# SVNBook clones

1. **9ccaf7f1-848d-4ce5-bfbe-a980da8c9f4a**
   1. 7 раз Select: (p) postpone, (df) diff-full, (e) edit, (mc) mine-conflict, (tc) theirs-conflict, (s) show all options:
2. **ecb4a7f2-5d8b-42b5-8c7f-2f6d7bd7dc20**
   1. 3 раза Select: (p) postpone, (df) diff-full, (e) edit, (m) merge, (mc) mine-conflict, (tc) theirs-conflict, (s) show all options:
3. **b6c62b25-31e2-4f4f-8fcc-e55c8a5bd022**
   1. 8 раз svn:auto-props \*.py = svn:eol-style=native \*.c = svn:eol-style=native \*.h = svn:eol-style=native
4. **36b1eddf-494c-49f5-b7c9-d180eb3fa50f**
   1. +The idea behind this project is to come up with a piece of  
      +client-server software that can remotely frab a naggily wort.  
      +Frabbing naggily worts is tricky business, and doing it incorrectly  
      +can have serious ramifications, so we need to employ over-the-top  
      +input validation and data verification mechanisms.
   2. The idea behind this project is to come up with a piece of  
      client-server software that can remotely frab a naggily wort.  
      Frabbing naggily worts is tricky business, and doing it incorrectly  
      can have serious ramifications, so we need to employ over-the-top  
      input validation and data verification mechanisms.
   3. 2 раза   
      The idea behind this project is to come up with a piece of  
      client-server software that can remotely frab a naggily wort.  
      Frabbing naggily worts is tricky business, and doing it incorrectly  
      can have serious ramifications, so we need to employ over-the-top  
      input validation and data verification mechanisms.
5. **2b8af518-b034-4497-ac5a-aa3ad3292793**
   1. 4 раза --config-dir DIR Instructs Subversion to read configuration information from the specified directory instead of the default location ( .subversion in the user&#39;s home directory).
   2. --config-dir DIR Read configuration information from the specified directory instead of the default location ( .subversion in the user&#39;s home directory).
6. **26e0a8b2-4887-4a9f-9ec7-24332a65e540**
   1. Version Control with Subversion [DRAFT] For Subversion 1.8 (Compiled from ) Ben Collins-Sussman Brian W. Fitzpatrick C. Michael Pilato Tatiana Apandi 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 Ben Collins-Sussman Brian W. Fitzpatrick C. Michael Pilato This work is licensed under the Creative Commons Attribution License. To view a copy of this license, visit or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.
   2. Copyright (c) 2002-2013 Ben Collins-Sussman, Brian W. Fitzpatrick, C. Michael Pilato. This work is licensed under the Creative Commons Attribution License. To view a copy of this license, visit or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.
7. **b00d1bc3-6d24-4691-9e4b-db7757a6b8e6**
   1. svn:sync-from-uuid Contains the UUID of the repository of which this repository has been initialized as a mirror by the svnsync tool. (This property is only relevant when it appears on revision 0.)
   2. svn:sync-from-url Contains the URL of the repository directory of which this repository has been initialized as a mirror by the svnsync tool. (This property is only relevant when it appears on revision 0.)
   3. svn:sync-currently-copying Contains the revision number from the source repository which is currently being mirrored to this one by the svnsync tool. (This property is only relevant when it appears on revision 0.)
   4. svn:sync-last-merged-rev Contains the revision of the source repository which was most recently and successfully mirrored to this one. (This property is only relevant when it appears on revision 0.)
8. **5371b479-35f5-4e1d-9c4d-f844e4e57794**
   1. As you can see, svn merge has merged the requisite changes into our working copy, flagging a conflict where the vendor modified the same region of one of the files as we did during our customizations. Subversion detects this conflict, and gives us the opportunity to resolve it (using the methods described in) so that our customizations to what is now libcomplex 1.0.1 continue to make sense. Once we&#39;ve resolved the conflicts and performed any testing or review we need, we can commit the changes to our vendor branch.
   2. As you can see, svn merge has merged the changes required to make libcomplex 1.0.0 look like libcomplex 1.0.1 into our working copy. In our example, it has even noticed and flagged a conflict on one file. It seems the vendor modified a region of one of the files we also customized. Subversion safely detects this conflict, and gives us the opportunity to resolve it so that our customizations to what is now libcomplex 1.0.1 continue to make sense. (See for more on resolving conflicts of this sort.) Once we&#39;ve resolved the conflicts and performed any testing or review we need, we can commit the changes to our vendor branch.
9. **4315caba-8a72-4f84-998d-885ebb99dd86**
   1. When creating a new repository, use a format that is compatible with versions of Subversion earlier than Subversion 1.4. --pre-1.5-compatible Deprecated . See option --compatible-version .
   2. When creating a new repository, use a format that is compatible with versions of Subversion earlier than Subversion 1.5. --pre-1.6-compatible Deprecated . See option --compatible-version .
   3. When creating a new repository, use a format that is compatible with versions of Subversion earlier than Subversion 1.6.
10. **86601698-b1e1-4572-82bf-888f9433e234**
    1. FILE and SECTION are the runtime configuration file (either config or servers ) and the section thereof, respectively, which contain the option whose value you wish to change. OPTION is, of course, the option itself, and VALUE the value (if any) you wish to assign to the option. For example, to temporarily disable the use of the automatic property setting feature, use --config-option=servers:global:http-library=serf . You can use this option multiple times to change multiple option values simultaneously.
    2. In this syntax, FILE and SECTION are the runtime configuration file (either config or servers ) and the section thereof, respectively, which contain the option whose value you wish to change. OPTION is, of course, the option itself, and VALUE the value (if any) you wish to assign to the option. For example, to temporarily disable the use of the automatic property setting feature, use --config-option=config:miscellany:enable-auto-props=no . You can use this option multiple times to change multiple option values simultaneously.
    3. In this syntax, FILE and SECTION are the runtime configuration file (either config or servers ) and the section thereof, respectively, which contain the option whose value you wish to change. OPTION is, of course, the option itself, and VALUE the value (if any) you wish to assign to the option. For example, to temporarily disable the use of the automatic property setting feature, use --config-option=servers:global:http-library=serf . You can use this option multiple times to change multiple option values simultaneously.
    4. In this syntax, FILE and SECTION are the runtime configuration file (either config or servers ) and the section thereof, respectively, which contain the option whose value you wish to change. OPTION is, of course, the option itself, and VALUE the value (if any) you wish to assign to the option. For example, to temporarily disable the use of the automatic property setting feature, use --config-option=config:miscellany:enable-auto-props=no . You can use this option multiple times to change multiple option values simultaneously.