Version Control

SNUG TechTalk

SciNet
www.scinet.utoronto.ca
University of Toronto
Toronto, Canada

October 13, 2010

SciNet

Outline

SciNet

Version Control

What is it?

- A tool for managing changes in a set of files.

SciNet

Version Control

What is it?

- A tool for managing changes in a set of files.
- Figuring out who broke what where and when.

SciNet

Version Control

What is it?

- A tool for managing changes in a set of files.
- Figuring out who broke what where and when.

Why Do it?

- Collaboration
- Organization
- Track Changes
- Faster Development
- Reduce Errors

SciNet

Collaboration

With others and yourself

Questions

SciNet

Collaboration
With others and yourself

Questions

- What if two (or more) people want to edit the same file at the same time?

SciNet

Collaboration
With others and yourself

Questions

- What if two (or more) people want to edit the same file at the same time?
- What if you work on SciNet and on your own computer?

SciNet

Collaboration
With others and yourself

Questions

- What if two (or more) people want to edit the same file at the same time?
- What if you work on SciNet and on your own computer?

Answers

Collaboration
With others and yourself

Questions

- What if two (or more) people want to edit the same file at the same time?
- What if you work on SciNet and on your own computer?

Answers

- Option 1: make them take turns
  - But then only one person can be working at any time
  - And how do you enforce the rule?

Collaboration
With others and yourself

Questions

- What if two (or more) people want to edit the same file at the same time?
- What if you work on SciNet and on your own computer?

Answers

- Option 1: make them take turns
  - But then only one person can be working at any time
  - And how do you enforce the rule?
- Option 2: patch up differences afterwards
  - Requires a lot of re-working
  - Stuff always gets lost

Collaboration
With others and yourself

Questions

- What if two (or more) people want to edit the same file at the same time?
- What if you work on SciNet and on your own computer?

Answers

- Option 1: make them take turns
  - But then only one person can be working at any time
  - And how do you enforce the rule?
- Option 2: patch up differences afterwards
  - Requires a lot of re-working
  - Stuff always gets lost
- Option 3: **Version Control**

Organize and Track Changes

Question

SciNet

Organize and Track Changes

Question

- Want to undo changes to a file
  - Start work, realize it's the wrong approach, want to get back to starting point
  - Like "undo" in an editor...
    ...but keep the whole history of every file, forever

SciNet

Organize and Track Changes

Question

- Want to undo changes to a file
  - Start work, realize it's the wrong approach, want to get back to starting point
  - Like "undo" in an editor...
    ...but keep the whole history of every file, forever
- Also want to be able to see who changed what, when
  - The best way to find out how something works is often to ask the person who wrote it

SciNet

Organize and Track Changes

Question

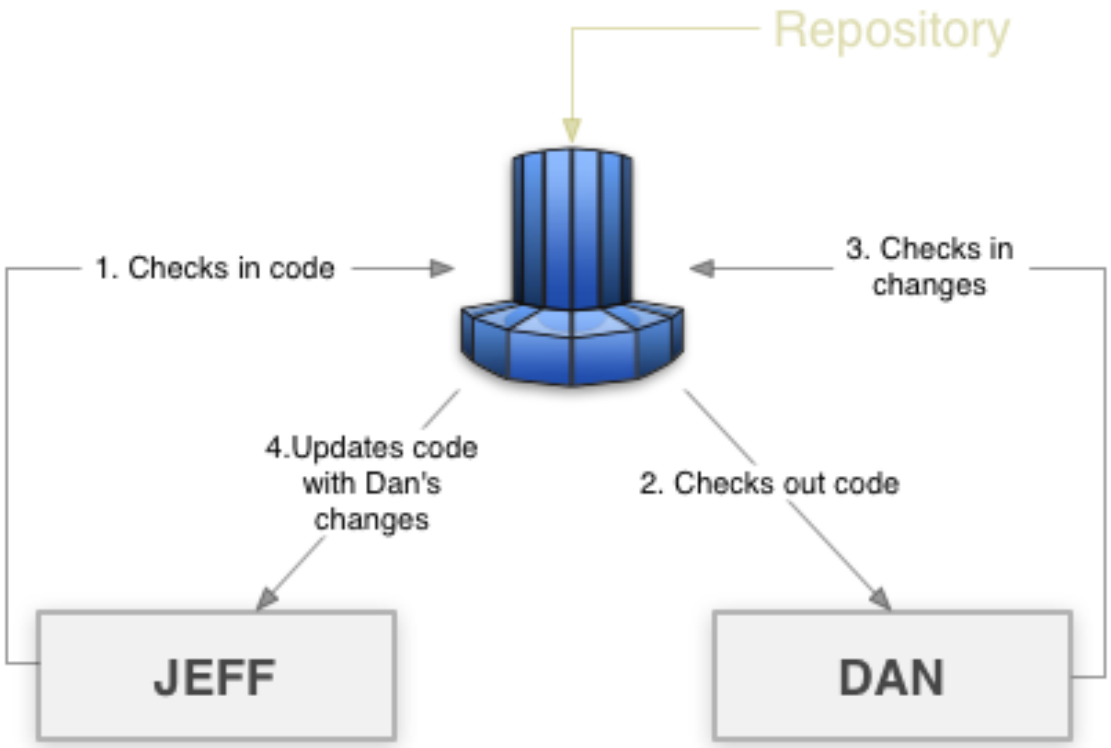
- Want to undo changes to a file
  - Start work, realize it's the wrong approach, want to get back to starting point
  - Like "undo" in an editor...
    ...but keep the whole history of every file, forever
- Also want to be able to see who changed what, when
  - The best way to find out how something works is often to ask the person who wrote it

Answer

- **Version Control**

SciNet

How it Works

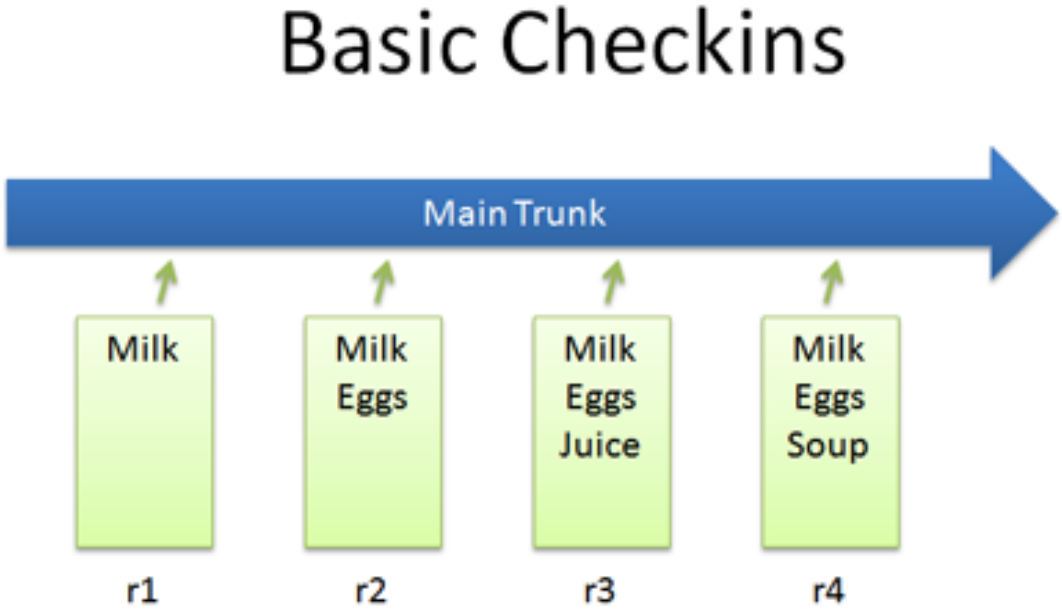


SciNet

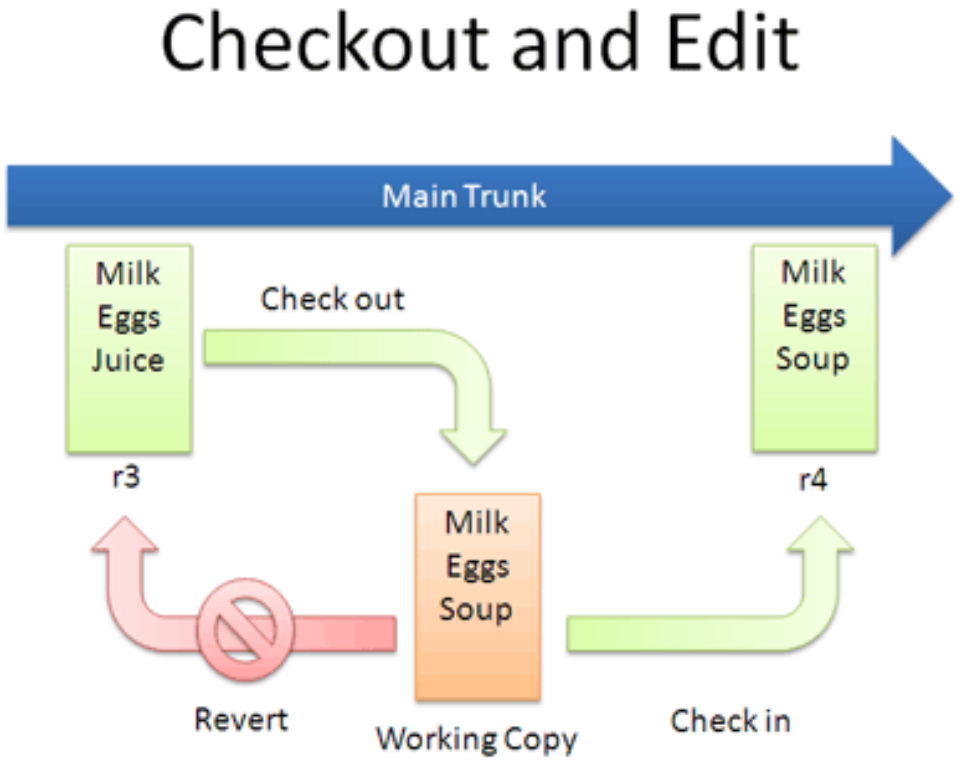
How it Works


Version Control,
Star Trek Style
revision 1
revision 2
branch 2a
branch 2b
Aieee! Roll back! Roll back!

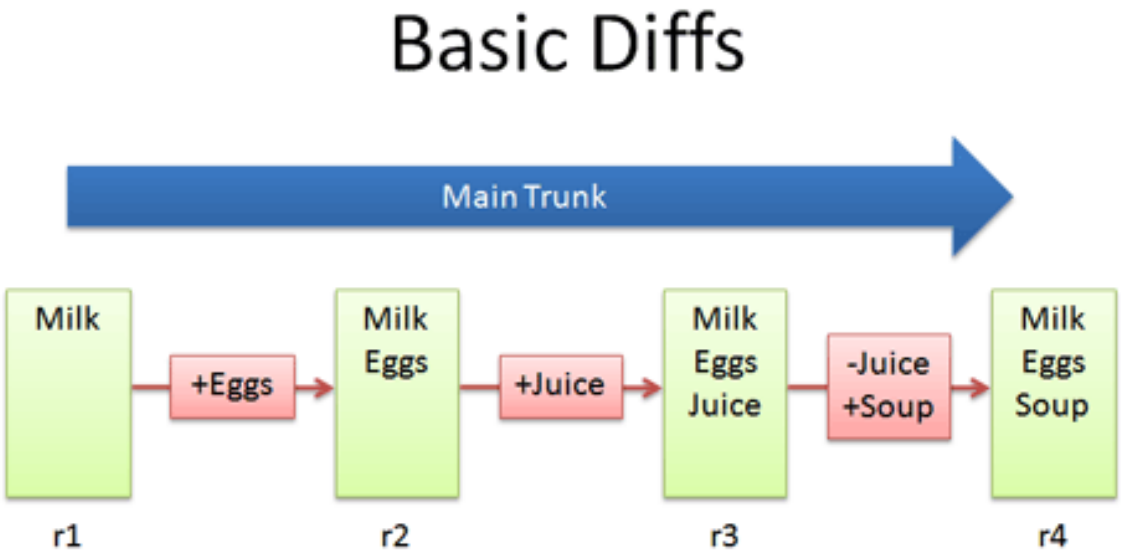
How it Works


Basic Checkins
Main Trunk
Milk
Milk
Eggs
Milk
Eggs
Juice
Milk
Eggs
Soup
r1
r2
r3
r4

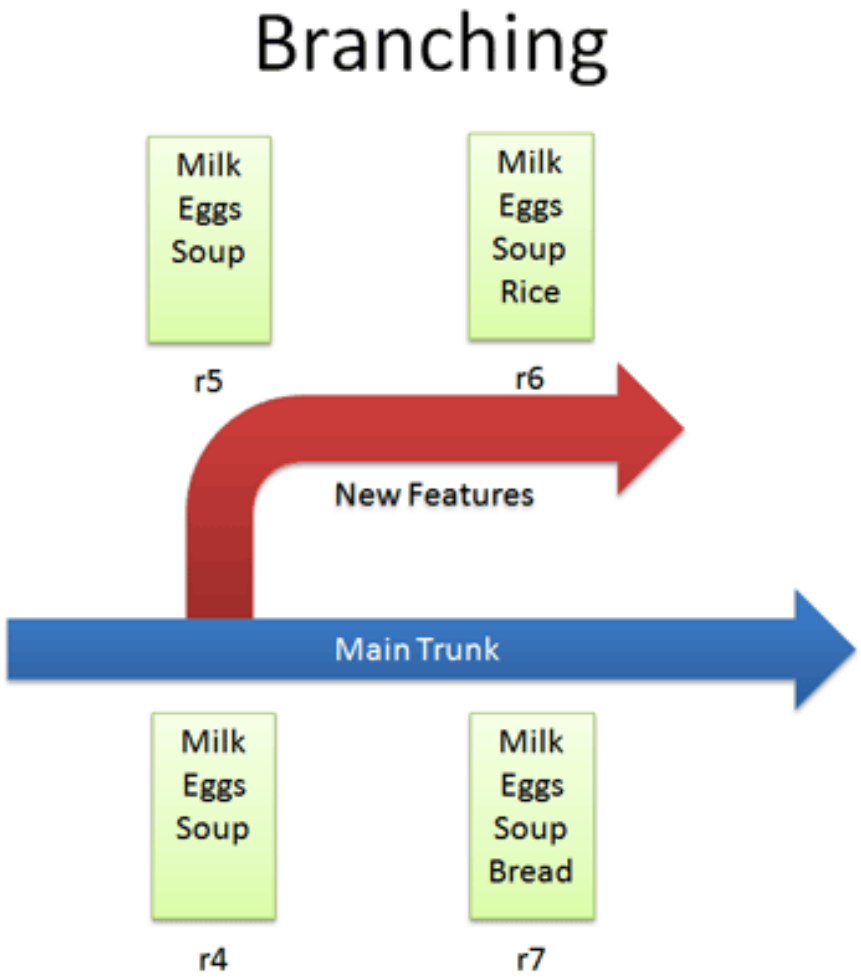
How it Works


Checkout and Edit
Main Trunk
Milk
Eggs
Juice
r3
Check out
Milk
Eggs
Soup
r4
Milk
Eggs
Soup
Working Copy
Revert
Check in

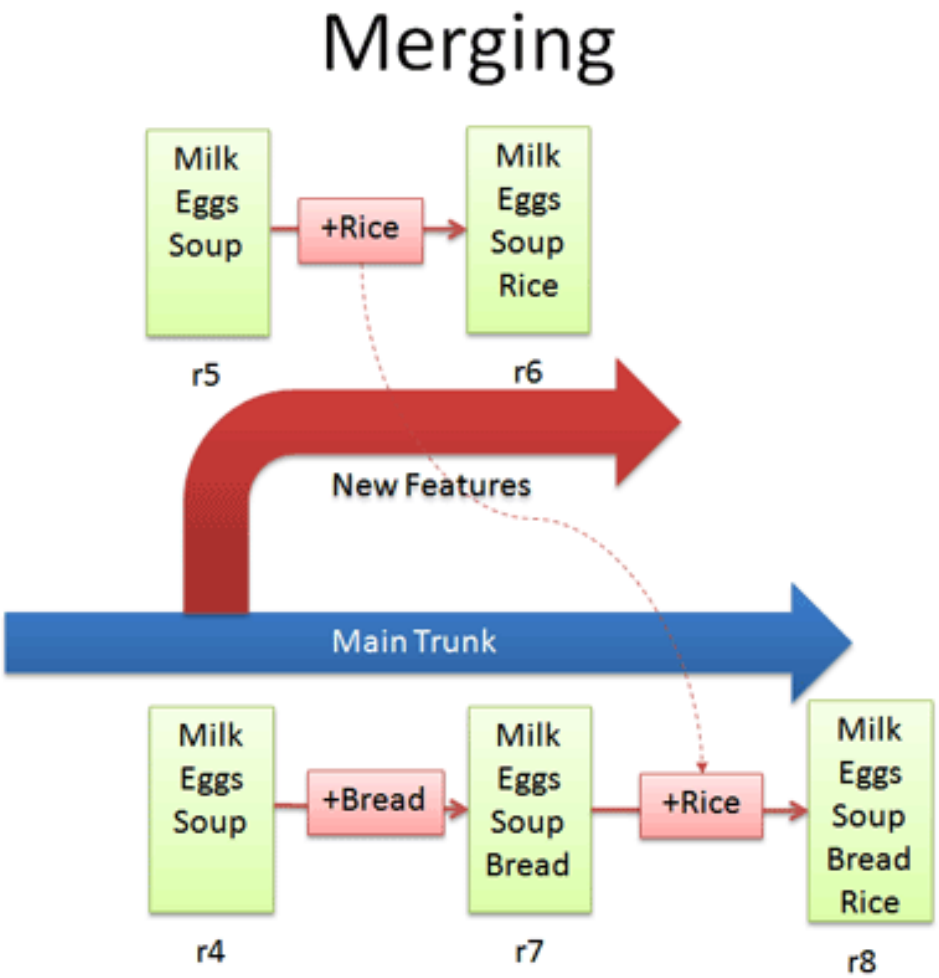
How it Works


Basic Diffs
Main Trunk
Milk
+Eggs
Milk
Eggs
+Juice
Milk
Eggs
Juice
-Juice
+Soup
Milk
Eggs
Soup
r1
r2
r3
r4

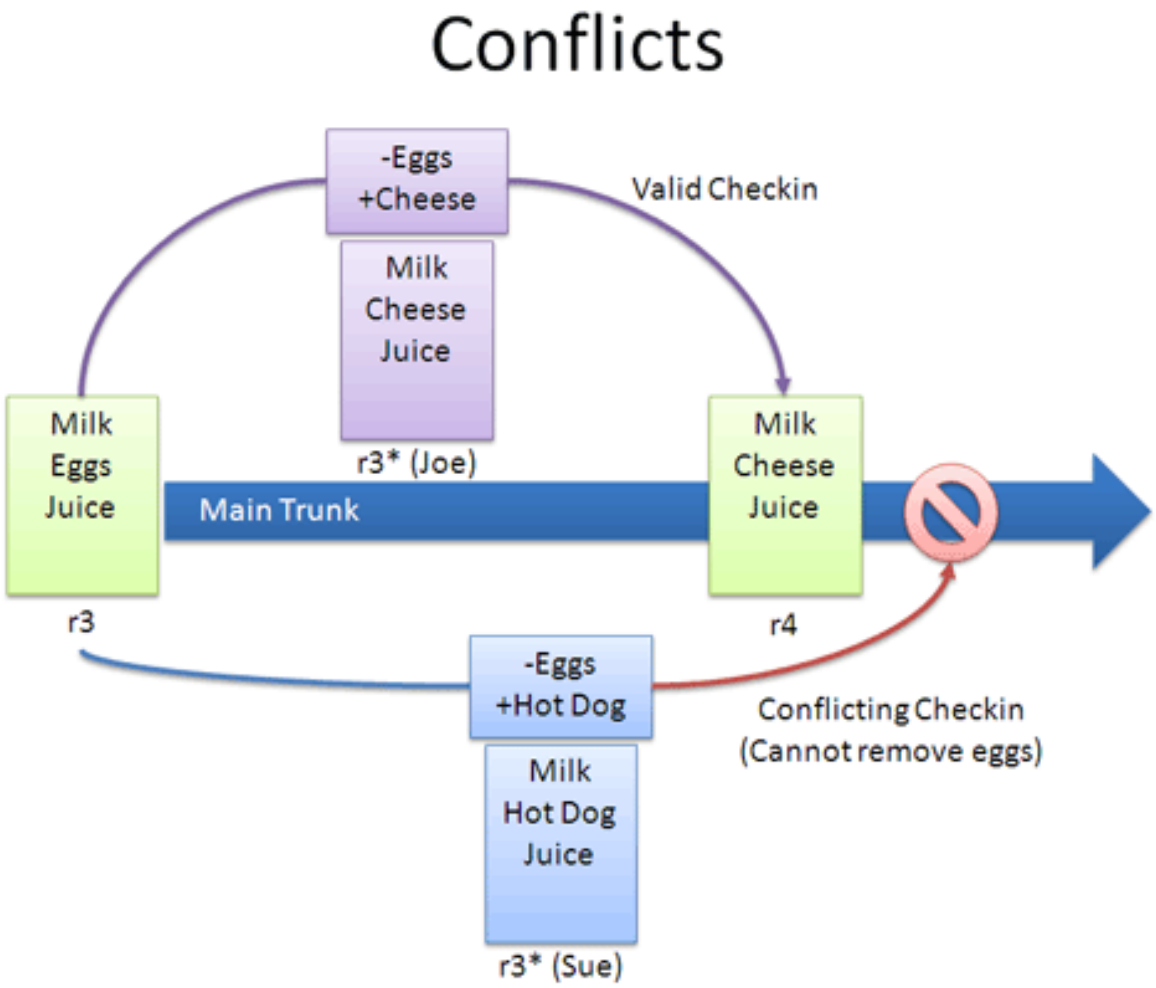
How it Works


Branching
Milk
Eggs
Soup
r5
Milk
Eggs
Soup
Rice
r6
New Features
Main Trunk
Milk
Eggs
Soup
r4
Milk
Eggs
Soup
Bread
r7

How it Works


Merging
Milk
Eggs
Soup
+Rice
Milk
Eggs
Soup
Rice
r5
r6
New Features
Main Trunk
Milk
Eggs
Soup
+Bread
Milk
Eggs
Soup
Bread
+Rice
Milk
Eggs
Soup
Bread
Rice
r4
r7
r8

How it Works



SciNet

How it Works

Resolving Conflicts: Optimistic Concurrency

```
Milk
<<<<<<<
Cheese
=======
Hot Dog
>>>>>>>
Juice
```

SciNet

What to Use

Software

- Open Source
  - Subversion, CVS, RCS
  - Git, Mercurial, Bazaar
- Commercial
  - Perforce, ClearCase

available as modules on SciNet

SciNet

Software

Subversion (svn)

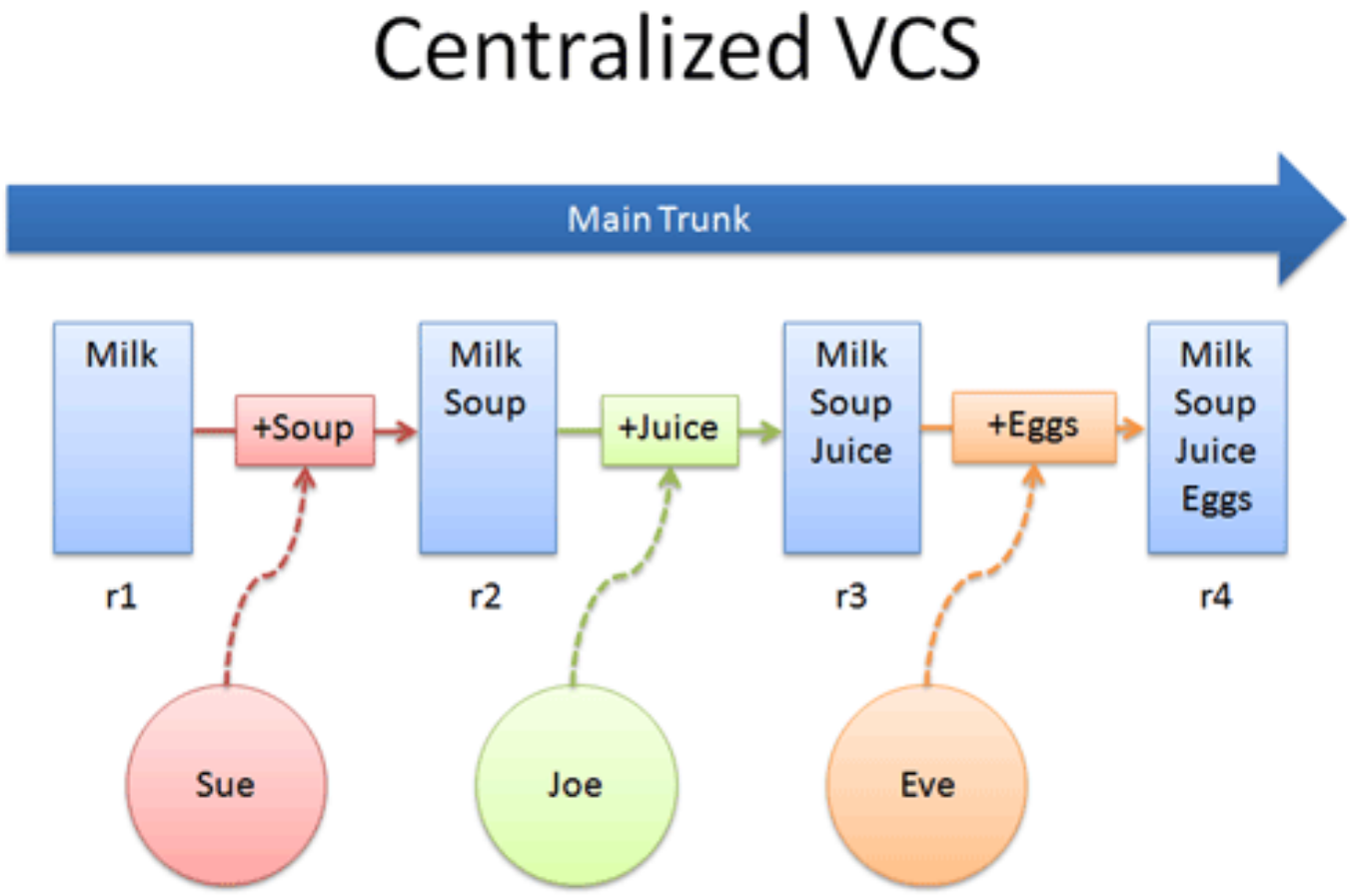
- Centralized Version Control
- Replaces CVS
- Lots of web and GUI integration
- Users: GCC, KDE, FreeBSD

Git

- Distributed Version Control
- *nix command line driven design model
- advanced features `git-stash`, `git-rebase`, `git-cherry-pick`
- Users: Linux kernel, GNOME, Wine, X.org

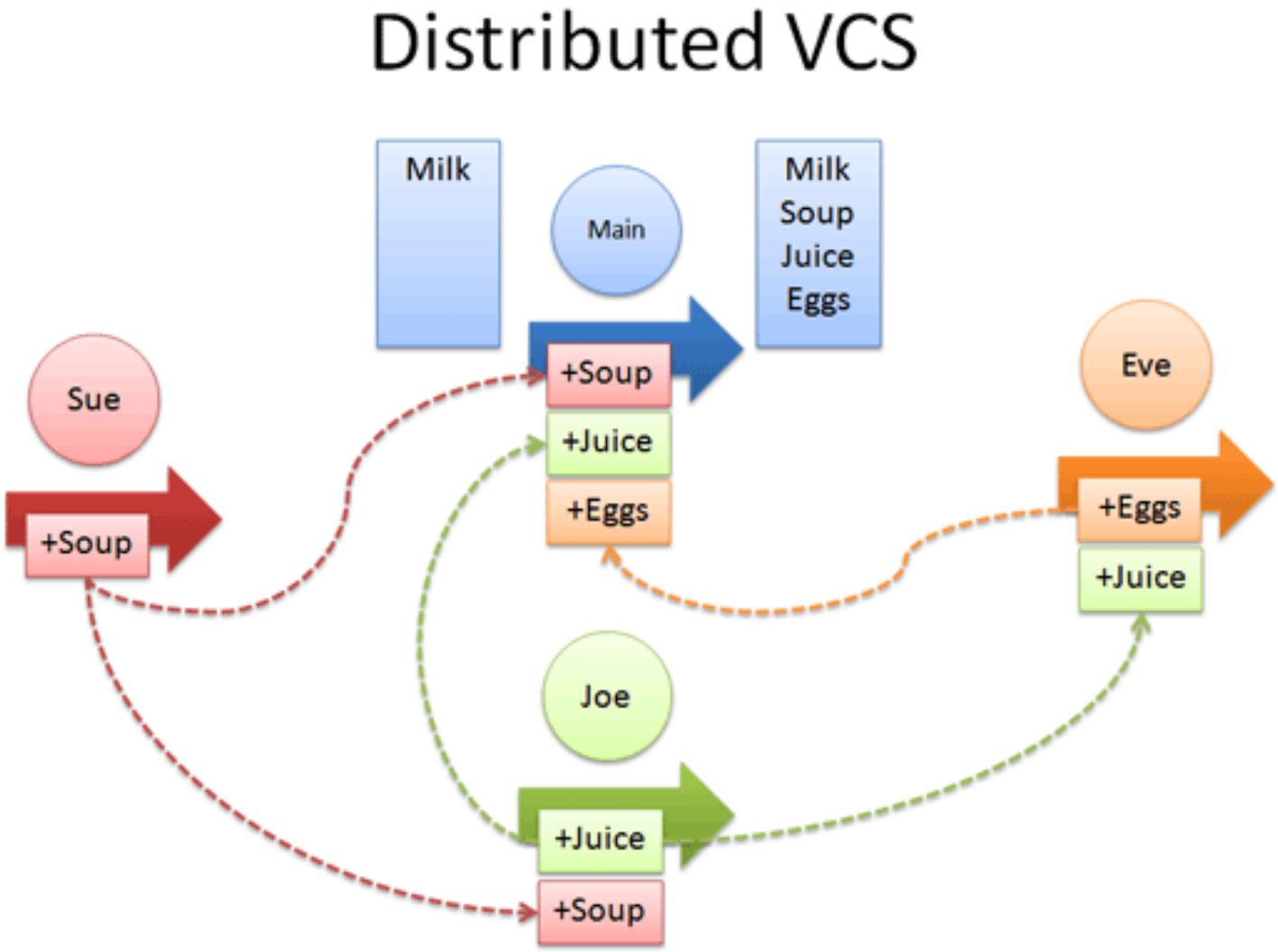
SciNet

Distributed vs. Centralized

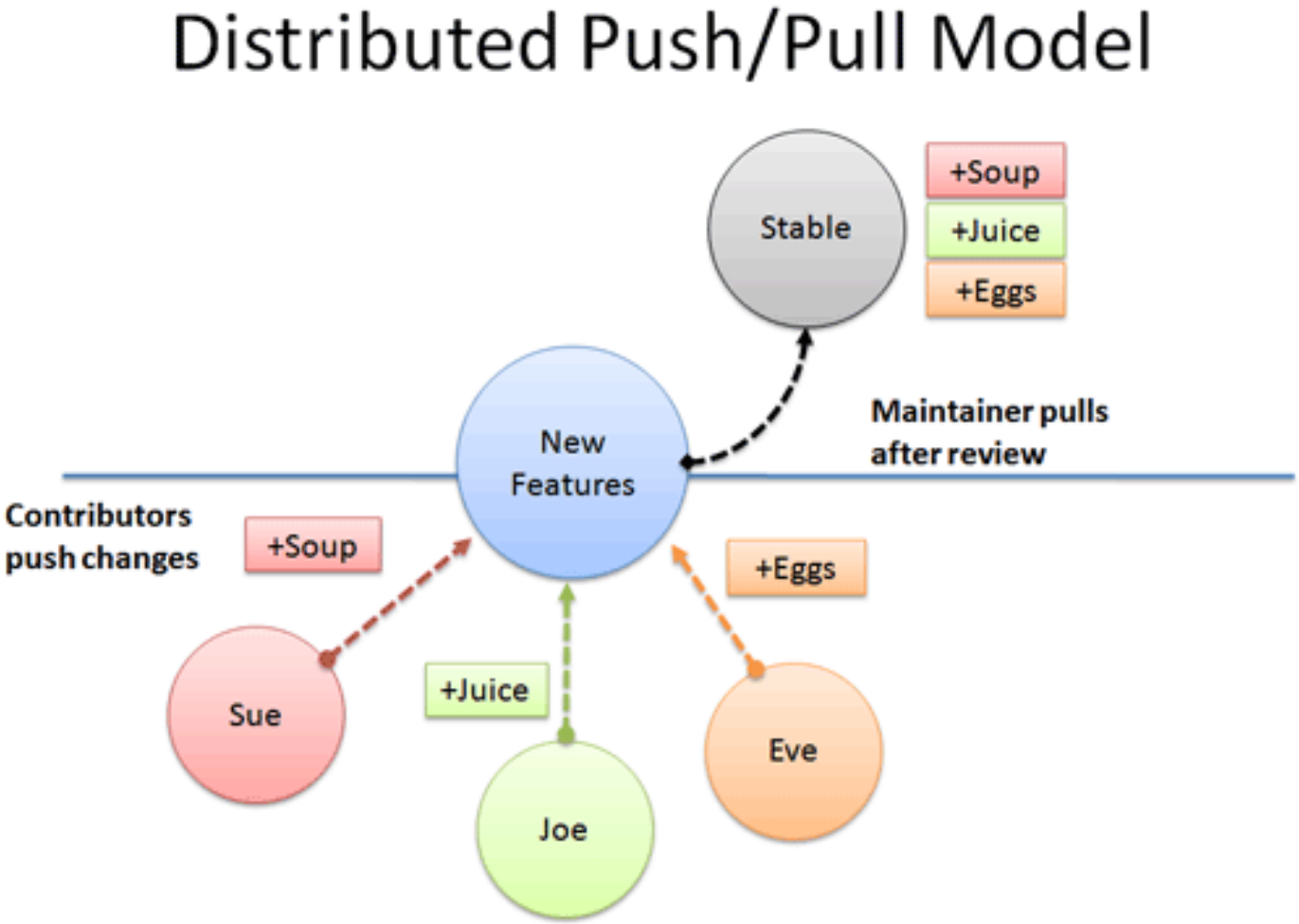


SciNet

Distributed vs. Centralized



Distributed vs. Centralized



Distributed vs. Centralized

Centralized (svn)

- Pros
  - Single Repository
  - Access Controls
  - Predictable Revision Numbers
  - GUI's
  - Simple to understand
- Cons
  - Online to access
  - Typically Slower
  - Merges can be painful

SciNet

Distributed vs. Centralized

Distributed (git)

- Pros
  - Simple setup and lightweight
  - Distributed
  - Very Fast
  - Branch and merging easier
  - Sub collaboration
- Cons
  - Revision numbering
  - Can be complicated conceptually
  - Not backed up

SciNet

New Repo: Subversion

Initialize

```
# svnadmin create  /path/svn
```

Create a Repository

```
# svn import /path/project \
       /path/svn/project/trunk -m 'Initial import'
# svn checkout  /path/project/
```

Make Changes

```
# vi list.txt
# svn ci -m "modified list.txt"
# svn status
```

New Repo: Git

Initialize

```
# git config --global user.name "SciNet User"
# git config --global user.email user@utoronto.ca
```

Create a Repository

```
# cd ~user/code/
# git init
# git add .
# git commit -m "create a git repo of my code"
```

Make Changes

```
# vi list.txt
# git commit -a -m "modified list.txt"
# git status
```

et

Existing Repo: Subversion

Checkout a Project

```
# svn checkout  /path/project/
```

Make Changes

```
# vi list.txt
# svn ci -m "modified list.txt"
# svn status
```

SciNet

Existing Repo: Git

Checkout a Project

```
# cd ~user/code/
# git clone /path/project/
# git checkout master
```

Make Local Changes

```
# vi list.txt
# git commit -a -m "modified list.txt"
```

Publish Changes

```
# git pull (fetch & merge)
# git push
```

References

Links

- Git http://git-scm.com/
- Subversion http://subversion.tigris.org/

SciNet