http://subclipse.tigris.org/update_1.8.x [Add...]

Find more software by working with the "Available Software Sites" preferences.

type filter text

Name	Version
☑ Subclipse	
☑ CollabNet Merge Client	3.0.12
☑ Subclipse (Required)	1.8.16
☑ Subclipse Integration for Mylyn 3.x (Optional)	3.0.0
☑ Subversion Client Adapter (Required)	1.8.3
☑ Subversion JavaHL Native Library Adapter	1.7.7
☑ Subversion Revision Graph	1.1.1
☐ SVNKit	

Select All | Deselect All | 6 items selected

Details

☑ Show only the latest versions of available software
☐ Hide items that are already installed
☑ Group items by category
What is already installed?
☐ Show only software applicable to target environment
☑ Contact all update sites during install to find required software

< Back | Next > | Finish | Cancel

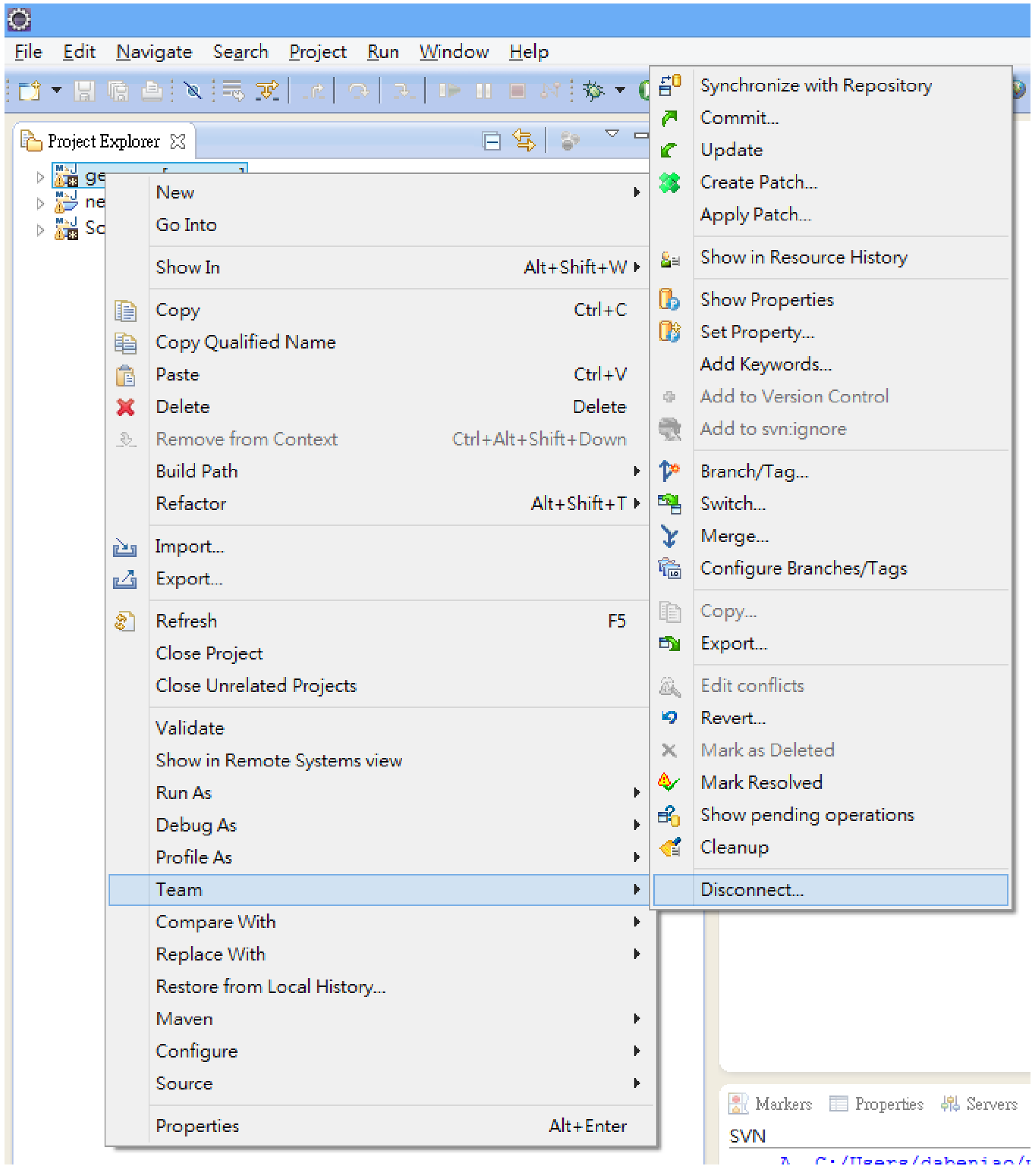
Outline

- Why do we need version control?
- Introduction to SVN
- Using SVN Client in Eclipse
- Exercise and Homework Submission

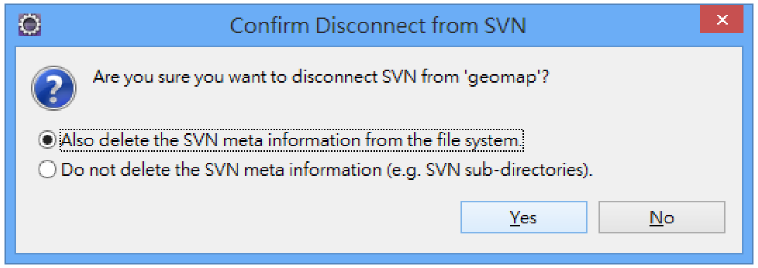
How to Submit Your Solutions to Assignments?

- Before the deadline, import your project into https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/students/${your-student-id}/${proj-name}
  - E.g., https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/students/100067890/geomap
- **Do not** modify the sample projects checked out from https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/samples directly!
  - They are connected to sample projects on SVN
  - You need to break the connection first

Breaking Connections



- After breaking the connection, you have a "local" project
- Then modify this project, and import it to SVN as your solution



Are you sure you want to disconnect SVN from 'geomap'?

● Also delete the SVN meta information from the file system.
○ Do not delete the SVN meta information (e.g. SVN sub-directories).

Exercise: Submitting A Dummy Solution

1. Checkout the latest version of *geomap* project from https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/samples/
2. Break the connection
3. Implement the following methods
   - addShape(Shape s)
   - addShapes(Shape[] s)
4. Import the project into your own repository
   - Notify TA when you are done
   - Open this link in your browser first so we can speed up the demo process https://netdb.cs.nthu.edu.tw/svn/courses/software_studio/2012_fall/students/${your-student-id}/geomap/

Authentication Required

- Account:
  - Your student ID
- Password:
  - Your student ID by default
  - Change it using this page: https://netdb.cs.nthu.edu.tw/svntools/passwd/index.php

A Note on Common Directory Structure

- In practice, many (open source) projects consist of the following 3 subdirectories:
  - trunk
    - Main line of development
    - Changing frequently
  - branches
    - Bug fixing
    - New features
    - Preparation of release
    - Should be merged back into trunk later
  - tags
    - Releases
    - Milestones of development
    - Making searches of older versions easier
- For simplicity, the projects we develop in this course contains only what resides in trunk