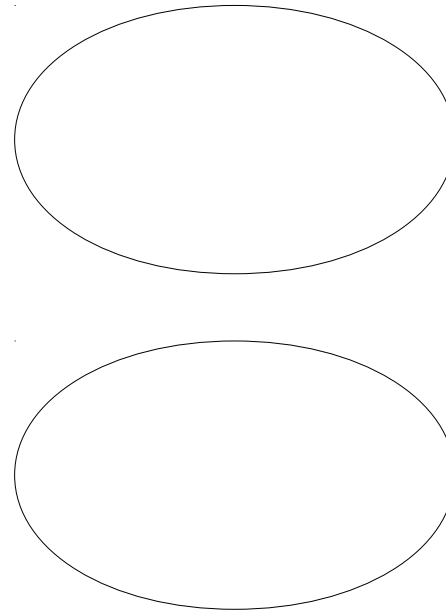
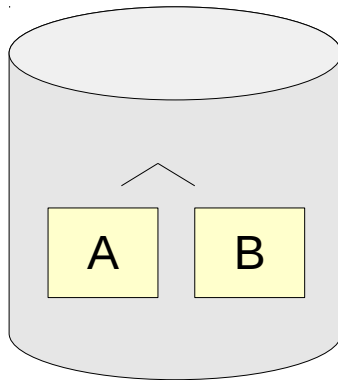
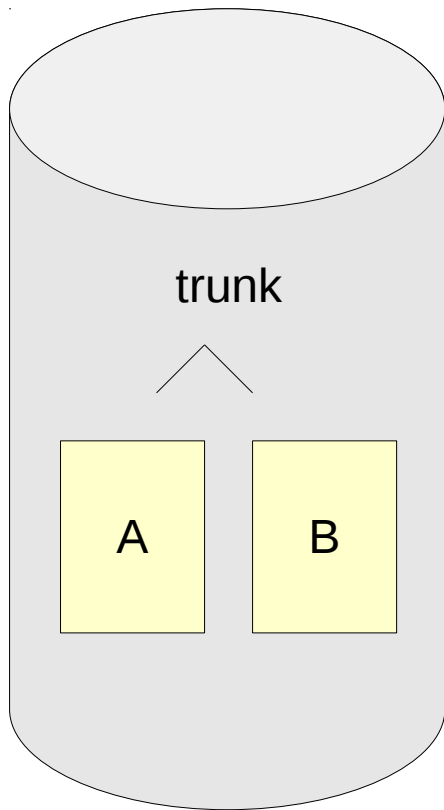


# Subversion Usage Walkthrough

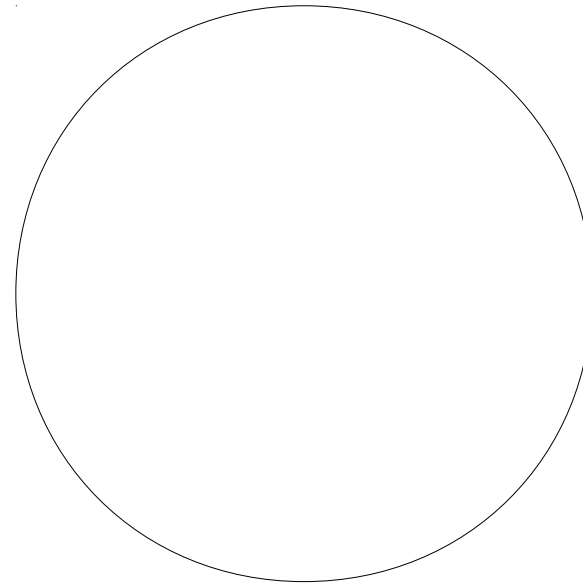
Illustrated conceptual diagram of multiple developers working in parallel from the same repository.



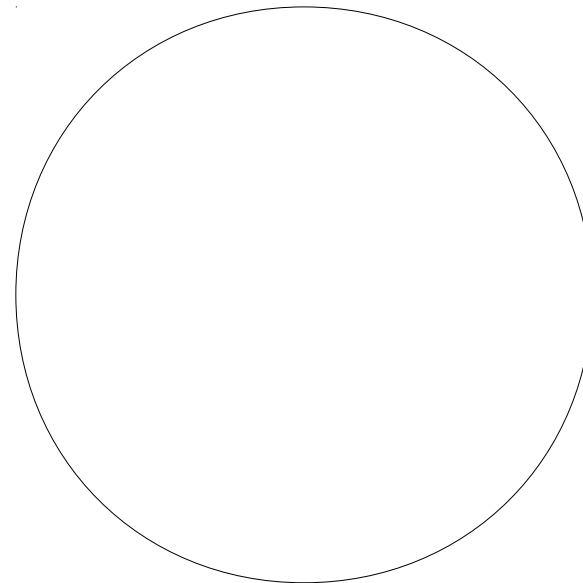


Repository r1

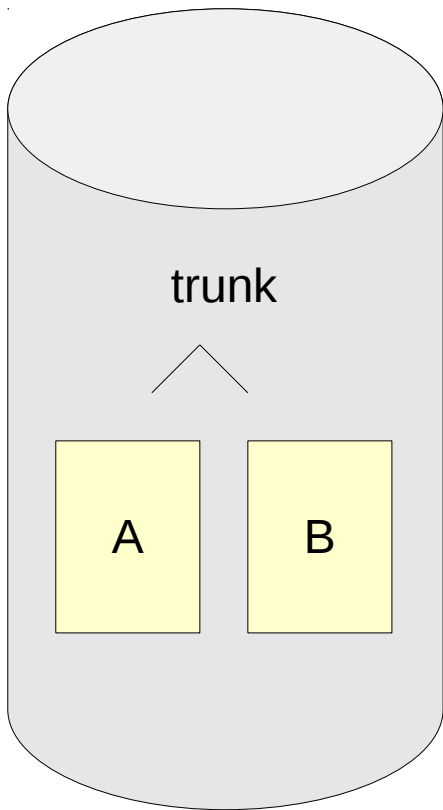
r1 – skeletons for modules A and B are added by the designer, Bob.



Joe's working directory

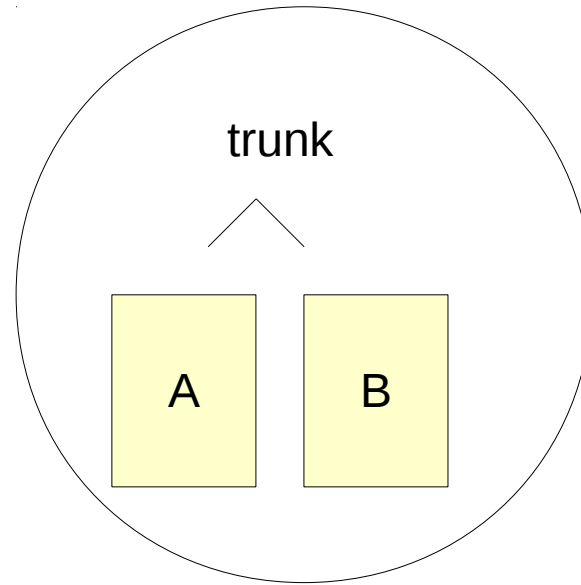


Sue's working directory

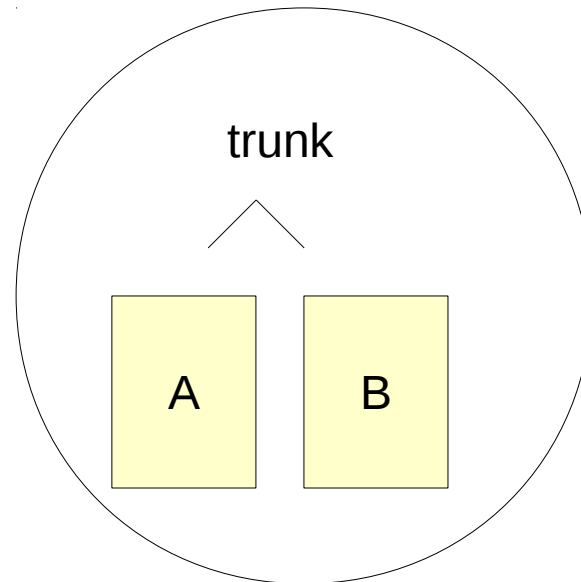


Repository r1

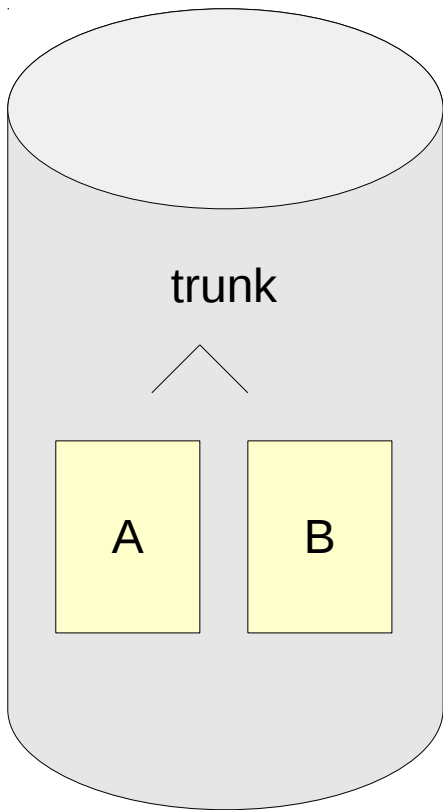
Joe and Sue checkout the repository to their working directories.



Joe's working directory

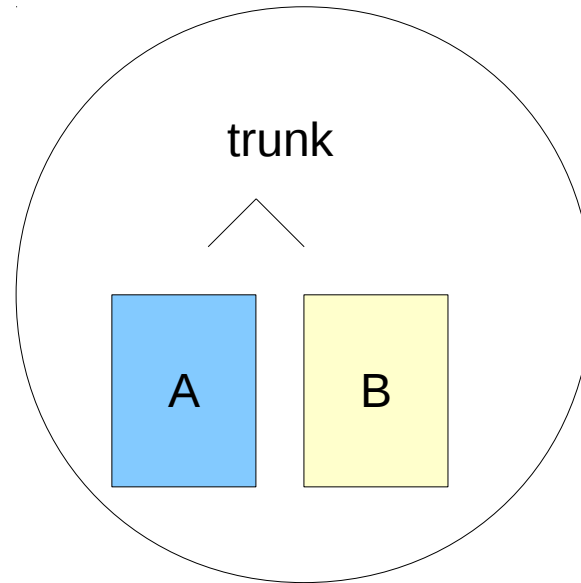


Sue's working directory

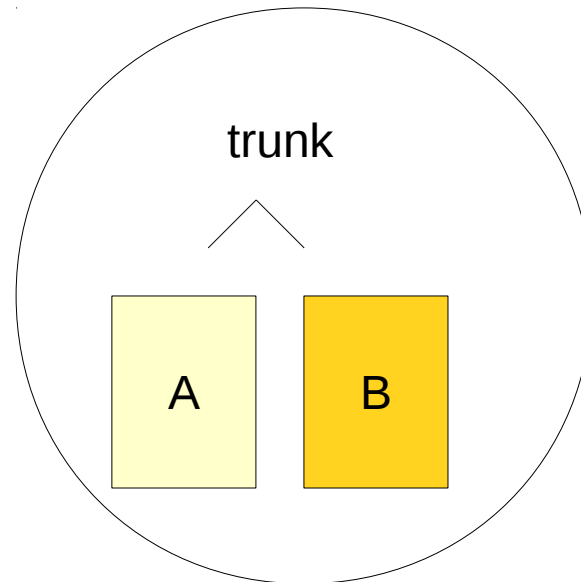


Repository r1

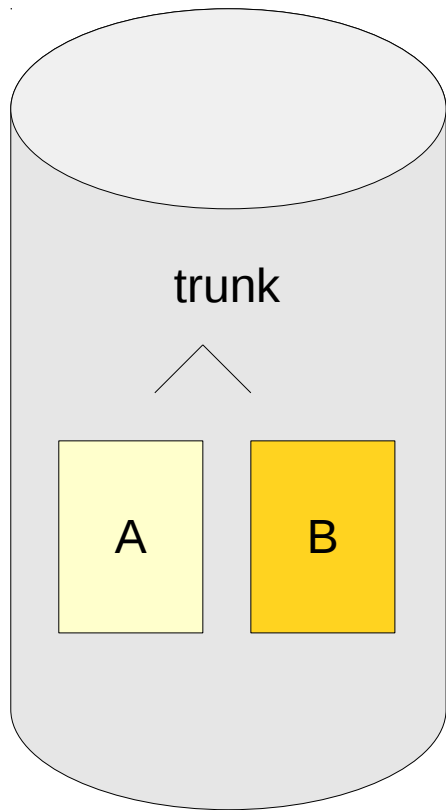
Joe implements module A,  
Sue implements module B.



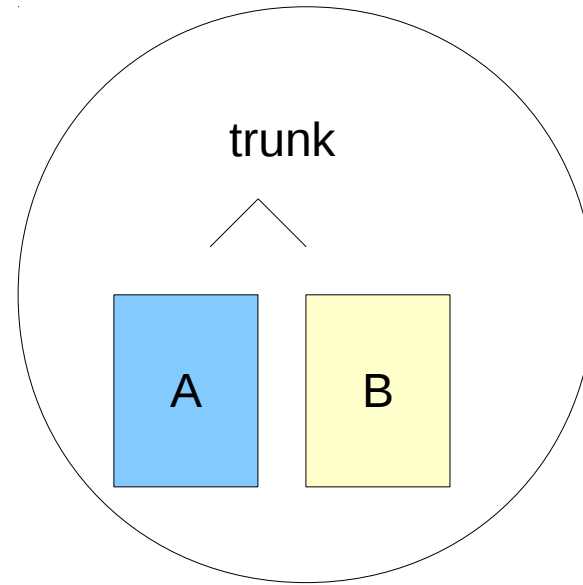
Joe's working directory



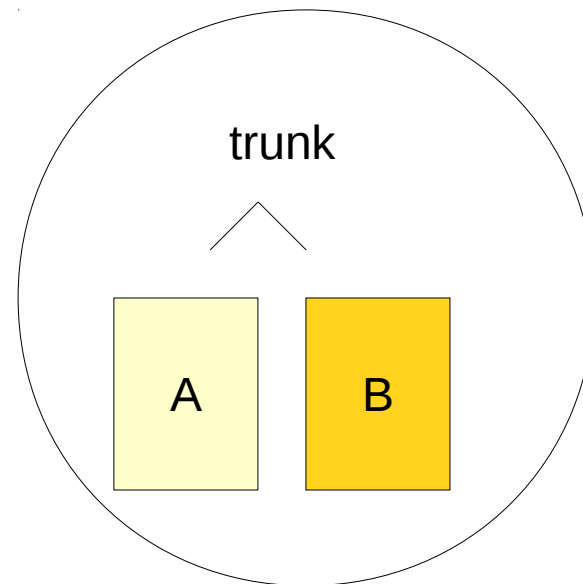
Sue's working directory



Repository r2

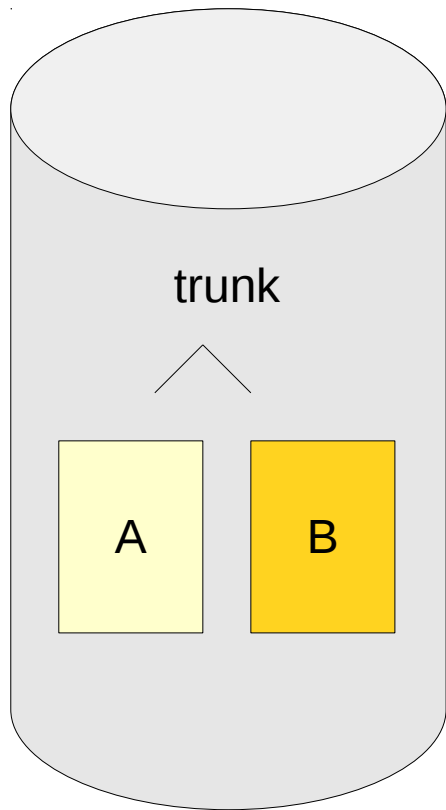


Joe's working directory



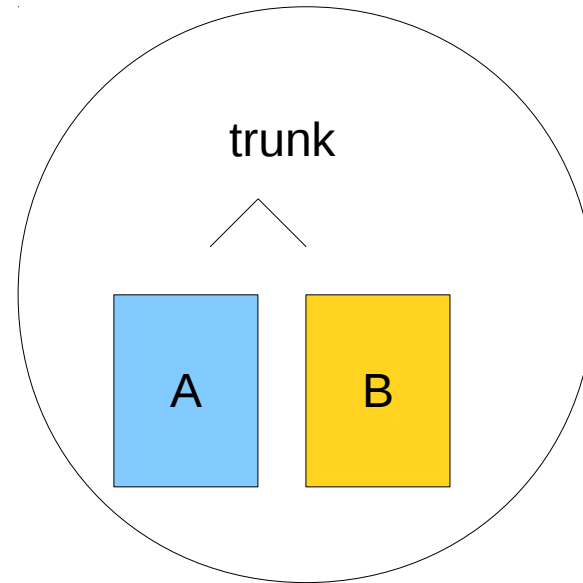
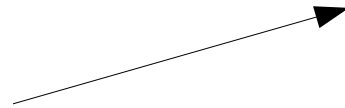
Sue's working directory

Sue updates, then commits module B.  
r2 – module B compiled and unit tested.

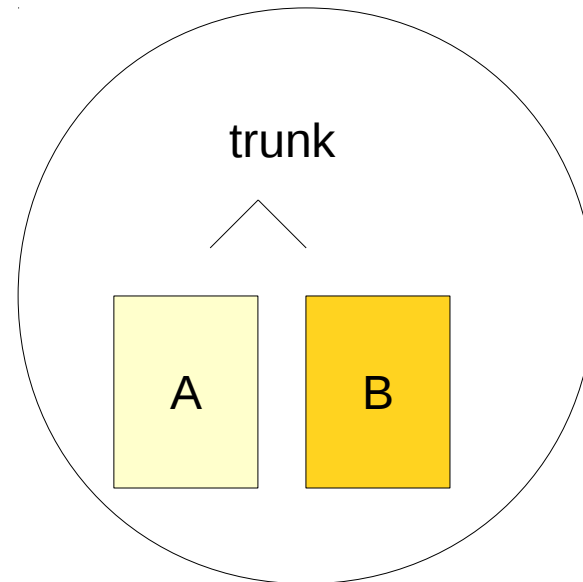


Repository r2

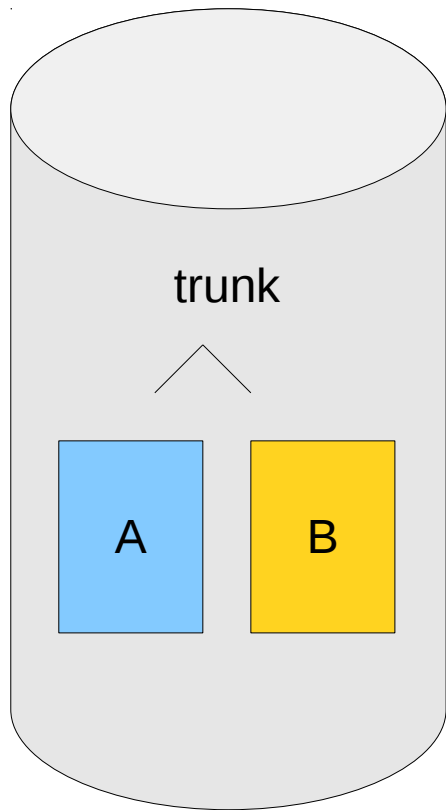
Joe finishes his work and before committing, does an Update, which brings down the most recent changes from the repository. There are no conflicts.



Joe's working directory

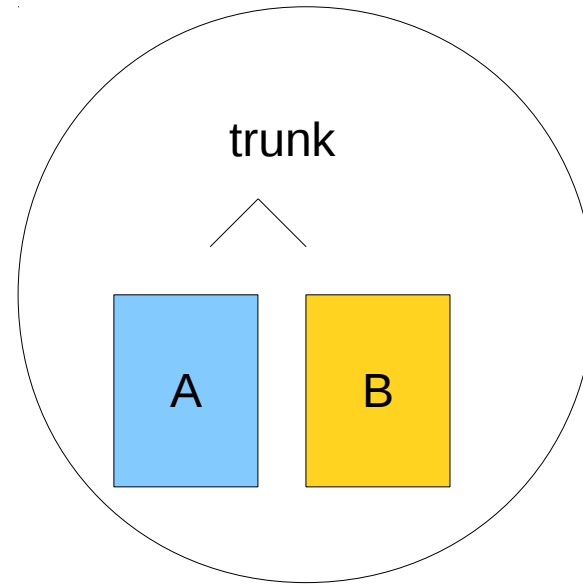


Sue's working directory

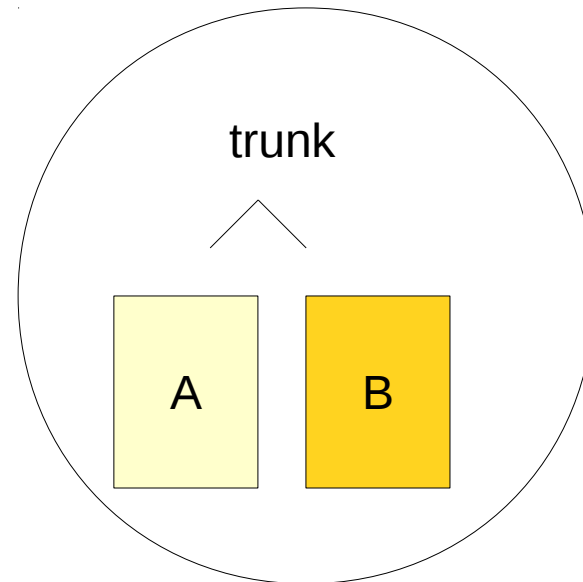


Repository r3

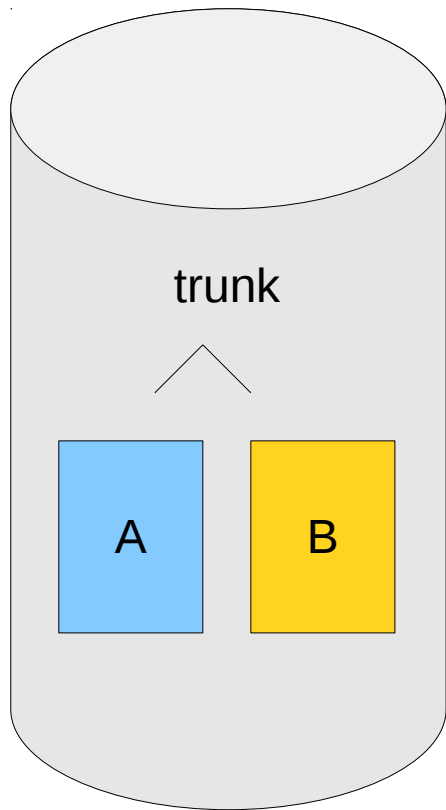
Joe commits module A.  
r3 – module A compiled and unit tested.



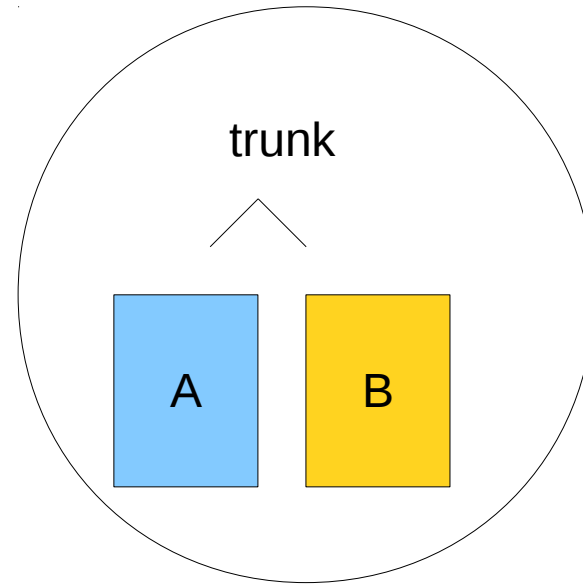
Joe's working directory



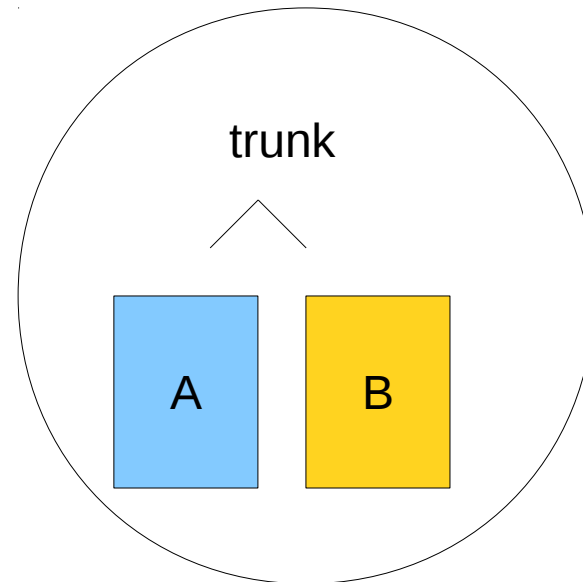
Sue's working directory



Repository r3



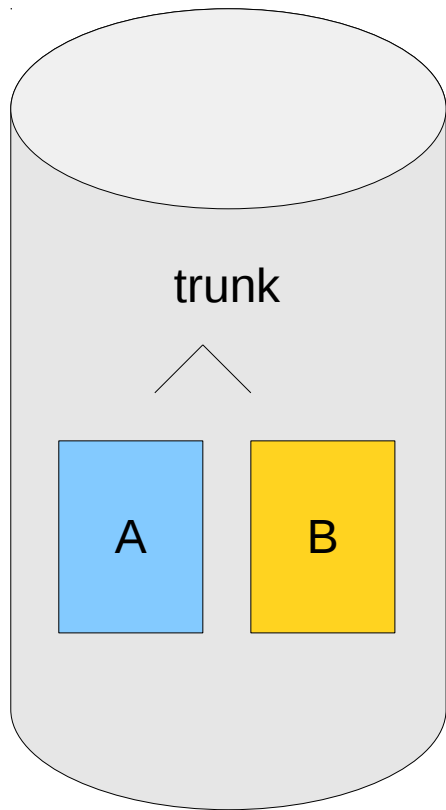
Joe's working directory



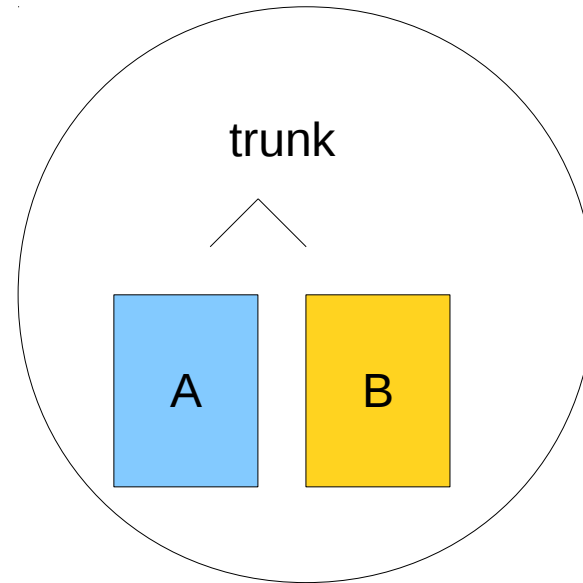
Sue's working directory

When Sue begins her next work session, she starts with an Update to grab the most recent versions from the repository. Now she has Joe's changes.

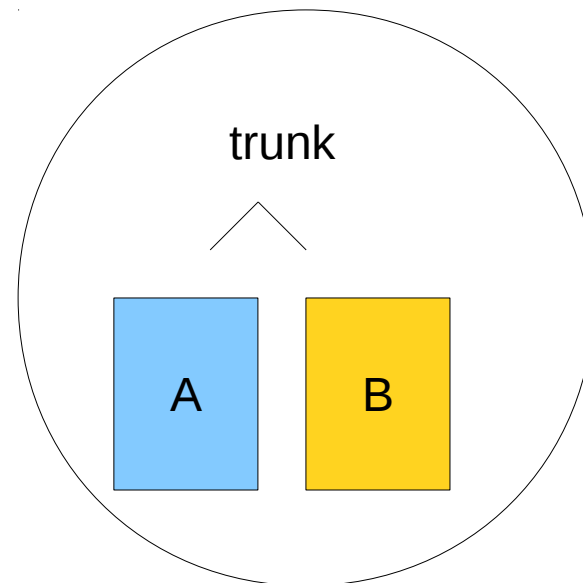




Repository r6

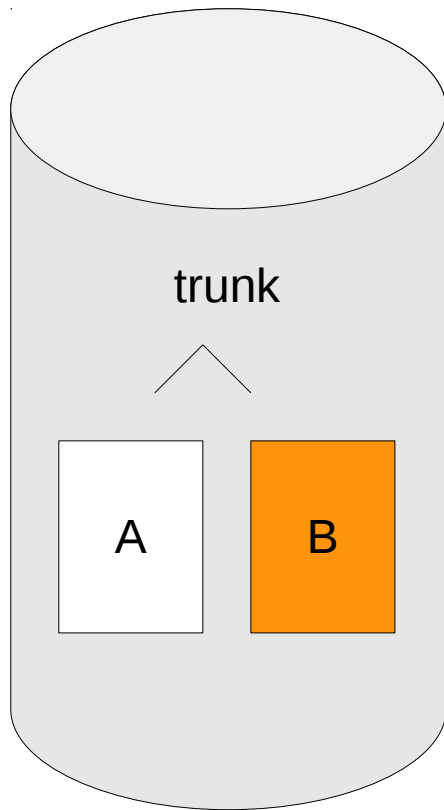


Joe's working directory

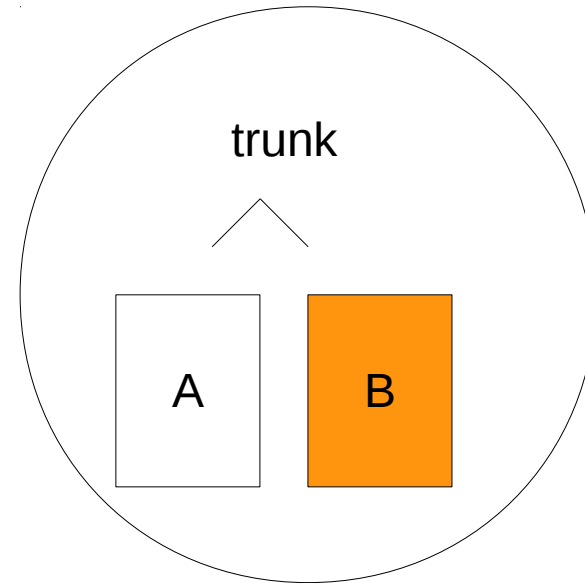


Sue's working directory

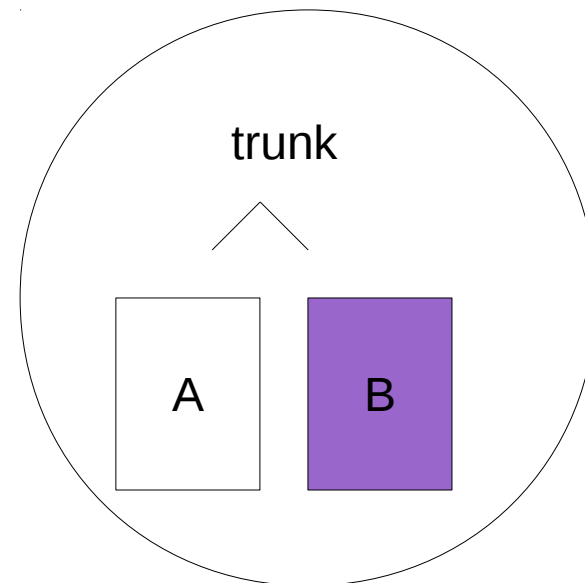
Both developers continue in this manner, maintaining their own modules.  
r4 – module A fixed getTotal() method.  
r5 – module B repaired calculation.  
r6 – module B improved javadoc.



Repository r7



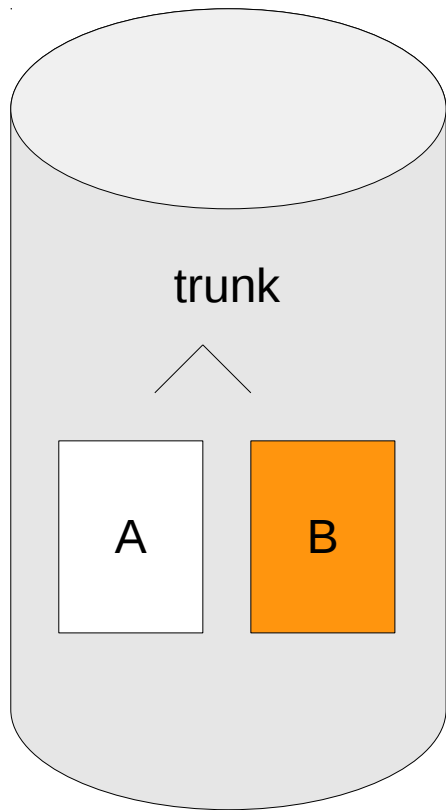
Joe's working directory



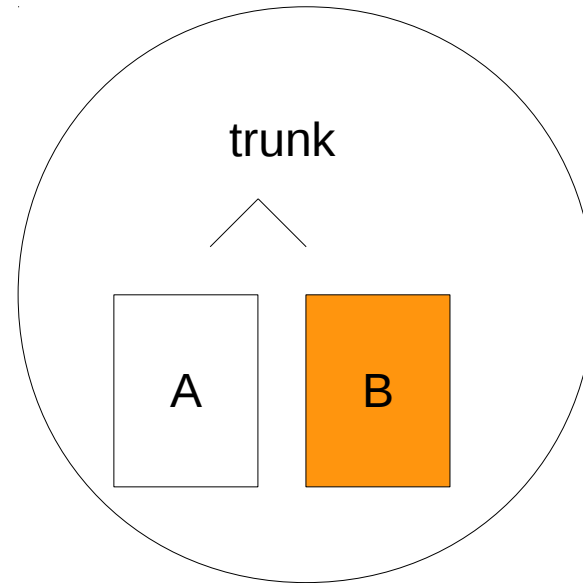
Sue's working directory

While Sue is working on an enhancement (purple), Joe fixes a defect in her code (orange). His repair is located in the same method Sue is modifying.

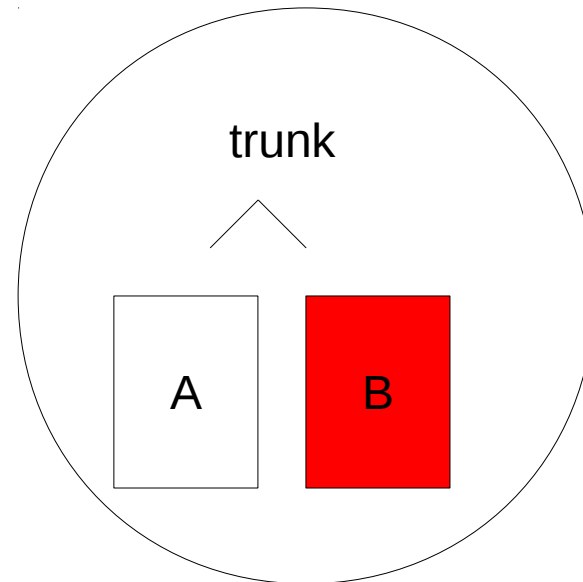
r7 – fix defect in module B.



Repository r7



Joe's working directory



Sue's working directory

Sue finishes her enhancement. When she does Update, SVN warns her of a conflict (red).

Name	Revision	Rep. Rev.	Author ▼	Status	Prop
Ⓜ .	6	2	john.dalbey		
Ⓜ resources	6	2	john.dalbey		
🔍 Card.java.mine				unversioned	
🔍 Card.java.r11				unversioned	
🔍 Card.java.r12				unversioned	
📄 AboutBox.form	6	2	john.dalbey		
📄 AboutBox.java	6	2	john.dalbey		
⚠ Card.java	12	12	john.dalbey	conflicted	
📄 ComputerPlayer.java	6	2	john.dalbey		
📄 ComputerPlayer.java	6	2	john.dalbey		

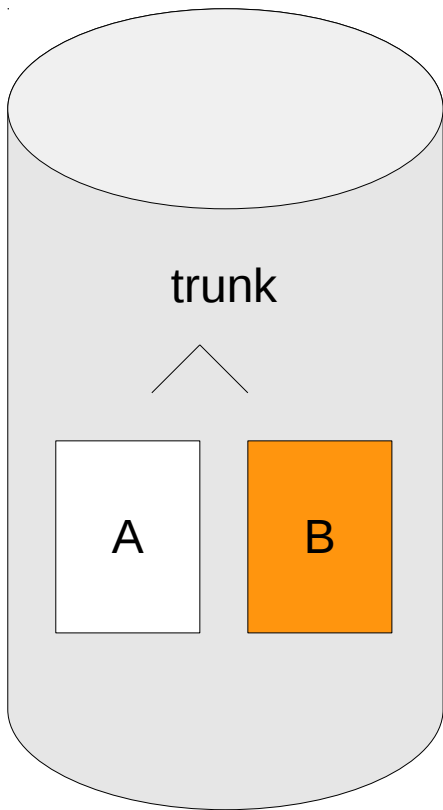
Update conflict as it appears in RapidSVN client.

.r11 is the original version.

.mine has my most recent work.

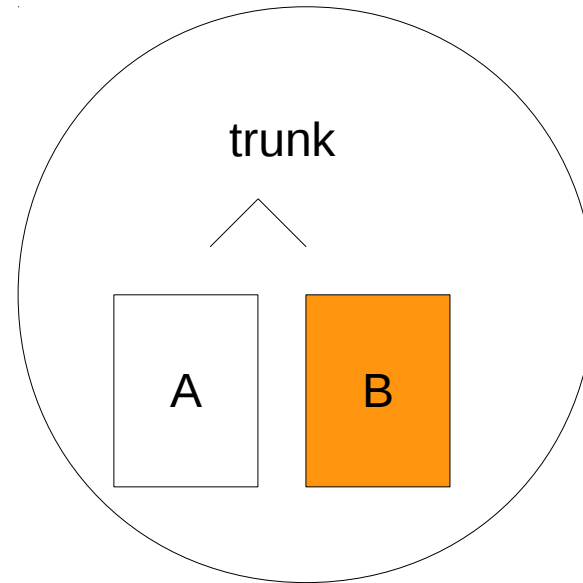
.r12 has someone else's changes that conflict with mine

.java contains annotations for each conflicting line. No longer compilable source. Must be edited to restore to a compilable state by fixing conflicts.

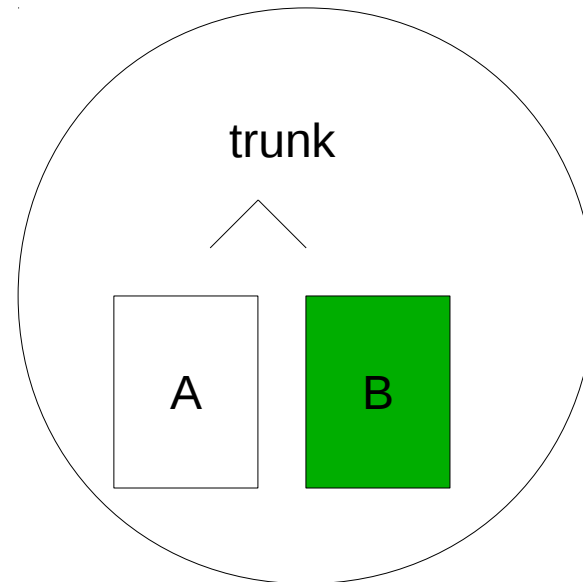


Repository r7

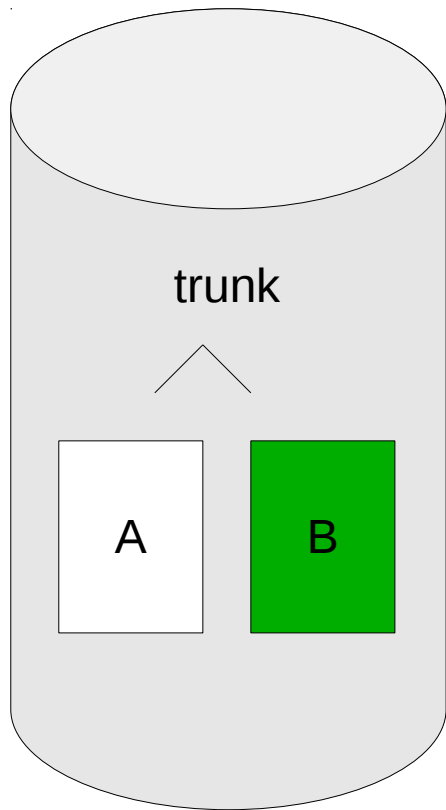
Sue and Joe consult to determine how to merge their changes. Sue edits the file then tells Subversion the conflicts are resolved (green). Joe apologizes and buys Sue lunch.



Joe's working directory



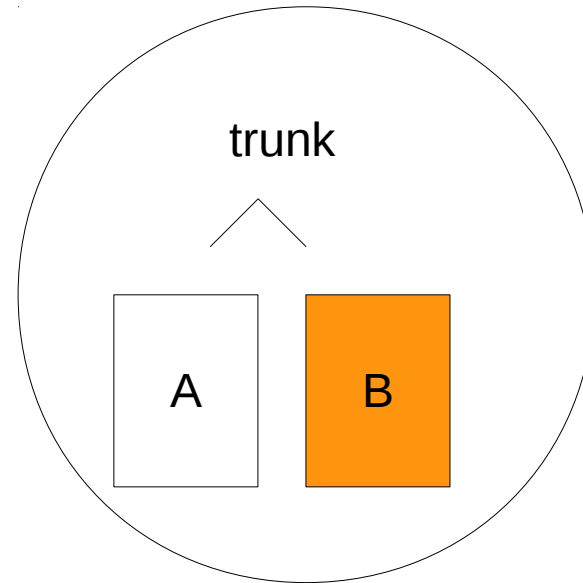
Sue's working directory



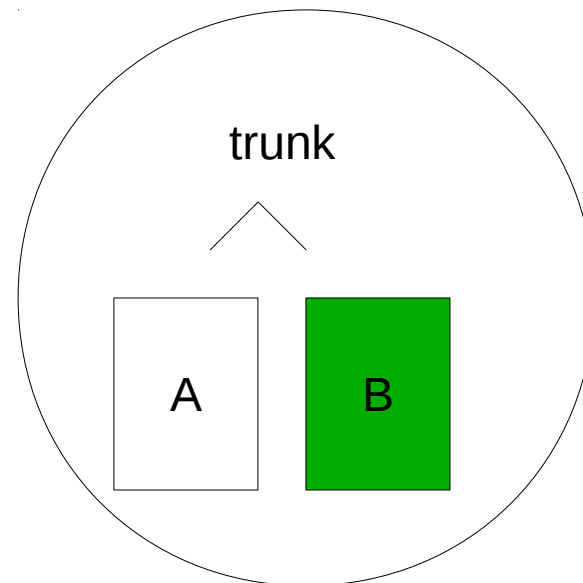
Repository r8

Sue commits her enhancements.  
r8 – enhanced module B.

What should Joe do as he begins his  
next work session?



Joe's working directory



Sue's working directory