CVS を活用したWebコンテンツの分散共有管理

馬場 肇
京都大学大学院理学研究科

概 要

「日本の Linux 情報」では *CVS* の導入以後、多人数での共同作業を円滑に行なっている。これをバージョン管理システムが文書生産活動においても質的変革をもたらす好例ととらえ、*CVS* の特徴とネットワークを活用した作業スタイル確立までの経緯、実体験に基づく運用ノウハウを紹介する。

1 「日本の Linux 情報」

「日本の Linux 情報」の管理チーム (*Webmasters*) は、`http://www.linux.or.jp/` のコンテンツを管理しているボランティアグループである。`http://www.linux.or.jp/` は JLUG (Japan Linux Users Group) の発足とともに 1996 年 5 月に公開され、当初から国内の Linux 関連情報を集めたポータルサイトとして広く利用されてきた。しかし 1998 年以降の Linux 情報の爆発的な増加にも関わらず、現状に追随できていない部分が多くなっていた。

これを憂えた有志の Linux ユーザーによって *Webmasters* が 1998 年 5 月に再結成された。多人数での共同作業を円滑に行う手法を模索した結果、*CVS* システムを取り入れて、1998 年 10 月にほぼ現状のかたちで公開した。現在も多くの Linux ユーザーに利用されている。

2 *CVS* の特徴

共同編集作業の基盤には、バージョン管理システムの一つである *CVS* (Concurrent Version System) を用いている。*CVS* はプログラム開発において利用するのが普通であり、とくにこの 1-2 年で急速に広まりつつある。その背景要因には、オープンソースを介したインターネットを使った「共有型開発」手法が一般的になったことが挙げられる。しかしバージョン管理システムの適用範囲は、もはやプログラム開発現場にとどまらない。その一例は、Web サイトのコンテンツをバージョン管理することである。

規模の大きな Web サイトは、机を並べて作業する数人のグループだけの情報で運営されていることは少なく、一般に組織全体のユーザーから、ときには複数の組織から情報を収集する。このような場合、情報を寄稿してくれる人たちが、地理的に離れた場所にいるということも珍しくない。このような情報寄稿者を共同作業者として迎え入れるために、バージョン管理システムを使って Web コンテンツを管理すれば、地理的な壁は簡単に乗り越えることができるし、しかも互いの作業に干渉することなく、共有する Web コンテンツを編集できるようになる。

また、バージョン管理システムの管理下に置けば、リポジトリに対して commit を行うだけで済み、リモートサーバでのファイルシステムと URL へのマッピングといった細かいことはサーバ管理者にまかせておけば良くなるので、本来すべき情報の更新そのものに注力できるようになる。

3　*CVS* の仕組み

CVS は、作業ファイルをインターネットを経由してアクセスできるリモートサーバの中にあるリポジトリに更新履歴つきで一元管理している。ネットワークに繋がった計算機を持っていれば、このリポジトリに入っているファイルを編集したり、リポジトリに新たなファイルを追加したりすることができる。リポジトリの内容を一度手元のマシンにコピーしてきて、それに対して作業を行うようなかたちになる。したがって回線の負担もそれほど大きくなく、効率的に分散作業を行うことが可能になる。各作業者の変更履歴は記録に残るので、どこが改変されたかがわかるし、必要ならば元の状態に戻すこともできる。

4　実体験に基づく運用ノウハウ

4.1　ネットワークを活用した作業スタイルを確立

前述の通り、作業者は全国各地に在住している。作業者間の連絡に使っているのは、メーリングリスト (*ML*) と *IRC* (Internet Relay Chat) である。*ML* は議論や作業の場としては一般的であるし、時間や OS を問わないことも適している。*IRC* はチャットシステムの一つで、即時性に優れることが特徴で作業現場向きだが、会話の帯域が狭いことが欠点である。そこで、*ML* と *IRC* は相互に補完しながら利用している。

また、我々の用いているシステムでは、ある作業者がリポジトリに対して変更を行うと、その内容がグループ全員にメールで自動的に配信される。これによって、作業の進行度合を知ることができるし、変更点に対するチェックも簡単にできる。*ML* や *IRC* が「能動的」であるのに対して、自動通知 *ML* は「受動的」であるとも言える。

4.2　作業ルールを適度に保つ

一般的には、HTML べた書きよりも SGML や XML その他のメタ形式で記述する方が中長期的に見た場合より効率が良いと考えられる。しかし、ボランタリスタッフからなるプロジェクトにおいてもそれが必ず真であるかには一考の余地がある。基本的に、作業にかかるための学習コストはできるだけ引き下げないと、作業者を増やすことができないということにもなりかねない。HTML べた書きは、そういう意味で言えば作業しやすいということである。

同時に、htmllint や checkbot といった使いやすいツール類を併用して、作業コストの低減と一定品質の維持との両立に努めている。

4.3　作業者のモチベーションを維持

多くの人々の作業を束ねてプロジェクトを推進していくためには、メンバー一人一人のやる気、モチベーションを維持していくことが重要です。では何がモチベーションにつながるかというと、これはやはり作業が公開されること、評価されることと言える。したがって、各人の作業を速やかに公開することが非常に重要である。

また、プロジェクトの運転を止めないためには、すべての作業段階 (今回の直接の対象としては文書の登録・公開作業) において、その作業を行える立場の人を複数置くようにすることが必要である。この面からも作業ルールを適度に保つ必要があるのである。