TEX w pracy grupowej

Adam Dawidziuk
ul. Karczewska 59/17
05-400 Otwock
ADawidziuk@gust.org.pl

Streszczenie

Wykorzystując programy TEX i CVS można stworzyć system umożliwiający grupowe tworzenie dokumentów. Podstawowymi zaletami rozwiązania jest bardzo niski koszt oprogramowania i wysoka niezawodność. System umożliwia tworzenie dokumentu przez wiele osób jednocześnie i może być używany do pracy rozproszonej przy wykorzystaniu sieci o małej przepustowości (np. połączenia modemowe).

Opis problemu

System TEX powstawał przy założeniu, że tworzeniem i składaniem dokumentu w danym momencie zajmuje się jedna osoba. Istniejące bezpośrednio w TEX-u mechanizmy (takie jak `\input` czy `\include`) pozwalają podzielić dokument na kilka mniejszych części, co poniekąd umożliwia pracę nad dokumentem grupie autorów. Taki styl pracy, mimo iż możliwy do zastosowania, nie jest ani wygodny, ani łatwy.

Jeśli praca grupowa zaczyna mieć charakter ciągły, a liczba autorów (lub raczej tych wszystkich, którzy dokument modyfikują) jest większa od dwóch, to pojawiają się problemy poważne. Trzeba uzgodnić, gdzie znajduje się „oficjalna” wersja plików składających się na dokument. Potrzebny staje się *serwer*, którym może być komputer należący do osoby koordynującej prace lub zupełnie niezależna maszyna sieciowa. W jakiś sposób trzeba rozwiązać problem jednoczesnego modyfikowania przez kilka osób tego samego pliku. Można próbować ustalić kolejność działań lub stosować mechanizmy blokujące dostęp do pliku „otwartego do modyfikowania”. Tego typu blokady mogą znacznie zmniejszyć efektywność działania zespołu, jeżeli podział dokumentu na pliki składowe nie jest zgodny z podziałem pracy między poszczególne osoby.

Praca autorów nie musi być niezależna. Jeżeli dokument nie jest zbiorem luźno związanych tekstów (materiały konferencyjne, czasopismo), to podczas pracy każdy z autorów będzie chciał wiedzieć, jak wygląda całość dzieła. Do tego potrzebna jest informacja o postępach prac pozostałych osób, w szczególności konieczne jest wymienianie komunikatów o „włożeniu na serwer” nowej wersji jakiegoś pliku. Jest to niesłychanie istotne, gdy wielkość dokumentu sprawia problemy w przesyłaniu plików między serwerem a komputerem używanym przez autora. Może to mieć miejsce np. przy wykorzystywaniu połączeń modemowych do pracy grupowej. W każdej chwili, każdy z autorów powinien wiedzieć, które pliki z serwera powinien skopiować do siebie.

Kolejna sprawa to kwestia kontrolowania wprowadzanych zmian. Potrzebna jest informacja o tym, kto, kiedy i po co zmienił każdy z plików, często wraz z opisem na czym zmiana polegała. Wygodny jest system kontroli wersji, w którym przechowywane są wszystkie wersje pliku zapisywane na serwerze, a każdy z użytkowników systemu ma dostęp do każdej z wersji.

I wreszcie ostatnia, ale wcale nie mało ważna, sprawa – bezpieczeństwo. Jeśli system jest sieciowy, to prawo do wprowadzania zmian powinny mieć tylko osoby do tego upoważnione. Przy czym system zapewniania bezpieczeństwa nie powinien być zbyt kłopotliwy dla użytkowników.

Przykłady dokumentów wymagających tak złożonego systemu przechowywania, to np. dokumenty firmy pracującej w kilku odległych od siebie oddziałach, czy też książka, której poszczególne fragmenty piszą różni autorzy wykorzystując do transmisji danych łącza o niewielkiej przepustowości (np. modemy)

Wymagania

Precyzując wymagania można powiedzieć, że do pewnych zastosowań potrzebny jest taki system wymiany dokumentów, który:

- przechowuje wszystkie wersje wszystkich dokumentów,
- rejestruje nazwy użytkowników wprowadzających zmiany,
- potrafi porównać wersje plików znajdujące się na dysku lokalnym z ich odpowiednikami na

serwerze i uaktualnić tylko te pliki, które były zmienione,

- identyfikuje użytkowników zdalnych w sposób pewny i niekłopotliwy,
- w jakiś sposób rozwiązuje problem jednoczesnego zmodyfikowania tego samego pliku,
- działa automatycznie (nie wymaga obsługi operatora).

Tego typu lista wymagań wcale nie wygląda na banalną. Zawiera wiele ważnych cech „przemysłowych" systemów wspierających obieg dokumentów w dużych instytucjach, których (systemów) istotną cechą jest to, że kosztują bardzo dużo i zwykle nie wspierają formatu takiego jakim jest TEX, co jednocześnie dyskwalifikuje takie systemy w zastosowaniach gdzie ważna jest wysoka jakość typograficzna.

Możliwe rozwiązanie

Tak naprawdę, to problem nie istnieje. Szukając skutecznego rozwiązania można i należy wykorzystać fakt, że źródła dokumentów w systemie TEX są zapisywane jako zwykłe pliki tekstowe, podobnie jak źródła programów. Problem pracy grupowej w programowaniu istnieje od dawna – wystarczy więc na potrzeby tworzenie dokumentów zaadaptować metody i narzędzia stosowane przez programistów. Narzędzia takie są szczególnie dobrze rozwinięte w systemach operacyjnych „sieciowych z natury" a więc we wszelkiej maści unixach, w tym w systemie Linux.

Najwięcej z wymienionych wymagań, a mówiąc wprost, *wszystkie*, spełnia system CVS (*version control system*) (który jest domyślnie instalowany w większości rozprowadzanych dystrybucji Linuxa). CVS jest narzędziem dobrze działającym, dostatecznie niezawodnym oraz dostępnym w wersjach dla MS Windows. Dodatkowo ma tę miłą cechę, że radzi sobie w sytuacji, kiedy dwóch autorów jednocześnie zmodyfikuje ten sam plik. W szczególności nie ma problemu, jeśli zmiany dotyczą różnych fragmentów tego samego pliku. Oczywiście na uwagę zasługuje fakt, że CVS (w konfiguracji domyślnej) nie blokuje dostępu do pliku pobranego do modyfikacji, co oznacza, że nawet w przypadku niezbyt przemyślanego podziału dokumentu na kawałki autorzy nie będą blokować sobie możliwości pracy.

CVS korzysta z *repozytorium*, gdzie znajdują się wszystkie zebrane wersje plików. Repozytorium to katalog z drzewem podkatalogów, który może znajdować się na maszynie lokalnej, bądź na jakiejś maszynie w sieci. Zmiany plików są zapamiętywane przyrostowo, tzn. w repozytorium znajduje się wersja podstawowa pliku wraz z informacjami o tym, czym różnią się od wersji podstawowej wersje kolejne.

W komunikacji z repozytorium znajdującym się na serwerze zdalnym można wykorzystać program szyfrujący połączenia `ssh`. W tym celu konta użytkowników systemu CVS należy powielić na serwerze zawierającym repozytorium. Jeśli warunki bezpieczeństwa na to pozwalają (tzn. do komputerów osobistych autorów nie mają dostępu osoby trzecie) w katalogach domowych na serwerze można umieścić odpowiednie klucze ssh, co umożliwi modyfikowanie plików w repozytorium bez konieczności podawania hasła – przy czym każdy autor jest przez system poprawnie identyfikowany. Stosowanie ssh wymaga ustawienia zmiennej środowiska `$CVS_RSH=ssh`.

W repozytorium wystarczy przechowywać dokumenty źródłowe w formacie TEX. W każdej chwili, każdy z autorów może otrzymać informacje o tym jakie pliki zostały zmodyfikowane przez innych autorów, skopiować do siebie zmienione i dodane pliki oraz zapisać w repozytorium nowe wersje plików, które sam zmienił.

Jeżeli autorzy mają jednakowe instalacje TEX-a, to na podstawie plików źródłowych każdy z nich może lokalnie wygenerować poprawne wersje formatów wynikowych dokumentów (DVI, PS, PDF, HTML). Jeśli przetwarzanie jest skomplikowane warto opis zależności między plikami i sposób ich przetwarzania opisać w formacie zrozumiałym przez program `make`. Po uaktualnieniu wersji lokalnych dokumentów, autor (użytkownik) wykonuje tylko polecenie `make`, co nie tylko jest łatwe do zapamiętania, ale pociąga za sobą wykonanie tylko niezbędnych przetworzeń dokumentów źródłowych.

Użytkownicy edytor Emacs zwykle oczekują od życia więcej. I słusznie – emacs traktuje system CVS jak coś dobrze znajomego i pozwala na jego wykorzystywanie z poziomu menu lub przez polecenia z klawiatury (analogiczne jak dla systemu kontroli wersji RCS).

Przykład

Używanie systemu CVS jest banalnie proste. W trakcie zwykłej pracy wykorzystuje się polecenie `update` lub `get` do pobrania z serwera aktualnych wersji plików. Polecenie

```
> cvs update
```

wydane w katalogu roboczym spowoduje wykonanie takiego uaktualnienia. Polecenie

```
> cvs get katalog
```

spowoduje pobranie z repozytorium wszystkich plików znajdujących się w katalogu `katalog` (i podkatalogach). Polecenie `get` można wykonać w dowolnym miejscu – katalog `katalog` zostanie utworzony w katalogu bieżącym, o ile go jeszcze nie ma. Podobnie jak `get` działa polecenie `checkout`. Nazwę repozytorium można podać stosując opcję `-d` lub ustawiając zmienną środowiskową `$CVSROOT`. Dla polecenia `update` położenie repozytorium jest ustalane na podstawie zawartości katalogu `CVS`, który znajduje się w katalogu bieżącym (`update` dotyczy katalogu bieżącego). Przykład pobierania plików z repozytorium zdalnego (wiersz złamany wyłącznie na potrzeby druku):

```
> cvs -d :ext:alpha.praca.com.pl:/cvs
      get dokumenty/
```

Wprowadzanie dokumentów do repozytorium jest równie proste. Nowy plik wpisuje się do repozytorium poleceniem `add`:

```
> cvs add nowy.plik
```

po czym należy zapisać do repozytorium zawartość tego pliku (złamanie wiersza jw.):

```
> cvs commit -m
      "komentarz dotyczący zmiany"
      nowy.plik
```

Funkcje systemu CVS są bardziej rozległe i doskonale opisane w dokumentacji. Na przykład

```
> cvs log
```

oznacza żądanie pokazania informacji o wszystkich przechowywanych wersjach plików znajdujących się w danym katalogu.

Problem może sprawić założenie nowego repozytorium. Jeśli w katalogu bieżącym znajduje się kilka plików tekstowych, które powinny znaleźć się w repozytorium, można je zapisać wszystkie jednocześnie poleceniem `import` (wiersz złamany, bo się nie mieścił w szpalcie):

```
> cvs -d /cvs import
      -m "Pliki początkowo wstawione''
      katalog aqq start
```

Pliki te znajdą się w repozytorium w katalogu o nazwie `katalog`.

Repozytorium musi oczywiście wcześniej istnieć. W razie potrzeby można je stworzyć poleceniem `init`:

```
> cvs -d /cvs init
```

W pliku źródłowym dokumentu można zapisać znacznik identyfikatora pliku:

```
% $Id: $
```

który po uruchomieniu programu `cvs` zostanie uzupełniony informacjami o wersji danego pliku (wiersz złamany, bo szpalta wąska):

```
% $Id: TeXiCVS.tex,v 1.3 2000/04/10
      12:05:33 adam Exp $
```

Używając modemu do łączenia się z repozytorium zdalnym warto pamiętać o zestawieniu połączenia ppp przed przystąpieniem do pracy.

Podsumowanie i wnioski

Zastosowanie systemu CVS znakomicie ułatwia wspólną pracę wielu osób nad dużymi dokumentami zapisanymi w TEXu czy też zbiorami wielu dokumentów (oferty, faktury, etc.).

Funkcjonalność uzyskiwaną przez wykorzystanie systemów TEX i CVS trudno zestawiać ze znacznie uboższymi możliwościami popularnych programów do obsługi zbiorów dokumentów. Zbudowany w ten sposób system tworzenia i gromadzenia dokumentów może zaspokoić potrzeby nawet średniej wielkości instytucji, automatyzując znaczną część prac związanych z oznaczaniem i opisywaniem dokumentów. Jest to przy okazji system bezpieczny, który nie wymaga do działania łączy o dużej przepustowości, jako, że pliki źródłowe TEXa są względnie małe.

CVS okazuje się również przydatny jako narzędzie do szybkiego tworzenia kopii zapasowych (np. na fizycznie innym dysku) w pracy indywidualnej, przy okazji dając możliwość zarządzania wersjami.

Niezawodność systemu jest wysoka (szczególnie w zestawieniu ze standardami obowiązującymi w oprogramowaniu biurowym), jednak jest to system skomplikowany i nieodporny na losowe poczynania użytkownika. Z tego względu wskazane jest tworzenie dodatkowych kopii dokumentów, na przykład poprzez archiwizowanie repozytorium.

Zasady higieny pracy nakazują również, aby do repozytorium wkładać tylko te pliki, które są poprawnie przetwarzane przez TEXa.

CVS oryginalnie był przeznaczony do zarządzania wersjami oprogramowania, może jednak współpracować ze wszystkimi plikami tekstowymi, np. źródłami dokumentów w TEXu, metapoście, SGML-u, HTML-u itp.

◇ Adam Dawidziuk
ADawidziuk@gust.org.pl