

Version Control Systems

The screenshot shows the SourceForge homepage. At the top, there's a navigation bar with links for Search, Browse, Enterprise, Blog, Help, and Jobs. Below that is another navigation bar with links for SOLUTION CENTERS, Go Parallel, Smarter IT, Resources, and Newsletters. The main content area features a large banner with the text "Find, Create, and Publish Open Source software for free". It includes a search bar with the placeholder "Search from thousands of software titles" and a "Search" button. Below the banner, there are statistics: TODAY 4,343,789 DOWNLOADS, 24,360 CODE COMMITS, 1,573 FORUM POSTS, 2,736 BUGS TRACKED, and a "MORE DETAILS" link. A sidebar on the left lists categories like Audio & Video, Business & Enterprise, Communications, Development, Home & Education, Games, Graphics, Science & Engineering, and Security & Utilities. The main content area also displays "Projects OF The Month" with entries for "Staff Choice Win32 Disk Imager" and "Community Choice Universal Media Server". Each entry has a thumbnail, a title, a brief description, and a "Download" button.

The screenshot shows a GitHub repository page for "openlink / virtuoso-opensource". The top navigation bar includes links for This repository, Search or type a command, Explore, Features, Enterprise, and Blog, along with Sign up and Sign In buttons. The repository details show it is PUBLIC and owned by "openlink". It has 10,000+ commits, 7 branches, 38 releases, and 6 contributors. A dropdown menu indicates the current branch is "develop7". The main content area shows a list of recent commits, such as "Merge branch 'develop6' into develop7" authored by "VOS Maintainer" 2 days ago, and other commits from "appsrc", "bin", and "binc" updated copyright year and merged into "develop7" 2 days ago.

- Source Forge, Google Code (shelved), Github

Version Control Systems (cont.)

Version Control Systems

- A software tool that can store source code @ a central location
- Keeps a record of changes that have been made from time to time
- Keeps a record of who did what changes from time to time
- Can be used as a backup if something goes wrong in your code.

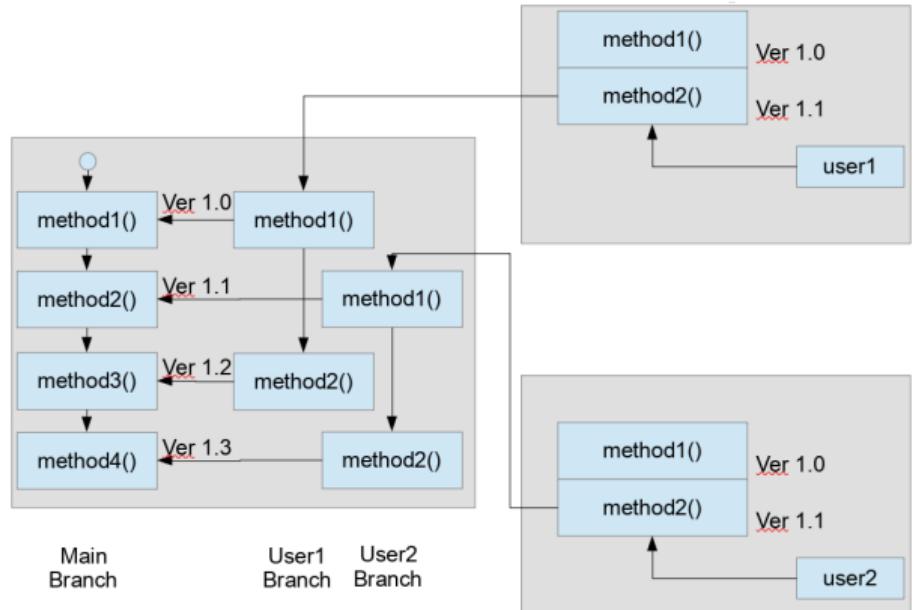
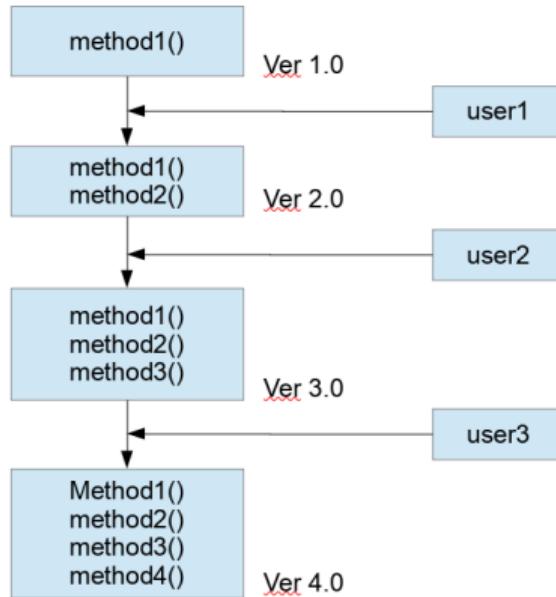
Individual Code Developer

- Can use version control as a backup tool
- Knows that the recent version is the live code
- Can maintain branches, e.g. developer branch, like branch, test branch, etc. etc. (Branches also known as Trunks)

Team Developers

- Each developer on a separate branch
- Developers can be in different parts of the world
- *upstream* can merge branches of two developers to do a version increment.

Centralized vs Distributed Version Control Systems



Centralized vs Distributed Version Control Systems (cont.)

Popular Version Control Systems

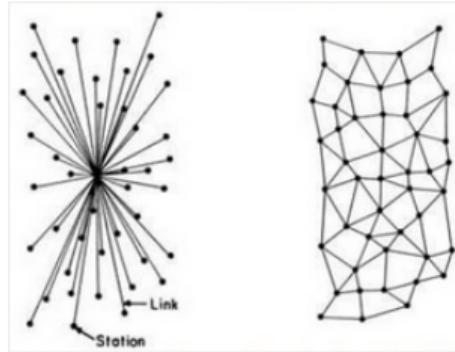
- SVN; Sub-version (Centralized)
- GIT (Distributed)
- Bazaar
- Mercurial
- ...

SVN (Centralized)

- Corruption risk. Will effect everybody
- Won't work on low network speeds
- Not scalable if users are too much
- Stores Complete Files

Git (Distributed)

- Corruption will only affect a local server
- Support for Load Balancing.
- Stores Partial Files.



Partial Storage

- Version 1.0

```
- public class test {
    public static void main(String[] args) {
    }
}
```

Stores This

- Version 2.0

```
- public class test {
    public test() {
    }
    public static void main(String[] args) {
        new test();
    }
}
```

Stores This

and This

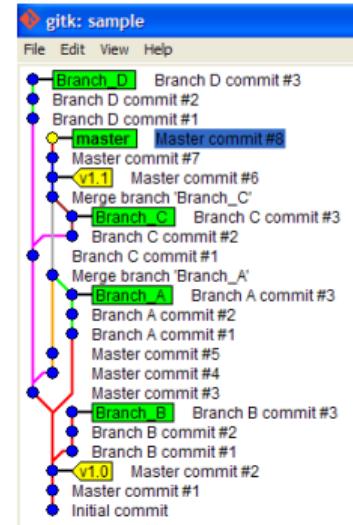
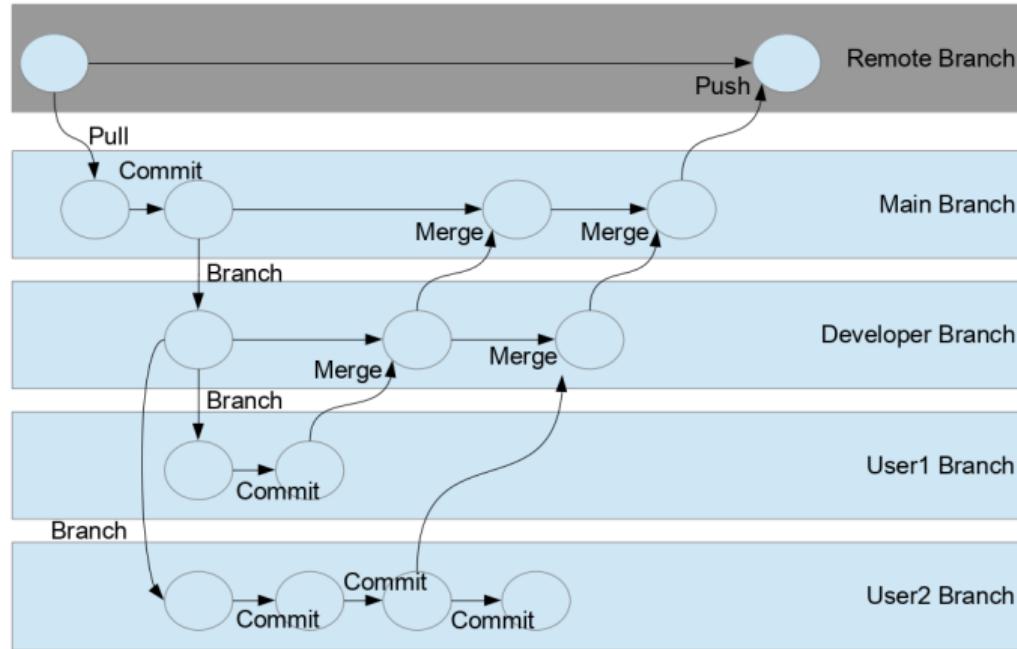
```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <assert.h>
5 #include <math.h>
6
7 #ifndef TORFFT_H
8 #define TORFFT_H
9 #include "torfft.h"
10#endif
11
12 void topo3Dexec( struct topoFFT *f,
13                   struct topoPlan3D *t)
14 {
15     int type = 1; // Do not Change
16     f->error = cSetKernelArg(f->kernelX, 7, sizeof(int), (void*)6type);
17     f->error |= cSetKernelArg(t->kernel_swap, 7, sizeof(int), (void*)6type);
18     f->error |= cSetKernelArg(t->kernel_x, 7, sizeof(int), (void*)6type);
19     f->error |= cSetKernelArg(t->kernel_y, 7, sizeof(int), (void*)6type);
20
21     /* Run Swapper */
22     t->globalSize[0] = t->x;
23     t->globalSize[1] = t->y;
24     t->globalSize[2] = t->z;
25     t->localSize[0] = t->x/2 + t->x/2;
26     t->localSize[1] = t->y/2 + t->y/2;
27     t->localSize[2] = 1;
28     f->error = cEnqueueDRangeKernel( f->command_queue, t->kernel_swap, t->dis, NULL,
29                                     t->globalSize,
30                                     NULL, 6f->event);
31     f->error |= cEnqueueDRangeKernel( f->command_queue, t->kernel_x, t->dis, NULL,
32                                     t->globalSize,
33                                     NULL, 6f->event);
```

Version 1.0

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <unistd.h>
4 #include <assert.h>
5 #include <math.h>
6
7 #ifndef TORFFT_H
8 #define TORFFT_H
9 #include "torfft.h"
10#endif
11
12 void topo3Dexec( struct topoFFT *f,
13                   struct topoPlan3D *t)
14 {
15     int type = 1; // Do not Change
16     f->error = cSetKernelArg(f->kernelX, 7, sizeof(int), (void*)6type);
17     f->error |= cSetKernelArg(t->kernel_swap, 7, sizeof(int), (void*)6type);
18     f->error |= cSetKernelArg(t->kernel_x, 7, sizeof(int), (void*)6type);
19     f->error |= cSetKernelArg(t->kernel_y, 7, sizeof(int), (void*)6type);
20
21     /* Run Swapper */
22     t->globalSize[0] = t->x;
23     t->globalSize[1] = t->y;
24     t->globalSize[2] = t->z;
25     t->localSize[0] = t->x/2 + t->x/2;
26     t->localSize[1] = t->y/2 + t->y/2;
27     t->localSize[2] = 1;
28     f->error = cEnqueueDRangeKernel( f->command_queue, t->kernel_swap, t->dis,
29                                     t->globalSize,
30                                     NULL, 6f->event);
31     f->error |= cEnqueueDRangeKernel( f->command_queue, t->kernel_x, t->dis,
32                                     t->globalSize,
33                                     NULL, 6f->event);
```

Version 2.0

Git Workflow



Git Workflow (cont.)

How to Work with these Workflows?

- Built-into Eclipse (no need to install anything else)
- Built-into Netbeans (no need to install anything else)
- Via Commands on the Command Line
- Via Clients: GitHub (browser), Gitlab (Browser), SourceTree, Git-Cola, Git-Eye, SmartGit, GitG (Linux), Giggle, ...
- Hosting your own Git: gitolite