



bbs.ChinaUnix.net

首页 | 论坛 [微博](#) [博客](#) [空间](#) [读书](#) [人才](#) [自测](#) | 新闻 [Linux](#) [AIX](#) [Windows](#) [存储](#) [网络](#) | [Wiki](#) [资料](#) [手册](#) [下载](#) | [搜索](#)

2011系统架构师大会 隆重召开 欢迎报名
1000人规模，近50位专家分享 >>更多请点击



免费注册 | 忘记密码 | 会员登录 | 搜索 | 帮助

[ChinaUnix社区PHP技术征文大赛](#) | [2011年度CU网友发帖月月奖奖励计划](#) | [9.9北京：2011系统架构师大会隆重推出！！](#) | [买域名送域名 海外主机免备案](#)

首页 » 论坛 » Linux文档专区 »

SVN vs CVS

发表主题

回复帖子

[收藏] [本帖文本页] [推荐此主题给朋友]

版块跳转

最近访问的版块

阅读 333 回复 0

分享到:

pascal4123 (Fowler)



精灵王
☆☆☆☆

帖子 359

主题 353

精华 0

可用积分 302

专家积分 0 (本版:0)

来自 上海

在线时间 1 小时

注册时间 2006-03-30

最后登录 2009-06-10

状态: ...当前离线...

[\[微博\]](#) [\[博客\]](#) [\[短信\]](#)

1 楼 发表于 2009-09-19 19:45 | 只看该作者

[倒序看帖]

TOP

版本控制系统里团队开发不免要用上CVS SVN ClearCase等工具。至于选择上，则是根据开发团队搭建的平台，使用的编程语言相关联。

如下节选一些网上的对比说明。当然，真正要弄懂这些版本控制系统，还是要花费巨大工夫学习研究，不可能google几下就能完成的。

一、Subversion包含绝大部分CVS功能

Subversion 作为CVS 的重写版和改进版，其目标就是作为一个更好的版本控制软件，取代目前流行的CVS。Subversion 的主要开发人员都是业界知名的CVS 专家。Subversion支持绝大部分的CVS 功能/命令；Subversion的命令风格和界面也与CVS 非常接近。当然，不同的地方正是对CVS 的改进。

二、全局性的版本编号

一个新的版本，并得到一个自增量的版本号N+1，该版本号并不针对某个特定的文件，而是全局性的、针对整个版本库的。因此，我们可以将Subversion 的版本库看作是一个文件系统或文件目录树的数组。

从技术的角度来说，在Subversion 中，"文件foo.c 的第5 版本"这个说法是错误的；正确的说法应该是："文件foo.c 在版本库被修改了5 次，即执行5 次commit 后是什么样子？"。显然，在Subversion 中，版本库被修改5 次后foo.c 的内容，和被修改了6 次后foo.c 的内容很可能完全一样，因为版本库的第6 次修改很可能只修改了版本库的其他部分，而并没有对foo.c 的进行修改。相反，在CVS 中，文件foo.c 的第1.1 版本和第1.2 版本总是不同的。

Subversion 的全局性版本编号为Subversion 带来了诸多的优势：如对目录或文件执行拷贝，无论涉及多少文件，Subversion 不需要对单个文件依次执行拷贝命令，仅仅需要建立一个指向相应的全局版本号的一个指针即可。

三、目录的版本控制

CVS 只能对文件进行版本控制，不能对目录进行版本控制，因此CVS 没有任何关于文件"移动"（move）操作的概念。当人为进行文件移动操作时，CVS 只能注意到，一个文件在一个位置被删除了，而在一个新位置创建了另外一个文件。由于它不会连接两个操作，因此也很容易使文件历史轨迹丢失。设置 CVS 存储库时，必须非常谨慎地为每个文件选择准确的位置，因为在设置之后，几乎就要一直使用这个位置了。

同样由于CVS 不记录目录的版本历史，CVS 不支持对文件的"重命名"（rename），人为的对文件进行重命名会使得命名前后的文件失去历史联系，而记录历史本来是版本管理的主要目的。

还有，CVS 不支持对文件的"拷贝"（copy），人为的拷贝对CVS 而言，只能看到新的文件的增加，而不能记录拷贝源文件和目标文件之间的联系。

综上所述，缺乏对文件"移动"、"重命名"、"拷贝"的支持的根源在于CVS 不能记录目录的版本历史，而这些操

http://bbs.chinaunix.net/thread-1957450-1-1.html[2011/8/21 21:56:10]

作在当前的软件开发过程中经常发生，这正是Subversion被开发并取代CVS 的主要原因之一。

Subversion

将目录作为一类特殊的文件来处理（事实上，从文件系统的角度来看，目录确实是一类特殊的文件，当目录中的子目录/文件被删除、重命名、或新的子目录/文件被创建时，目录的内容将发生改变）。因此，Subversion象记录普通文件的修改历史一样记录对目录的修改历史，当发生文件/目录的移动、重命名或拷贝操作时，Subversion能够准确记录操作前后的历史联系。同样，象对文件的不同历史版本进行比较一样，Subversion支持对目录的不同历史版本的比较，清晰展现目录的变化历史。

四、原子性提交

从使用者的角度来看，CVS 和Subversion 都支持对多个文件修改的批量提交，但二者在实现方式上存在本质的区别。

CVS 采用线性、串行的批量提交，即依次地，一个接一个地执行提交，每成功提交一个文件，该文件的一个新的版本即被记录到版本库中，提交时用户提供的日志信息被重复地存储到每一个被修改的文件的版本历史中。

CVS 串行批量提交模式的弊端在于

— 当任何原因造成批量操作的中断时（典型原因包括：网络中断、客户端死机等），版本库往往处于一个不一致的状态：原本应该全部入库的文件只有一部分入库，很有可能版本库中的最新版本不能顺利编译，更为严重的是，随着其他的用户执行cvs update操作，该不一致性将迅速在开发团队中扩散，从而严重影响团队的开发效率，并存在质量隐患。另外，假如该批量提交的中断没有被及时发现，开发团队往往要花更多的时间进行软件调试和排错。

CVS 即使在批量提交不发生中断时也会造成不一致：假设用户A 启动一个需要较长时间才能完成的批量提交；与此同时，用户B 执行cvs update 操作。此时，用户B 很有可能得到一个不一致的更新，即用户B 通过"更新"操作，得到用户A 的部分修改文件。

Subversion 彻底消除了CVS

的以上弊端。无论批量提交包含多少文件修改，只有当全部文件修改都成功入库，该提交才变得有效，才对其他用户可见；否则，无论任何原因造成中断，Subversion 都会自动执行"回滚"（rollback）操作。换一个说法，Subversion 保证所有的修改要么全部入库生效，要么一个也不入库，即对版本库不作任何的修改。这就是Subversion 的原子性提交（atomic commit）。

由于Subversion 的原子性提交特性和全局版本编号方式，当提交成功完成时，一个唯一的、新的全局版本编号产生，而提交时用户提供的日志信息与该新的版本编号关联，只进行一次存储（区别于CVS 的按文件重复存储）。

五、支持变更集概念

由于Subversion

的所有提交是原子性的，每次成功提交形成的唯一的全局版本号对应此次批量提交的所有文件修改，也就是说，一个Subversion版本号其实对应了一个逻辑上的变更集（change set），该变更集可能对应于对一个BUG 的修复，或者对应于对一个已有功能的改进，或者对应于一个新功能的实现。可以说，变更集是一个软件开发活动的逻辑结果，该变更集可以通过其对应的版本号在软件开发的其它过程中（如软件合并/集成过程，软件发布管理，变更管理系统，缺陷追踪系统）被引用。因此，Subversion 将版本管理从单纯的、单个的文件修改的层次通过逻辑上的抽象，上升到更便于理解和交流的开发活动的层次。

六、差异化的二进制文件处理

由于历史原因，CVS 主要是为早期的程序员设计的，CVS 能够有效处理文本文件（或ASCII文件，源代码文件），可以对文本文件进行差异化的存储、新旧版本的比较，文件合并等；但对于二进制文件，CVS则明显力不从心。在CVS 的版本库中，对于二进制文件的历史版本，CVS 唯一能做的就是对不同的版本进行独立的、冗余的存储，哪怕版本之间其实只存在微小的差异。举例而言，一个10M 的二进制文件（照片、图形文件、机械设计文件、电子设计文件）假如每周修改一次，无论每次修改的大小，一年下来，仅该文件就要消耗500M 以上的存储空间。而且，客户端每次获取该文件的新版本都要消耗10M 的网络流量。

对于目前的开发团队，无论是软件开发，Web 站点的开发，手机等电子产品的研发，需要进行版本管理的不仅是源代码等文本文件，还需要管理需求文档、设计文档、测试文档、用户手册，图形图像文件，机械/电子设计文件等诸多的二进制文件，CVS 显然不是一个好的选择。

与CVS 不同，Subversion 采用统一的二进制差异算法（binary differencing algorithm），即对文本文件和二进制文件采用相同的差异比较算法，并以相同的方式在版本库中进行存储：每次提交后版本库中只存储相对于先前版本的差异，从而可以节省大量的存储空间。

该二进制差异算法不仅应用在版本的存储上，更为重要的是，Subversion 对二进制文件与文本文件一视同仁，当客户端需要获取新的版本时（如执行svn update），在网络上只有版本的差异被传输，从而大大减少对网络带宽的消耗。更多细节参见"七、双向的差异化—压缩网络传输"。

七、双向的差异化—压缩网络传输

如上所述，CVS 对二进制文件不能进行有效的差异化处理。对于文本文件，CVS 仅仅支持单向的差异化传输：从CVS 到客户端的传输是差异化的，即执行cvs update 时，只有差异的部分从服务器传输到客户端；而当执行cvs commit 时，无论代码变化多少，CVS 都需要从客户端向服务器完整传输被修改文件的全部内容，不能只传输差异。

相反，无论是文本文件还是二进制文件，Subversion 都进行双向的差异化传输，并且差异化内容还要进行压缩/解压缩的过程：在服务器端获取差异显而易见，与CVS 类似；Subversion 在客户端获取差异的秘密在于 — Subversion 在客户端的工作拷贝中隐含了每个文件的一个"只读的、干净的"副本（该副本隐藏在隐含目录.svn 里，通常不可见，该副本还有更多的妙用，参见"十二、更多的本地/离线操作"），通过比较用户在客户端的修改和该隐含的副本，Subversion 获取需要真正传送到服务器的差异，并对差异进行压缩后才进行网络传输。

对CVS 而言，操作的成本（网络带宽消耗是最大的操作成本）与被修改的文件的大小成比例，而与修改本身的大小无关；对Subversion 而言，操作成本只与修改本身的大小成比例，而与被修改的文件的大小无关。因此，与CVS 相比，Subversion 消耗更少的网络带宽（以客户端的存储空间换取更少的带宽消耗在目前的计算环境下应该是个相当不错的选择！）。Subversion 更加适合基于互联网（或广域网）进行协作开发的地理上分布的团队 — 版本服务器集中、单一；客户端广泛分布。

八、高效、快捷创建分支和基线

CVS 和Subversion 都支持分支（branch）和基线（tag），通过分支与合并，可以有效支持大项目的并行开发模式；通过基线管理，可以准确标识一组文件的版本，有效进行软件发布管理和必要时的历史回溯。

但CVS 和Subversion 在实现分支和基线的方式上存在很大的不同。CVS 在创建分支的时候，需要对所有进行分支的文件进行依次的操作，因此分支的建立成本（主要是建立分支所需的时间，或消耗的计算资源）与参与分支的文件数量成比例，项目越大，版本库越大，文件越多，分支的建立成本越高；基线（tag）的建立与此类似。

Subversion

的分支和基线是通过执行"拷贝"来建立的：回想一下在没有引入版本管理工具的时候我们是如何进行所谓的"分支"和"基线"管理的？答案显然是"拷贝"

—
我们通过"拷贝"或"备份"来建立基线；同样，为支持多个开发人员可以同时进行开发，我们为每个开发人员创建一份"拷贝"。由此看来，Subversion 通过"拷贝"来建立分支和基线显得非常自然，有点"返朴归真"的意思。

由于Subversion 的全局版本号特性，Subversion 中分支或基线的创建过程，或Subversion中的"拷贝"过程，真正的操作是在版本库中创建一个到某一全局版本号的指针（pointer），不再需要针对众多的单个文件依次执行操作。因此，该操作的成本为一个很小的常数，与项目大小，版本库大小，文件数目的多少无关；并且，分支或基线的建立不需要进行版本的冗余存储，新建立的分支或基线基本不占用版本库空间，分支的后续存储空间的开销也只与修改的大小有关。

九、集成Apache Web Server，提供更多的特性

Subversion 通过与Apache Web Server 的集成，可以提供基于http/https 协议的版本库访问机制，从而支持Subversion 跨越防火墙的安全访问。除此以外，Subversion 还可以利用更多的Apache 特性，包括但不限于：Apache 丰富的用户认证机制（包括通过LDAP服务器如Windows Active Directory 服务器的用户认证），基于目录路径的精细粒度的访问控制，对传输的网络流量进行压缩/解压缩，浏览版本库目录结构等等。

十、支持WebDAV

WebDAV（Web-based Distributed Authoring and Versioning）是一种基于 HTTP 1.1 协议的通信协议.它扩展了HTTP 1.1，在GET、POST、HEAD 等几个HTTP 标准方法以外添加了一些新的方法，使应用程序可直接对Web Server 直接读写，并支持写文件锁定(Locking)及解锁(Unlock)，还可以支持文件的版本控制。Microsoft windows2000/XP 及IE, Office 还有Adobe/MicroMedia 的DW 等都支持WebDAV，这又大大增强了Web 应用的价值，以及效能。对于需要大量发布内容的用户而言，应用WebDAV 可以降低对CMS 系统的依赖，而且能够更自由的进行创作。上传、下载变得轻松自如。Subversion 通过与Apache Web Server 的集成，支持WebDAV 协议，使得业务用户（business users）或非技术用户在不安装任何版本管理客户端的情况下轻松访问Subversion 版本库，不改变业务用户已有使用习惯，支持分布的业务用户对文档的评审、修改并实现版本控制，真正将软件开发生命周期从开发/技术团队扩展到项目的全部干系人（stakeholder），避免通过电子邮件传递文档的混乱与无序、通过Windows 操作系统共享造成的安全漏洞、病毒攻击、历史版本被覆盖或丢失、审计困难等诸多典型问题。

十一、更好的冲突标识与处理

CVS 和Subversion

都支持通过分支与合并进行并行开发，并可以自动检测到合并时的冲突（conflicts），并在合并结果中以>>>>标识合并的冲突部分。
在CVS 中，经常会出现由于用户的疏忽（如，没有注意到冲突，或没有完全处理好冲突）而将仍然带有>>>>冲突标识符号的文件直接进行提交（commit），从而在版本库中产生垃圾版本。
Subversion 有效解决了CVS 的以上问题：Subversion 记录并保持文件的冲突状态，只有当用户明确执行svn resolved 命令后，该冲突状态标识才被复位，该文件才能被提交，从而大大减少了将仍然带有>>>>冲突

标识符号的文件直接进行提交的可能性。

十二、 更多的本地/离线操作

众所周知，CVS 客户端的工作拷贝中包含了一个隐含目录CVS，该目录中记录了客户端需要的一些管理信息；与此类似，Subversion 的客户端工作拷贝中也包含了一个隐含目录.svn，该目录中同样记录了客户端需要的一些管理信息，如版本库URL，当前访问版本号等。

与CVS 不同的是，Subversion 的.svn 目录中还包含了工作拷贝中每一个文件的一个"只读的、干净的"副本。正是由于该副本的存在，使得Subversion 与CVS 相比，可以执行更多的本地/离线操作，即某些操作不需要访问版本库服务器，因此不需要存在从客户端到服务器的网络链接，当然也不消耗任何网络带宽，这进一步增强了Subversion 对广域网的友好支持。

Subversion 的以下命令可以进行离线操作：

svn status — 显示工作拷贝上的本地修改概况；

svn diff — 显示工作拷贝上的本地修改细节，比较修改前后的内容；

svn revert — 撤销工作拷贝上的本地修改；

十三、 对符号链接进行版本管理

在Unix 文件系统中，符号链接（symbolic links，包括硬链接和软链接）是一种重要的文件系统元素。CVS 不能对符号链接进行版本管理；Subversion 则可以对符号链接进行版本管理。

十四、 元数据管理

与CVS 相比，Subversion 增加了元数据（metadata）管理机制。即可以对版本库中的文件或目录附加任意的"属性"（property），并记录属性的变化历史，也就是对元数据进行版本管理。一个Subversion 属性是一个"属性名称/属性值"的二元组，如"BugNumber=100"就是一个属性，可以将该属性附加到版本N 上，以说明版本N 改正了编号为100的BUG。

Subversion 元数据的目的是提供附件的信息以满足流程或过程自动化的需要，以增强Subversion 的管理能力和自动化程度。Subversion 自身就通过"属性"来存储一些特殊的信息。一个使用Subversion 元数据的例子：可以在一些批处理的脚本程序或Subversion的钩子程序（hooks）中创建、访问、修改"属性"元数据来满足流程自动化的要求。

本文来自ChinaUnix博客，如果查看原文请点击：http://blog.chinaunix.net/u/16651/showart_2056021.html

ChinaUnix 移动开发好书送不停! | 探寻高校HPC的生命力赢限量版U盘 | FreeBSD企业部署应用 | 有奖讨论：我的内核之路

发表主题 回复帖子

首页 » 论坛 » Linux 文档专区 »

版块跳转 最近访问的版块

Copyright © 2001-2010 ChinaUnix.net All Rights Reserved 联系我们: admin2@staff.chinaunix.net
感谢所有关心和支持过ChinaUnix的朋友们 转载本站内容请注明原作者名及出处
京ICP证: 060528号
北京市公安局海淀分局网监中心备案编号: 1101082001

清除 Cookies - ChinaUnix - Archiver - WAP - TOP