

Nowe cechy CODA – system rozproszony z luźnym wiązaniem

- struktura CODA jest wzorowana na AFS
- fizyczna lokalizacja plików ukryta przed użytkownikiem
- wolumin jako jednostka przechowywania plików na serwerach
- replikacja i przenaszalność danych
- synchronizacja na poziomach lokalno-globalnym i globalnym
- jednolita przestrzeń nazw z korzeniem w katalogu /coda
- przechowywanie plików, katalogów i atrybutów w lokalnym systemie plików (pamięć podręczna luźno powiązana)
- pliki przechowywane permanentnie
- zarządzanie na poziomie woluminów
- otwarty kod źródłowy

WADY DZISIEJSZEJ IMPLEMENTACJI

- brak zaawansowanych mechanizmów autoryzacji dostępu
- niezaimplementowana synchronizacja przy użyciu rozgłaszania
- możliwość wykorzystania jedynie w systemach Unixowych (choć jest klient pod Cygwina dla Windows 95)

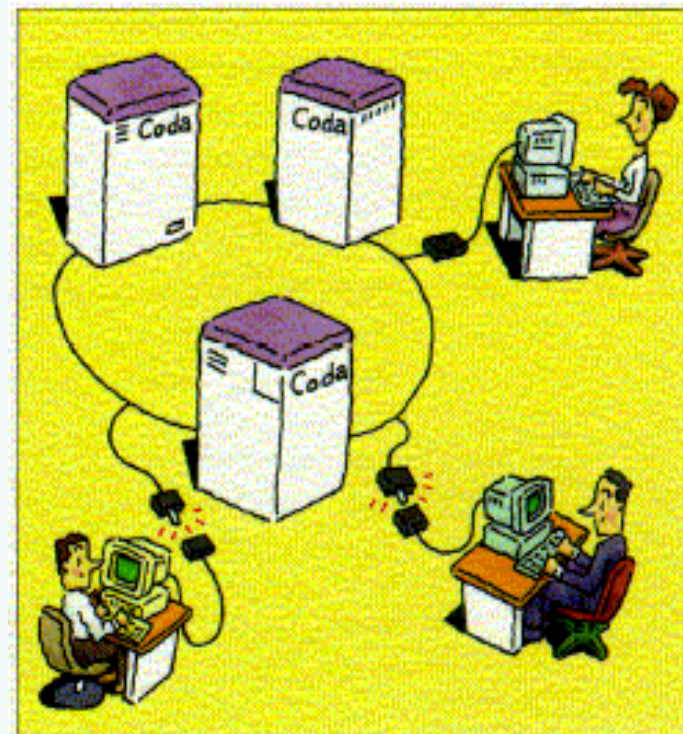
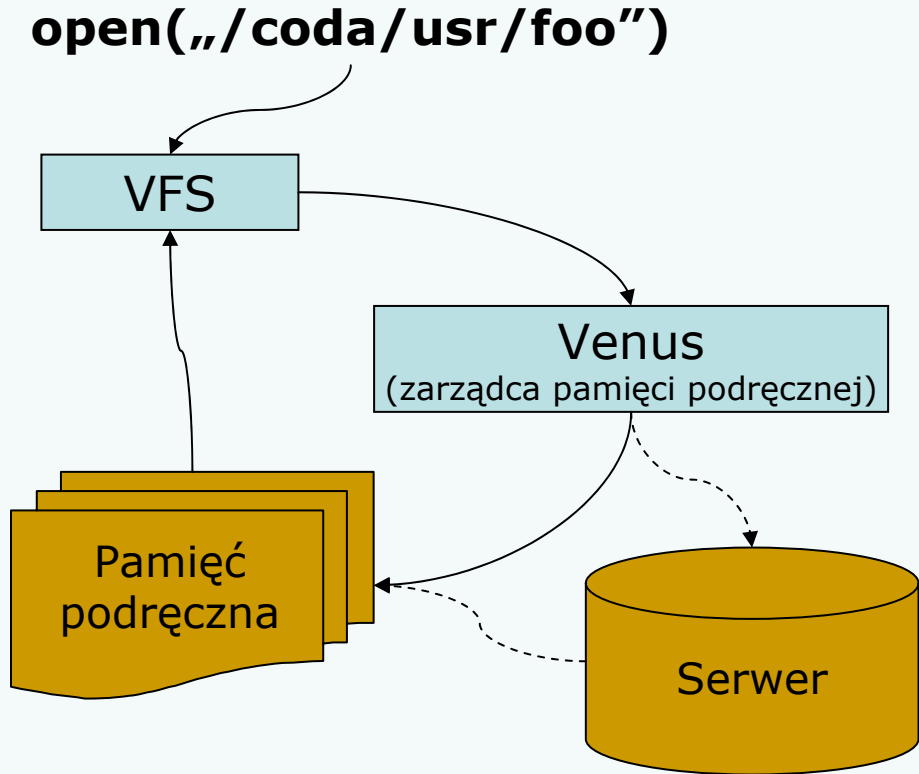


Figure 1: Coda Logo (Illustration by Gaich Muramatsu)

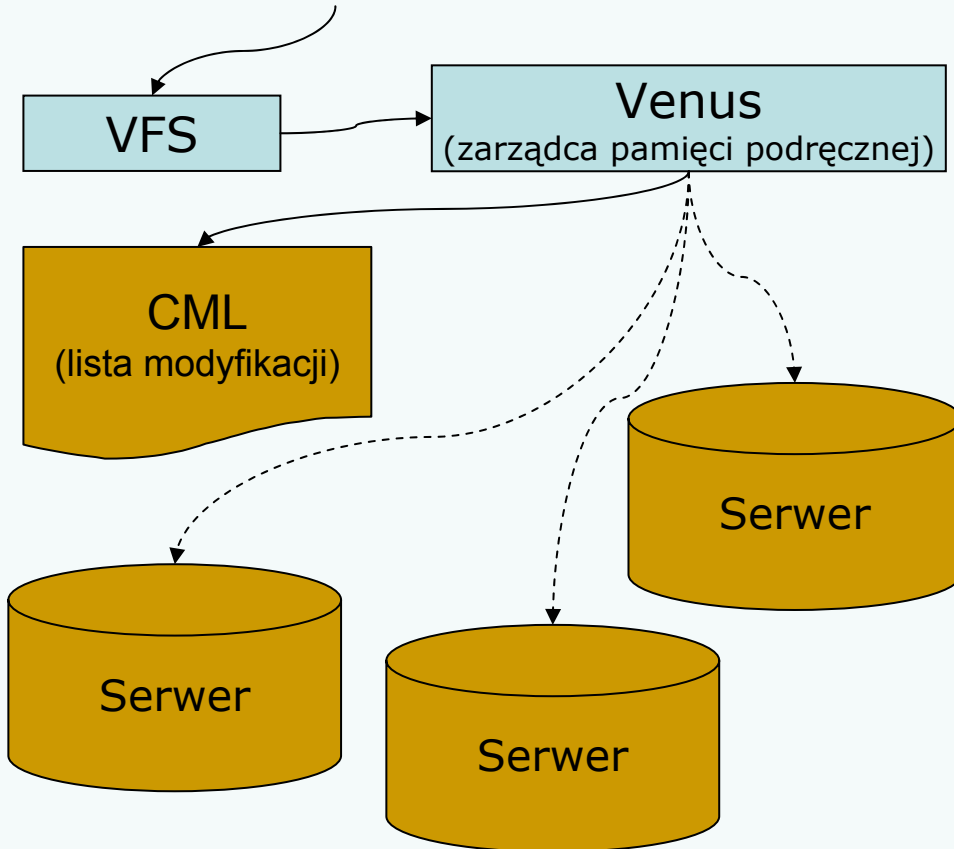
Otwieranie pliku w CODA – wykorzystanie pamięci podręcznej



- VFS otrzymuje od procesu użytkownika żądanie otwarcia pliku
- zarządca pamięci podręcznej (Venus) sprawdza czy nie ma aktualnej kopii roboczej żądanego pliku
- jeżeli nie, kopia jest sprowadzana z serwera
- proces użytkownika za pośrednictwem VFS otrzymuje dowiązanie do lokalnie przechowywanej kopii roboczej

Operacje read i write na plikach udostępnianych przez CODA wykonywane są na lokalnych kopiach roboczych, a więc z prędkością lokalnego systemu plików.

close(„/coda/usr/foo“)

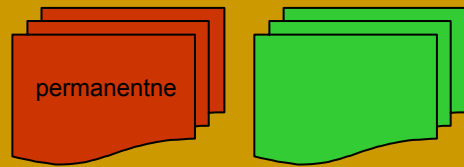


- VFS otrzymuje od procesu użytkownika żądanie zamknięcia pliku
- zarządca pamięci podręcznej (Venus) uaktualnia obrazu pliku przechowywane na serwerach
- jeżeli aktualizacja nie jest możliwa (zerwane połączenie), zarządca pamięci zapisuje zmiany wykonane przez użytkownika na lokalnie przechowywanej liście modyfikacji (CML)

Listy modyfikacji (CML) są używane do aktualizowania zawartości serwerów kiedy połączenie zostaje ponownie nawiązane (synchronizacja lokalno-globalna). Zawartość list jest optymalizowana pod kątem operacji np. utworzenia i usunięcia pliku.

Woluminy i luźno powiązane zarządzanie plikami

Lokalna pamięć podręczna



Pamięć podręczna umożliwia permanentne przechowywanie określonych plików lokalnie. Umożliwia to np. korzystanie z Xów lub innych rozbudowanych aplikacji podczas rozłączenia z siecią.

Na serwerach pliki są grupowane w **woluminy**. Umożliwia to np. grupowanie plików z katalogów domowych użytkowników w strukturach ułatwiających zarządzanie, relokację czy backup.

Volumeld
Vnodeld
Uniquifier

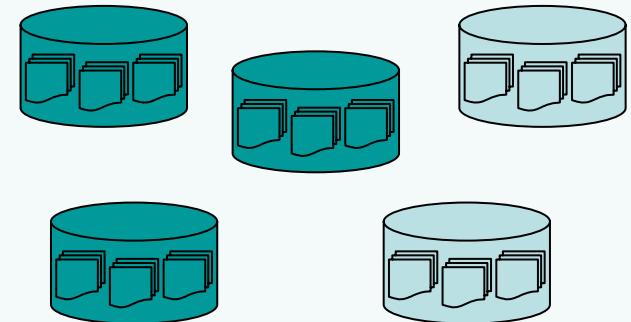
Plik jest identyfikowany przez jednoznaczny identyfikator **Fid**:

- **Volumeld** – określa numer woluminu przechowującego plik
- **Vnodeld** – numer i-węzła pliku na woluminie
- **Uniquifier** – numer używany do synchronizacji kopii

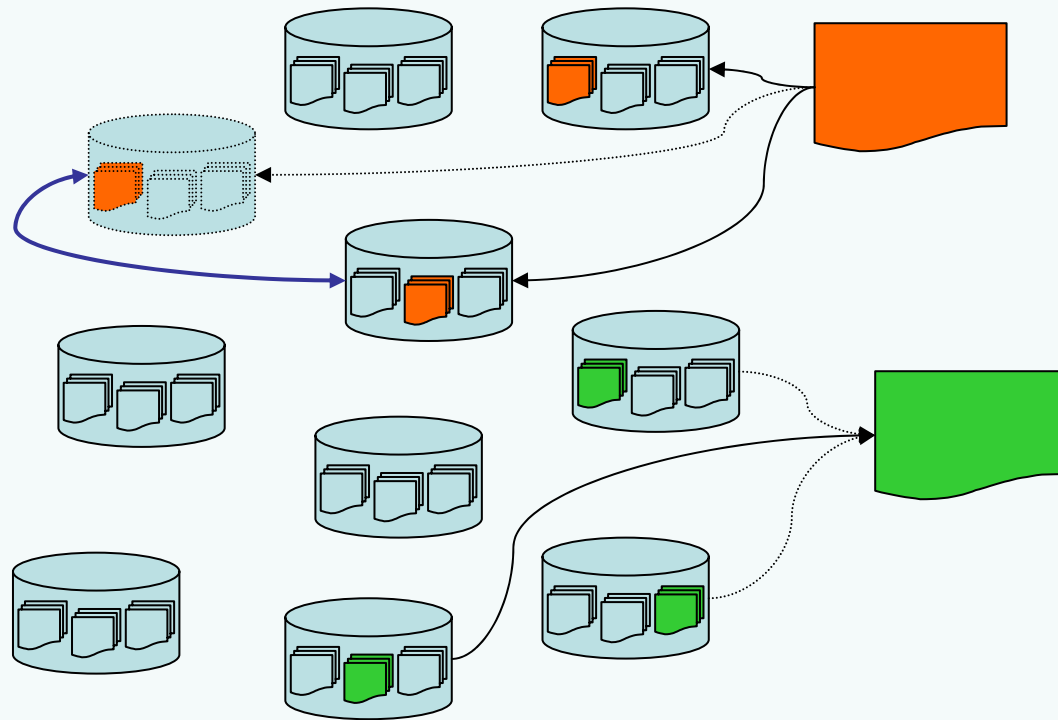
Serwery przechowujące wszystkie kopie woluminu tworzą **VSG** (Volume Storage Group).

Volumeld identyfikuje grupę serwerów jako VSG jak również wolumin na każdym z członków VSG.

Na każdym z serwerów przechowywany jest komplet metadanych związanych z woluminem, m.in. położenie wszystkich kopii woluminu.



Woluminy i luźno powiązane zarządzanie plikami



Podczas pobierania pliku dane pobierane są z jednego z serwerów należących do VSG.

Podczas zapisu aktualizacja propagowana jest do wszystkich aktywnych serwerów z VSG.

Podczas pobierania pliku Venus sprawdza aktualność kopii na wszystkich serwerach z VSG i w przypadku desynchronizacji uruchamia proces aktualizacji odpowiedniego serwera (synchronizacja globalna).

Mechanizm synchronizacji zaimplementowany w CODA wykorzystuje zewnętrzne mechanizmy wspomagające aktualizację plików specyficznych dla aplikacji. Niektóre aktualizacje wykonywane przez różnych klientów mogą być uznane za zgodne, inne powodują konflikty, które muszą być rozwiązywane ręcznie.

Dużą zaletą takiego systemu aktualizacji jest odporność na awarie. W przypadku utraty pojedynczego nośnika wystarczy podmienienie go na nowy i uruchomienie procedury synchronizacji serwerów.