Efektywność rozproszonych baz danych na podstawie implementacji X.500 Directory

Stan obecny

Maria Górecka
Tomasz M. Wolniewicz

1995–07–18

1 Wstęp

Podstawowym zagadnieniem w zakresie konstrukcji baz danych jest problem efektywności wytworzonego produktu. Ostateczny efekt działania projektantów jest możliwy do oceny dopiero przy badaniu konkretnej aplikacji, jako że zależy od ogromnej ilości czynnikow, często wzajemnie się wykluczających, a optymalne działanie osiąga się jako kompromis pomiędzy różnymi możliwymi wariantami.

Celem niniejszego raportu jest przygotowanie gruntu do badania efektywności systemu X.500 Directory jako przykładowej rozproszonej bazy danych. Przedstawiamy w nim problematykę i zwracamy uwage na najistotniejsze, naszym zdaniem, elementy systemu mające wpływ na jego sprawność. Ten przygotowawczy charakter raportu powoduje, ze zdecydowaliśmy się wprowadzić dosyć obszerne fragmenty opisujące specyfikę rozproszonej bazy danych X.500, wydają się one być niezbędne w celu właściwego zrozumienia opisywanego zagadnienia.

Jako kontunuację niniejszej pracy planujemy przedstawienie wyników rozważań i testów przeprowadzonych na konkretnej implementacji, w postaci zaleceń dla administratorów systemu X.500 w Polsce.

2 Informacje ogólne o X.500 Directory

2.1 X.500 jako termin

Terminem X.500 określa się potocznie funkcjonującą jako usługa sieci Internet, rozproszoną bazę danych, zawierającą informacje o instytucjach i ich pracownikach. Faktycznie X.500 jest standardem OSI, a funkcjonująca usługa jedynie jego implementacją. Terminu X.500 będziemy używać zarówno w potocznym jak i w formalnym znaczeniu, z kontekstu będzie wynikać, o jakie znaczenie chodzi.

2.2 Standard X.500

Seria zaleceń X.500–X.521, zebrana w [X500], jest standardem OSI[1] definiuje obiektową, rozproszoną bazę danych opartą na modelu sieciowym (Directory). Zalecenia te zostały opublikowane przez CCITT[2] i obejmują opis koncepcji i modelu bazy X.500, zasady ochrony i bezpieczeństwa danych, definicję usług, sposób realizacji zdalnych operacji i specyfikację protokołu.

Równolegle do serii X.500 opracowano standard ISO-9594 będący *oficjalnym* miedzynarodowym dokumentem definiującym Directory. [X500] zawiera, poza definicją samego standardu również szereg dodatków określających podstawowe typy atrybutów i klas obiektów, jak również wskazania co do możliwych sposobów implementacji Directory.

Użytkowy opis standardu X.500 i przykłady zastosowania można znaleźć w [ROSE].

Definicja standardu bazuje na notacji ASN.1 (Abstract Syntax Notation One), utworzonej dla potrzeb reprezentacji złożonych struktur danych, występujących w warstwie aplikacji modelu sieciowego OSI, a docelowo rozwiniętej dla szerokich zastosowań związanych z opisem protokołów OSI. Notacja ta umożliwia definicję skomplikowanych struktur danych, abstrahującą od środowiska implementacji oraz opis protokołów i operacji. Dodatkowo, dla tak opisanych struktur, określono sposób transmisji sieciowej, poprzez kodowanie według zasad przyjętych w standardzie BER (Basic Encoding Rules). Formalny język ASN.1 został zdefiniowany w roku 1987, jego wyczerpujący opis zawarty jest w [ASN1].

Pomimo, że podstawowym założeniem przyświecającym twórcom standardu było opracowanie światowej „książki adresowej”, X.500 dopuszcza przechowywanie w bazie danych najzupełniej dowolnych typów, np. zbiorów dokumentów, publikacji, zdjęć, czy informacji o zasobach programowych i sprzętowych.

[1] Open Systems Interconnection
[2] The International Telegraph and Telephone Consultative Committee

Standard X.500 został opublikowany w 1988 roku, a pod koniec roku 1994 oczekiwana jest publikacja nowej jego edycji. Zmieniony standard (nazywany X.500 — wersja 1993) znacznie rozszerza definicję z roku 1988, między innymi o elementy mające bardzo istotny wpływ na efektywność działania bazy. Standard 1993 został formalnie opisany za pomocą nowej edycji języka ASN.1, opublikowanej w 1993 roku.

W niniejszym raporcie bazujemy na wersji X.500 z roku 1988 oraz jej rozszerzeniach wprowadzonych przez eksploatowaną w Polsce implementację tego standardu zwaną QUIPU, będącą częścią pakietu ISODE[3]. Należy oczekiwać, że jeszcze przez dłuższy czas dominować będą implementacje tej właśnie wersji standardu. Przewidujemy, że po ukazaniu się nowej wersji niezbędna będzie konfrontacja naszych przemyśleń i wyników badań z nowymi ustaleniami.

X.500 jest bazą „obiektową". Obiekty (entries) przechowywane w bazie X.500 definiowane są w oparciu o pojęcie klasy. Definiuje się klasy podstawowe, a następnie ich podklasy. Podklasy dziedziczą konieczne i opcjonalne atrybuty swojej nadklasy, wprowadzając przy tym własne rozszerzenia, specyficzne dla opisywanej grupy obiektów. Bardzo często stosowane jest tworzenie nowej podklasy w postaci przekroju kilku klas wyjściowych. W ten sposób klasa obiektów identyfikuje rodzinę obiektów o wspólnej charakterystyce, a każdy obiekt bazy X.500 musi należeć co najmniej do jednej klasy. Opis konkretnego obiektu dokonywany jest poprzez określenie odpowiednich atrybutów. Atrybut składa się z typu, identyfikującego obiekt (OID — object identifier) i zbioru wartości. Atrybuty obiektów muszą zachowywać określone reguły syntaktyczne, które są przetwarzane zgodnie ze zdefiniowanymi dla nich metodami. Obiekty rozróżniane są poprzez swoje jednoznaczne nazwy (Distinguished Names), zaleca się takie dobieranie nazw, by dotarcie do szukanej informacji było proste i samonarzucające się. Większość typów atrybutów dopuszcza nadawanie kilku wartości, sprzyja to opisywaniu obiektu na wiele sposobów, satysfakcjonując różnych odbiorców.

Ta obiektowość bazy X.500 w połączeniu z sieciowym modelem przypomina bardzo model funkcyjny Bunemana-Frankela opisany w [M-R]. Struktura sieciowa X.500 jest w zasadzie ściśle hierarchiczna (drzewiasta), dopuszcza jednak odsyłacze (aliasy) komplikujące strukturę i mające istotny wpływ na efektywność systemu.

Podstawowym założeniem X.500 jest rozproszenie systemu i ogromny docelowy rozmiar bazy. W pełni użyteczny system oparty o X.500 powinien posiadać tysiące węzłów w każdym kraju i w skali świata miliardy obiektów.

[3]Patrz cześć 2.4

2.3 Specyfika bazy X.500

Jak wspomnieliśmy powyżej, X.500 zdefinowano przede wszystkim dla celów szeroko rozumianej informacji adresowej. Została ona zaprojektowana z myślą o aplikacjach i procesach zarządzania OSI oraz dla potrzeb usług telekomunikacyjnych. Nie było zamiarem twórców standardu X.500 utworzenie modelu bazy uniwersalnej, ogólnego użytku, w pełni niezawodnej i spójnej. Fakt ten ma istotny wpływ zarówno na sam standard, jego implementacje jak i funkcjonujące systemy. Podstawowe implikacje takiego podejścia to:

- Konieczność przechowywania danych w sposób użyteczny, zarówno przy dostępie wewnątrz jednego kraju jak i z zewnątrz; istotną cechą bazy X.500 jest „przyjazne" nazewnictwo obiektów, mające na celu uproszczenie wyszukiwania konkretnych danych.
- Możliwość stosowania lokalnych rozszerzeń klas obiektów, zarówno na poziomie kraju jak i instytucji w celu dodefiniowania cech obiektów.
- Tworzenie drzewa informacji zgodnego z naturalną strukturą administracyjną świata.
- Odporność na współpracę różnorodnych implementacji standardu oraz na lokalne ustalenia co do sposobu administrowania bazą.
- Założenie znacznej przewagi operacji odczytu nad operacjami modyfikacji bazy i dopuszczenie do czasowych niekonsystencji pomiędzy kolejnymi replikacjami fragmentów rozproszonej bazy danych.

2.4 Implementacja ISODE

ISODE (ISO Development Environment) jest zespołem bibliotek i programów jezyka C implementujących hierarchiczny model protokołów komunikacyjnych OSI. Pakiet ten dostarcza niezbędnych narzędzi do budowania aplikacji OSI, bazujących zarówno na typowych sieciach OSI, jak i na sieciach Internetowych — TCP/IP. Początkowo oprogramowanie to rozwijane było w celach badawczych, zmierzających do testowego tworzenia aplikacji OSI, docelowo obejmuje implementacje podstawowych usług sieciowych, eksploatowanych w wielu międzynarodowych programach pilotowych. QUIPU jest implementacją standardu X.500 (1988) w ramach ISODE. Poza tym ISODE zawiera implementację usługi poczty elektronicznej (MHS — Message Handling System), wirtualnego terminala (VT — Virtual Terminal), transferu plików (FTAM — File Transfer Access and Management), a także usługi zarządzania sieciami (SNMP — Simple Network Management Protocol — to zastosowanie nie bazuje na protokole OSI). Oprogramowanie QUIPU stało się podstawą dla uruchomienia w środowiskach akademickich pilotowych projektów usługi X.500 (PARADISE w

Europie i White Pages w Ameryce Północnej). Ten funkcjonujący system X.500 Directory, obejmujący również Polskę, jest areną doświadczalną dla standardu i był podstawą dla rozwoju jego nowej wersji. Nasze doświadczenia z X.500 pochodzą z udziału w projekcie PARADISE, w którym uczestniczymy od połowy 1992 roku.

2.5 Porównanie z DNS

X.500 jako usługa sieciowa jest z reguły porównywana z DNS (Domain Name Service) — systemem zarządzającym bazą informacji o adresach sieciowych, opisanym w [DNS]. Tendencja do takiego porównania jest naturalna, jako że DNS jest podstawą systemu dostarczania poczty elektronicznej w Internecie, a X.500 ma stać się jego odpowiednikiem dla systemu poczty X.400. DNS, tak jak X.500, jest rozproszoną bazą danych o strukturze hierarchicznej, podobnie jak w przypadku X.500 dopuszcza się czasowe niekonsystencje danych. W tym miejscu kończą się jednak podobieństwa. Podstawowymi różnicami w zakresie założeń funkcjonalnych są:

1. Sposób przechowywania danych.
2. Obiektowość i rozszerzalność X.500 w porównaniu ze ściśle narzuconą strukturą rekordów DNS.
3. Oparcie X.500 o standard w przeciwieństwie do tworzenia DNS *de facto* na bazie jednej aplikacji (BIND — Berkeley Internet Name Domain system).
4. Ograniczenie funkcjonalności DNS do operacji odczytu — **read** i **list**, bez możliwości przeszukiwania — **search**.
5. Brak wbudowanych w system mechanizmów aktualizacji danych.

W zakresie funkcjonujących służb różnica polega na tym, że DNS stanowi pełną bazę danych, umożliwiającą przesyłanie poczty elektronicznej, podczas gdy istniejąca baza X.500 jast dalece niekompletna. Należy przy tym jednak zauważyć, ze DNS pozwala na realizację łączności komputerowej pod warunkiem, że nadawca wiadomości zna dokładne dane adresata, podczas gdy X.500 pozwala odszukać konkretne dane na podstawie przybliżonych. Podobieństwa DNS i X.500 w zakresie technologii oraz zastosowania skłaniają do połączenia obu usług. Ponieważ dane DNS można bez trudu uczynić podzbiorem informacji zawartych w X.500, podejmowane są prace zmierzające do powiązania obu systemów ([HK91a]). Punktem wyjściowym do tego typu działań jest adaptacja bazy X.500 do przyjmowania danych pochodzących z DNS. W tym celu zdefiniowano rozszerzenia standardu X.500, umożliwiające wprowadzanie danych specyficznych dla DNS: m.in. przypisanie odpowiednim obiektom atrybutów określających nazwę domeny adresowej i zawartość typowego rekordu DNS.

3 Elementy mające wpływ na efektywność rozproszonej bazy danych

Efektywność działania bazy danych zależy od wielu składników, w przypadku baz rozproszonych ich ilość wyraźnie wzrasta ze względu na różnorodne i na ogół bardzo rozległe środowisko eksploatacji.

Podstawą zrozumienia działania bazy rozproszonej i czynników mogących wpływać na jej działanie jest pojęcie *modelu funkcjonalnego.* W jego zakresie określa się sposób współdziałania elementów systemu.

Rozproszona baza danych to fizycznie zasób informacji, składającej się ze zbiorów rezydujących na wielu komputerach, bedący, z logicznego punktu widzenia, spójną całością. W pełni funkcjonalna baza rozproszona zapewnia niezawodny dostęp do informacji z dowolnego miejsca. W przypadku, gdy istnieje konieczność dzielenia zasobów danych, najkorzystniejszym rozwiązaniem jest model rozproszony, wieloużytkowy. Charakteryzuje się on tym, że dane i aplikacje rezydują na wielu serwerach. Kluczowym zagadnieniem w takim modelu funkcjonalnym jest zapewnienie szybkiego i niezawodnego dostępu do danych, co wiąże się z następującymi wymogami:

- koniecznościa utrzymywania w ramach serwerów aktualnej informacji o rozmieszczeniu poszczególnych komponentów danych — służą do tego specjalne systemowe katalogi — sposób ich realizacji i aktualność danych decydują w dużej mierze o efektywności systemu;

- właściwą synchronizacją przepływu danych pomiędzy serwerami, czyli prawidłową obsługą wszelkich operacji rozproszonych, dzięki czemu eliminowane są sytuacje, w których realizacja poszczególnych operacji kończy się niepowodzeniem, czy prowadzi do zapętlenia systemu.

Opis modelu funkcjonalnego bazy X.500 zamieszczamy w części 4.

Poniżej przedstawiamy podstawowe czynniki, które uznaliśmy za najistotniejsze z punktu widzenia badań jakości operacyjnej rozproszonych baz danych.

1. **Stopień rozproszenia.**

   (a) Opóźnienia sieciowe. W rozległym systemie sieciowym rozproszenie bazy wprowadza oczywiste opóźnienia wynikajace z niedoskonałości połączeń.

(b) Czas reakcji serwera. Innym, zauważalnym efektem jest również czas pierwszej reakcji pojedynczego serwera — im wieksze rozproszenie, tym wieksze prawdopodobieństwo odwoływania się do serwerów pozostających w stanie uśpienia. Oczywiste jest, że rozproszenie bazy zmniejsza objętość danych w pojedynczych serwerach, a więc mamy tu do czynienia z dwoma przeciwnie pracującyni czynnikami.

(c) Wpływ obciążenia na czas reakcji serwera. W pewnym zakresie wzrost obciażenia może mieć dodatni wpływ na średni czas reakcji serwera (zmniejszenie ilości okresów uśpienia), nadmierne obciażenie wpływa natomiast niekorzystnie na efektywność.

2. **Zgodność systemu serwerów z logiką układu bazy.** Naszą analizę ograniczamy do przykładu bazy hierarchicznej, jako że z taką strukturą mamy do czynienia w systemie X.500. Baza hierarchiczna odwzorowuje przedstawianą rzeczywistość w postaci drzewa. To jakie fragmenty drzewa będą przechowywane w ramach pojedynczych serwerów ma istotny wpływ na czas reakcji systemu. Przy projektowaniu systemu serwerów trzeba brać pod uwagę najbardziej prawdopodobne zapytania i tak dobierać rozproszenie danych, aby minimalizować ilość niezbędnych połączeń sieciowych.

3. **Stosowanie odsyłaczy (aliasów).** Odsyłacze pozwalają na wielokrotne występowanie tego samego obiektu w różnych miejscach hierarchii lub pod różnymi nazwami. System taki zapewnia konsystencję bazy i jest absolutnie niezbędny, wymaga jednak niezawodnego algorytmu realizującego odwzorowanie aliasu na właściwą nazwę obiektu.

4. **Replikacja danych.** Nadzwyczaj ważnym elementem, z punktu widzenia sprawności, jest stosowanie replikacji danych, tj. kopiowanie fragmentów zasobów informacji i przechowywanie ich na wielu serwerach. Może to ograniczać ilość niezbędnych połączeń sieciowych, jak i podwyższać niezawodność działania bazy (w przypadku niesprawności pojedynczych serwerów).

5. **Sprawność poszczególnych serwerów.** Czynnik ten jest oczywisty i musi być brany pod uwagę w wielu sytuacjach — niecelowe jest na przykład replikowanie znacznych zbiorów danych z serwerów szybkich na wolne, gdyż doprowadzi to do generalnego spowolnienia działania serwera oraz do prowadzenia przeszukań na wolnym serwerze replikującym, zamiast na szybkim podstawowym. Replikacja taka może być jednak uzasadniona, gdy połączenie z serwerem podstawowym jest czasochłonne lub zawodne.

6. **Algorytmy generowania połączeń.** W przypadku, gdy potrzebna informacja jest niedostępna na bieżącym serwerze, konieczne jest wykonanie połączenia sieciowego z serwerem, który informację taką posiada. System zarządzający bazą danych musi wówczas zrealizować połączenie możliwie optymalne, wybierając serwer o najwyższej jakości operacyjnej. Generowanie niewłaściwych połączeń może doprowadzić bazę do niemal całkowitego zastoju.

7. **Stosowanie standardowych zapytań.** Kontakt z typową bazą danych realizowany jest zawsze za pomocą interfejsu użytkownika. Projektant takiego interfejsu musi znać specyfikę bazy i przewidywać, w jaki sposób będzie wykorzystywana. Ponieważ tego typu zbiory informacji są w zdecydowanej większości nastawione na operacje przeszukiwania, należy zdawać sobie sprawę z tego, jakiego typu zapytania są optymalne i tylko takie oferować użytkownikowi.

8. **Dokładność dopasowywania do wzorców.** Jedną z ważnych cech bazy danych jest możliwość wyszukiwania w sytuacji, kiedy dane wejściowe są niedokładne. W takim przypadku baza powinna wygenerować listę przybliżonych trafień. Dobór algorytmu dopasowywania przybliżonego jest nadzwyczaj istotny, jako ze może wydatnie wpłynąć na czas dostępu do danych, a jednocześnie decyduje o wielkości odbieranej listy przybliżeń. Zbyt luźno, lub błędnie funkcjonujące przybliżenia wprowadzają do wyników znaczny szum informacyjny, utrudniając i przedłużając czas pracy z bazą.

9. **Indeksowanie.** Indeksowanie bazy wpływa pozytywnie na czas przeszukiwań, ale równocześnie znacznie powiększa obszar pamięci potrzebny do przechowania danych istotne jest zatem aby stosować ten mechanizm w sposób optymalny.

4 Czynniki efektywnościowe w bazie X.500 i możliwość sterowania jakością operacyjną

Po określeniu elementów, które, naszym zdaniem, mają najistotniejszy wpływ na ostateczną jakość usług dostępu do bazy rozproszonej, poniżej, odwołując się do wymienionych czynników, analizujemy ich specyfikę w bazie X.500 Directory. Jednocześnie przedstawiamy elementy sterowania bazą QUIPU udostępnione obecnie każdemu administratorowi. Pomimo, że dokumentacja oprogramowania ([QUIPU]) opisuje rolę poszczególnych czynników, nie przedstawia żadnych ilościowych zestawień dotyczących wpływu rodzaju ustawień na efektywność operacyjną systemu. Szczegółowe przetestowanie, a następnie stworzenie raportu przedstawiającego optymalne ustawienia będzie ostatecznym celem prowadzonych przez nas prac.

Baza X.500 składa się z szeregu serwerów współpracujących ze sobą za pomocą ustalonych protokołów. Każdy serwer jest jednak w znacznym stopniu autonomiczny co do sposobu obsługi danych, generowania zapytań, replikacji danych itp. Jak już wspomniano, standard 1988 nie precyzuje zupełnie działania bazy w tym zakresie, implementacja QUIPU posiada szereg pozastandardowych metod sterowania. Nowy standard 1993 został rozbudowany o wiele elementów wpływających na współdziałanie serwerów, ale jak na razie nie jest on opublikowany w wersji końcowej i z pewnością, ze względu na sporą ilość modyfikacji, nie zostanie szybko w pełni zaimplementowany.

Na wstępie naszkicujemy podstawy modelu funkcjonalnego X.500 Directory.

Informacja przechowywana w bazie uporządkowana jest w strukturę drzewa (DIT — Directory Information Tree) i jest podzielona na fragmenty, tzw. „poddrzewa", zarządzane przez serwery zwane DSA (Directory System Agent), współpracujące ze sobą w celu wymiany informacji. Dostęp do bazy danych następuje poprzez wybrany interfejs użytkowy, zwany DUA (Directory User Agent), który w imieniu użytkownika komunikuje się z DSA. Interakcja między DUA i DSA odbywa się według prostej zasady pytanie–odpowiedź. Ponieważ dane wewnątrz DIT są rozproszone, na ogół pojedynczy serwer nie posiada informacji niezbędnej do odpowiedzi na zadane przez DUA zapytanie, musi natomiast potrafić zidentyfikować inne DSA, zdolne, jego zdaniem, do wykonania zlecenia. Model funkcjonalny definiuje, w ramach interakcji pomiędzy serwerami DSA, operacje typu odesłanie (referral) oraz łańcuchowanie (chaining).

Operacja odesłania polega na wskazaniu adresu innego DSA, do którego zaleca się przekazanie zapytania, w gestii DUA pozostaje kontynuacja realizacji zlecenia.

Operacja łańcuchowania, to przekazanie otrzymanego zlecenia innemu serwerowi DSA, który, w przypadku niemożliwości odpowiedzi, będzie kontynuować pracę poprzez wykonanie operacji odesłania lub łańcuchowania. W takiej sytuacji DUA nie jest świadome tego, jaką drogę przebywa zlecenie. W ramach łańcuchowania wyodrębniona jest operacja rozgłaszania (multicasting), w tym przypadku DSA łączy się jednocześnie z wieloma innymi serwerami i zadaje im to samo zapytanie, oczekując na pozytywną odpowiedź.

Opisany model funkcjonalny implikuje następujące wymagania, dotyczące serwerów DSA:

- możliwość sprawdzenia, czy realizacja zlecenia może nastąpić w oparciu o lokalny zasób danych,

- jeżeli nie można zlecenia wykonać lokalnie, umiejętność identyfikacji DSA, któremu należy je przekazać i wybór optymalnego sposobu współpracy z innymi serwerami (łańcuchowanie lub odesłanie).

Wymienione elementy mają zasadniczy wpływ na efektywność działania X.500 Directory. Wynika z nich, że zarządzający poddrzewem bazy X.500 serwer musi w ramach swoich danych posiadać prawidłowo skonstruowany zasób bazy informacyjnej (knowledge) o innych DSA, jest to realizowane przez umieszczenie odpowiednich odsyłaczy — adresów serwerów. W sytuacji, gdy zlecenie nie może być zrealizowane lokalnie, a równocześnie istnieje kilka serwerów zdolnych do wykonania zadania niezbędny jest dobrze funkcjonujący system optymalnego wyboru serwera, do którego zostanie wysłane zapytanie.

Przewidujemy przeprowadzenie szeregu testów i wyspecyfikowanie optymalnych ustawień parametrów wpływających na sposób realizacji połaczeń z innymi serwerami.

Patrząc na model funkcjonalny X.500 Directory należy pamiętać, że jest to przykład bazy, w której operacje typu przeszukanie i odczyt mają wyraźnie większy priorytet od innych (jak np. modyfikacja). Spowodowane jest to funkcjonowaniem aplikacji w konkretnej sytuacji, kiedy modyfikacje dokonywane są stosunkowo rzadko, a dynamika zmian jest głównie uzależniona od mobilności w ramach organizacji zbierających dane, nie od dynamiki sieci. Czas niezbędny do aktualizacji w bazie nie był w założeniach twórców standardu, elementem krytycznym.

Porównajmy teraz przedstawione wcześniej elementy efektywnościowe ze specyfiką bazy QUIPU X.500.

1. **Stopień rozproszenia.** Pomimo, że wydaje się oczywisty teoretyczny wpływ rozproszenia na działanie bazy, konieczne jest przeprowadzenie szczegółowych testów na znacznych zbiorach danych, aby określić optymalny model rozproszenia. W tym celu przewidujemy utworzenie szeregu przykładowych konfiguracji serwerów, pracujących na różnych poziomach drzewa informacji. Czas reakcji uśpionego serwera zależy od wielu czynników, miedzy innymi od wielkości przechowywanej bazy. Serwer w implementacji QUIPU przechowuje dane w pamięci operacyjnej, stąd konieczność przeprowadzenia analizy wpływu wielkości bazy na szybkość powrotu ze stanu uśpienia. Problem rozproszenia w naturalny sposób łaczy się z realizacją replikacji danych. Decyzję co do replikacji określonego fragmentu drzewa podejmuje administrator serwera biorcy w porozumieniu z administratorem serwera dawcy. Brakuje opisu algorytmów poruszania się po zreplikowanym drzewie X.500, a co za tym idzie administratorzy pozostawieni są samym sobie i mogą podejmować błędne decyzje dotyczące replikacji. Takie błędy mogą powodować znaczne spowolnienie dostępu do fragmentów drzewa informacji.

   Narzucające się rozproszenie danych będzie dawało efekt pozytywny pod warunkiem, że nie dojdzie do długich opóźnień w sieci i wspomnianych powyżej okresów uśpienia.

2. **Zgodność systemu serwerów z logiką układu bazy.** Fragmenty drzewa informacji DIT rezydujące na serwerach X.500 odzwierciedlają rzeczywisty podział organizacyjny świata. Projektując podział danych należy wyjść naprzeciw najbardziej popularnym sposobom przeszukiwań bazy i dażyć do realizowania wyłącznie niezbęnej ilości połączeń sieciowych. Istnieją międzynarodowe ogólne zalecenia dla administratorów DSA, dotyczące tworzenia poddrzew bazy X.500 ([BKL]). Głównie kładzie się nacisk na minimalizowanie głębokości poddrzewa (wielopoziomowe poddrzewo implikuje długie nazwy obiektów, które stają się mniej naturalne, a przede wszystkim zmniejszają efektywność, w przypadku częstych operacji przeszukiwania). Jeżeli w poddrzewie dominuje operacja odczytu, ilość poziomów nie ma znaczącego wpływu na jakość usługi. Jednocześnie dane powinny zostać tak rozmieszczone, by uprościć wykonywanie bardzo częstej operacji przeszukiwania jednopoziomowego (single–level search). Ponieważ obiekty podporządkowanego bieżącemu, kolejnego poziomu mogą być zlokalizowane w różnych DSA, operacja ta wiąże się na ogól z koniecznością kontaktu z wieloma serwerami. Aby temu zapobiec należy stosować odpowiednią replikację danych.

   Jako przykład wpływu rzeczywistego podziału administracyjnego na logikę układu bazy można rozważyć system serwerów regionalnych X.500. System taki przewiduje w zasadzie, że serwery regionalne przechowują dane o lokalnych instytucjach. W takiej sytuacji przeszukanie w zakresie jednej instytucji odbywałoby się na pojedynczym serwerze i byłoby najbardziej efektywne. Jednocześnie lokalni administratorzy bazy danych mają najlepszy dostęp do informacji. System taki nie uwzględnia jednakże istnienia dużych instytucji o oddziałach rozproszonych w wielu regionach kraju. Włączenie takich instytucji do systemu serwerów regionalnych wymaga rozważenia, czy bardziej typowe będą przeszukania w ramach danej (rozproszonej) instytucji, czy też w ramach konkretnego regionu. W zależności od odpowiedzi na to pytanie informacje o instytucji należy lokalizować lub rozpraszać. Do rozważenia pozostaje również odpowiedni dla takiej sytuacji system replikacji danych.

3. **Stosowanie odsyłaczy (aliasów).** Aliasy pozwalają na odwoływanie się do tego samego obiektu w różnych miejscach bazy X.500, dzięki temu te same obiekty nie muszą być definiowane wielokrotnie, a drzewo informacji jest spójne. Niestety, w implementacji QUIPU stosowanie odsyłaczy ma negatywny wpływ na sprawność algorytmu rozstrzygania nazwy obiektu (name resolution) oraz znacząco obniża efektywność algorytmu przeszukania, zwłaszcza w sytuacji, gdy konieczne jest generowanie wielokrotnych połączeń sieciowych, czyli, gdy obiekt jest zdefiniowany w innym serwerze.

Administratorzy muszą mieć klarowne informacje o tym jak aliasowanie wpływa na sprawność bazy, tak aby mogli podejmować trafne decyzje w momencie definiowania struktury swojego wycinka drzewa.

4. **Replikacja danych.** Mechanizm replikacji nie został określony w standardzie X.500 z roku 1988. Jego wprowadzenie jest jednak niezbędne w celu uzyskania niezawodności i efektywności serwisu, na co wskazano w [HK91b]. Rozszerzenia standardu dotyczące replikacji i służacych do tego celu dodatkowych rozproszonych operacji zdefiniowane są w [HK91c]. Stały się one podstawą do implementacji replikacji w oprogramowaniu QUIPU.

   QUIPU dopuszcza trzy techniki replikacji:

   - Pobieranie kopii wybranego fragmentu drzewa (shadowing) — technika ta jest stosowana w celu lokalnego przechowywania tych części bazy X.500, do których przewiduje się częste odwoływanie poprzez operacje przeszukiwania lub odczytu. Polega na łączeniu się serwera biorcy z serwerem dawcą i wykonaniu operacji transmisji bloku danych.
   - Automatyczną aktualizację danych dotyczących wybranych, specjalnych obiektów bazy (spot-shadowing). Podstawowa informacja (master copy) o takich obiektach jest wprawdzie trzymana w ramach opiekującego się tym fragmentem poddrzewa serwera, jednak *de facto* danymi tymi zarządza inny DSA. W celu aktualizacji serwer łączy się z odpowiednim DSA i dokonuje odczytu danych dotyczących konkretnego obiektu. Przykładem obiektów podlegających tego typu replikacji jest klasa `quipuDSA`, definiująca serwery X.500 w implementacji QUIPU.
   - Czasowe utrzymywanie odebranych z innych serwerów danych w „pamięci podręcznej” (caching). Należy zwrócić uwagę na fakt, że ta technika replikacji danych jest stosowana zarówno po stronie serwera DSA, jak i po stronie interfejsu użytkowego DUA. Z punktu widzenia serwera możliwość korzystania z wcześniej odebranych informacji implikuje konieczność dezaktualizacji kopii danych po określonym czasie, a także decydowania czy kopia danych może zostać użyta do odpowiedzi na pytanie postawione przez inny interfejs użytkowy.

   Replikacja ma oczywiste czynniki uboczne:

   - doprowadza do możliwych czasowych niekonsystencji w bazie,
   - powieksza ilość danych przechowywanych na poszczególnych serwerach.

Efektywność replikacji zależy zatem od doboru powielanej informacji tak, aby dopasować ją do najbardziej prawdopodobnego stylu zapytań, nie generując jednocześnie nadmiernych ilości danych.

Ponieważ dane replikowane niekoniecznie muszą być pewne, w sytuacjach, gdy znalezienie poprawnej odpowiedzi jest krytyczne, trzeba korzystać z danych podstawowych, a nie kopii. W sytuacjach, gdy przewiduje się niezbędność pracy na danych pewnych, replikacja jest bezcelowa. Należy podkreślić, że korzystanie z danych niepewnych jest w wielu zastosowaniach dopuszczalne i nie prowadzi do utraty prawdziwej informacji. Jeżeli możliwa jest późniejsza weryfikacja poprawności odpowiedzi, nie jest przeciwwskazane podanie (jako pierwszego przybliżenia) informacji niepełnej lub nawet błędnej. Np. informacja o routingu poczty otrzymana w celu dostarczenia konkretnego listu zostanie zweryfikowana przez system nadawczy. Błąd przy dostarczeniu listu wygeneruje odpowiednią informację i będzie sygnałem, że konieczne jest odwołanie się do danych pewnych. W przypadkach baz rzadko modyfikowanych, sytuacje takie są sporadyczne i jak najbardziej uzasadnione jest godzenie się na takie pojedyncze „awarie" systemu.

5. **Sprawność poszczególnych serwerów.** Czynnik ten jest decydujący, gdy mamy do czynienia z bazą rozproszoną — korzystanie z usług serwerów zawodnych doprowadza do znacznego spowolnienia pracy systemu. Standard 1988 nie zawiera ustaleń odnośnie jakości operacyjnej serwerów (QOS — Quality of Service). Atrybuty niezbędne do wprowadzania takich informacji do bazy X.500 zostały zdefiniowane w [BK91], zalecaną formę zastosowania rozszerzeń standardu w implementacjach X.500 opisano w [HK91d].

   W celu kontrolowania jakości operacyjnej partnerów obiekt reprezentujący w DIT serwer DSA może dodatkowo zostać opisany poprzez atrybut `DSAQuality`, w którym określa się stopień świadczonych usług (eksperymentalny, usługa pilotowa, pełny serwis). Została również dodana możliwość wyspecyfikowania jakości danych umieszczonych w określonym fragmencie drzewa, zarówno z punktu widzenia ich ilości jak i jakości konkretnych atrybutów opisujących obiekt.

   Istnieją już pierwsze interfejsy użytkowe analizujące jakość serwisu. W trakcie badań wydajnościowych X.500 zamierzamy przeprowadzić odpowiednie testy, głównie w celu sporządzenia wytycznych co do ustawień atrybutów dla administratorów serwerów, ale także by zebrać doświadczenia niezbędne do tworzenia interfejsów użytkowych w pełni wykorzystujących kontrolę jakości serwisu.

6. **Algorytmy generowania połączeń.** Dla uzyskania pełnej funkcjonalności bazy X.500 należy prawidłowo dobrać algorytm wyboru serwera,

predysponowanego do wykonania zadania, które nie może zostać zrealizowane lokalnie. Algorytm taki opiera się w implementacji QUIPU na utrzymywanej w ramach DSA informacji o jakości operacyjnej partnerów oraz na odpowiednim doborze parametrów administracyjnych, dających możliwość określenia sposobu interakcji pomiędzy DSA i adresów preferowanych serwerów. Czynnik ten ma również istotny wpływ na mechanizm replikacji danych. Przewidujemy potrzebę analizy skuteczności zrealizowanego algorytmu i ewentualnej jego weryfikacji do lokalnych potrzeb.

7. **Stosowanie standardowych zapytań.** W przypadku X.500 specyfika bazy daje możliwość przewidywania najbardziej typowych zapytań i dostosowywania do nich układu samej bazy, indeksowania itp. Istnieją możliwości blokowania zapytań bardzo czasochłonnych, np. z reguły zabrania sie standardowego przeszukiwania drzewa w zakresie całego kraju, jako że na ogół niosłoby to za sobą konieczność kolejnego uruchamiania wielu serwerów i, w konsekwencji, długiego czasu reakcji. Samo zabezpieczenie administracyjne bazy przed obciążającymi system przeszukiwaniami nie jest wystarczające. Bardzo istotną rolę odgrywa tu twórca interfejsu użytkownika, który musi uwzględnić jakiego typu zapytania są obsługiwane szybko, a jakie wolno i starać się realizować obsługę optymalnie.

8. **Dokładność dopasowywania do wzorców.** Jest to bardzo istotna cecha systemu, poprawnie skonstruowany algorytm dopasowywania przybliżonego znacznie podnosi wydajność. Różnorodne testy, na podstawie analizy informacji o przebiegu realizacji operacji konkretnych przeszukań (zapisy typu `log`) dadzą obraz działania algorytmu w konkretnej implementacji QUIPU. Wydaje się również niezbędne zrozumienie zasady działania odpowiednich algorytmów. Głównie uwagi wymaga algorytm SOUNDEX, odpowiedzialny za wyszukiwanie przybliżone, realizowane w QUIPU dla potrzeb języka angielskiego i dające w polskich warunkach niespodziewane rezultaty.

9. **Indeksowanie.** Implementacja QUIPU dopuszcza możliwość indeksowania wskazanego poddrzewa (subtree) lub jednego poziomu drzewa (siblings) oraz optymalizowania przeszukiwań struktur danych w pamięci według podanego klucza — typu atrybutu.

   Należy pamiętać, że indeksowanie ponad miarę może doprowadzić do efektu odwrotnego niż zamierzony. Na celowość korzystania z indeksu wpływa też to, jak współpracuje z nim konkretna aplikacja, istnieją przypadki, w których uszczegółowienie pytania może spowodować, że zamiast korzytania z indeksu baza przeprowadzi standardowe przeszukanie. Oznacza to, że administrator bazy musi być w stanie przewidzieć, jakie zapytania są standardowo generowane i do nich dobrać sposób indeksowania. Decyzję o indeksowaniu bazy danych należy podejmować na podstawie analizy częstotliwości stosowania różnego rodzaju przeszukiwań bazy (jednopoziomowe lub całego poddrzewa) oraz typów atrybutów, według których przeszukiwania występują. Jak już wspominaliśmy nieodpowiednie indeksowanie nie przyspiesza działania bazy, a nawet je spowalnia na skutek zwiększenia ilości przechowywanych danych. Również w tym zakresie potrzebne są szczegółowe wytyczne co do optymalnych modeli.

5 Narzędzia badania niezawodności systemu X.500

Przeprowadzenie testowania wpływu wymienionych powyżej elementów sterujących wymagać będzie tworzenia przykładowych instalacji, w których uwypuklone zostaną odpowiednie specyficzne składowe bazy. W ten sposób będzie można również zasymulować współpracę większej ilości serwerów, umieszczonych na różnych poziomach drzewa informacji X.500. Konieczne będzie równocześnie stworzenie narzędzi do badania zachowania bazy. Narzędzia takie potrzebne są w instalacjach testowych, ale należy pamiętać, że pewne metody monitorowania pracy bazy potrzebne są również w codziennej, docelowej eksploatacji serwisu X.500.

Obecnie w projekcie PARADISE stosuje sie monitorowanie serwerów krajowych sieci X.500, narzędzie do tego służące nie jest jednak powszechnie dostępne, a na dodatek mocno niedoskonałe, jako że generuje jedynie informacje na temat zachowania się bazy w całości, nie analizując przy tym powodów ewentualnych awarii. Nie są na przykład prowadzone statystyki uwzględniające wpływ awarii samej sieci Internet na bazę X.500. Utworzenie dobrego zestawu narzędzi monitorujących będzie jednym z celów przyszłych prac.

Podobnie w przypadku generowania statystyk korzystania z bazy — istnieje oprogramowanie sporządzające raporty dla poszczególnych serwerów, brakuje natomiast systemu gromadzącego statystyki dotyczące wiekszych wycinków drzewa i odpowiednio przetwarzającego tego typu dane.

Bibliografia

[ASN1] D. Steedman, ASN.1 The Tutorial and Reference, 1990.

[BK91] P. Barker and S.E. Kille. The COSINE and Internet X.500 schema, RFC 1274, University College London, November 1991.

[BKL] P. Barker, S. E. Kille, T. Langgenhager: Naming and Structuring Guidelines for X.500 Directory Pilots, May 1994.

[DNS] P.V. Mockapetris, Domain names — Concept and Facilities, RFC 1034, November 1987;

[HK91a] S. E. Hardcastle-Kille, X.500 and Domains, RFC 1279, November 1991.

[HK91b] S. E. Hardcastle-Kille, Replication Requirements, RFC 1275, November 1991.

[HK91c] S. E. Hardcastle-Kille, Replication and distributed operation extentions to provide an Internet Directory using X.500, RFC 1276, November 1991.

[HK91d] S. E. Hardcastle-Kille, Handling QOS (Quality of service) in the Directory, March 1991.

[PVM] P.V. Mockapetris, Domain names — Implementation and Specification, RFC 1035, November 1987;

[QUIPU] C.J. Robbins and S.E. Kille, The ISO Development Environment: User's Manual, Volume 5: QUIPU, July 1992.

[ROSE] Marshall T. Rose, The Little Black Book: Mail Bounding with OSI Directory Services, Prentice Hall, 1992.

[X500] Data Communication Networks: Directory, Recommendations X500-X.521, CCITT, Fascile VIII.8 of Blue Book.