Wprowadzenie

Plan wykładu

Celem wykładu jest wprowadzenie studenta w tematykę przetwarzania rozproszonego. Wykład rozpocznie się od krótkiego wprowadzenia, którego celem będzie zapoznanie słuchacza z bieżącym stanem wiedzy w tej dziedzinie. Następnie przedstawione zostaną proste definicje dotyczące środowiska przetwarzania. W kolejnych punktach wykładu przedstawione zostaną pewne reprezentatywne środowiska przetwarzania rozproszonego. Pierwszym takim środowiskiem będzie Internet. Przedstawiona zostanie jego krótka historia i obecne trendy rozwoju. Następnie krótko scharakteryzowane zostanie środowisko typu GRID. W kolejnym punkcie wykładu opisane zostaną popularne obecnie projekty nazywane potocznie @HOME. Jako przykłady posłużą projekty Seti@HOME, CureCancer, FightAnthrax. Na zakończenie omówiony zostanie projekt środowiska rozproszonego wyszukiwarki internetowej Google.

Cechy systemów rozproszonych

Rozproszony system informatyczny obejmuje środowisko przetwarzania rozproszonego oraz zbiór procesów rozproszonych. Na środowisko przetwarzania rozproszonego składają się autonomiczne jednostki przetwarzające (węzły) zintegrowane siecią komunikacyjną (łączami transmisyjnymi). Każda z jednostek przetwarzających jest wyposażona w procesor, lokalną pamięć i własne oprogramowanie zarządzające. Proces rozproszony (przetwarzanie rozproszone) jest natomiast działaniem wynikającym ze współbieżnego i skoordynowanego wykonania zbioru procesów sekwencyjnych realizujących wspólny cel przetwarzania w środowisku rozproszonym. Systemy rozproszone charakteryzują się brakiem pamięci współdzielonej przez węzły i dlatego komunikacja odbywa się w nich tylko za pomocą wymiany wiadomości (komunikatów). Wyróżnia się systemy rozproszone asynchroniczne i synchroniczne. W systemach asynchronicznych, poszczególne węzły wykonują operacje z różnymi prędkościami w takt niezależnych zegarów, a czas transmisji wiadomości jest skończony lecz nieznany. W systemach synchronicznych natomiast, działanie wszystkich węzłów jest zsynchronizowane wspólnym zegarem lub też maksymalne opóźnienie transmisji jest ograniczone i znane a priori.

Ze względu na rosnące wymagania dotyczące efektywności i niezawodności systemów informatycznych, rozwój asynchronicznych systemów rozproszonych – w tym sieci komputerowych, systemów przetwarzania równoległego z rozproszoną pamięcią, rozproszonych środowisk programowania, rozproszonych systemów baz danych czy wreszcie Internetu i niezliczonych już jego aplikacji - jest obecnie jednym z najważniejszych i najbardziej obiecujących kierunków rozwoju informatyki. Stwierdzenie takie uzasadniają szczególne właściwości tych systemów:

- duża wydajność (duża moc obliczeniowa i maksymalna przepustowość, krótki czas odpowiedzi) wynikająca z możliwości jednoczesnego udziału wielu jednostek i systemów w realizacji wspólnego celu przetwarzania;
- duża efektywność inwestowania (względnie niskie koszty niezbędne do uzyskania wymaganej wydajności systemu) wynikająca z korzystnego stosunku ceny do wydajności;
- wysoka sprawność wykorzystania zasobów (wysoki stopień wykorzystania zasobów i współczynnik jednoczesności) wynikająca z możliwości współdzielenia stanowisk usługowych, specyficznych urządzeń, programów i danych przez wszystkich użytkowników systemu, niezależnie od fizycznej lokalizacji użytkowników i zasobów;
- skalowalność (możliwość ciągłego i praktycznie nieograniczonego rozwoju systemu bez negatywnego wpływu na jego wydajność i sprawność) wynikająca z modularności systemu i otwartości sieci komunikacyjnej;

- wysoka niezawodność (odporność na błędy) wynikająca z możliwości użycia zasobów alternatywnych;
- otwartość funkcjonalna (łatwość realizacji nowych, atrakcyjnych usług komunikacyjnych, informatycznych i informacyjnych) wynikająca z integracji otwartej sieci komunikacyjnej i efektywnych, uniwersalnych jednostek przetwarzających.

Problemy związane z konstrukcją systemów rozproszonych

Pełne wykorzystanie powyższych cech systemów rozproszonych wymaga jednak efektywnego rozwiązania wielu problemów. Podstawowa trudność związana jest z konstrukcją poprawnych, efektywnych i niezawodnych algorytmów rozproszonych opisujących procesy rozproszone. Charakterystyczny dla systemów rozproszonych asynchronizm komunikacji i działania procesów implikuje niedeterminizm przetwarzania. Dodatkowo brak wspólnej pamięci ogranicza dostępne wprost mechanizmy komunikacji i synchronizacji. Dlatego też konstrukcja i weryfikacja procesów (algorytmów) rozproszonych ma swoją specyfikę i rodzi wiele trudnych problemów, takich jak:

- optymalne zrównoleglenie algorytmów przetwarzania;
- ocena poprawności i efektywności algorytmów rozproszonych;
- alokacja zasobów rozproszonych;
- synchronizacja procesów;
- ocena globalnego stanu przetwarzania;
- realizacja zaawansowanych modeli przetwarzania;
- niezawodność;
- bezpieczeństwo.

Jak wiadomo, problem zrównoleglenia sprowadza się do takiej transformacji algorytmu rozwiązywania zadania obliczeniowego w zbiór wzajemnie powiązanych procesów (wątków) wykonywanych równolegle albo sekwencyjnie, by zminimalizować najdłuższą ścieżkę obliczeń sekwencyjnych, abstrahując od ograniczeń fizycznych i funkcjonalnych rzeczywistego środowiska przetwarzania.

Trudność oceny poprawności i efektywności algorytmów rozproszonych związana jest z koniecznością analizy wszelkich możliwych realizacji niedeterministycznego w ogólności algorytmu rozproszonego.

Problem alokacji zasobów polega na takim przydziale (alokacji) dostępnych zasobów (procesorów, pamięci, urządzeń wejścia/wyjścia, danych, programów itd.) do procesów (zadań), by przy spełnieniu przyjętych bądź narzuconych warunków podzielności i ograniczeń kolejnościowych, zoptymalizować wybrane kryterium efektywności.

Problem synchronizacji procesów, związany w ogólności z kooperacją procesów lub ich współzawodnictwem o dostęp do wspólnych zasobów, polega na realizacji w asynchronicznym środowisku rozproszonym mechanizmów umożliwiających wzajemne oddziaływanie procesów na ich względne prędkości przetwarzania, w celu dochowania ograniczeń kolejnościowych i zagwarantowania poprawności obliczeń.

Problem oceny stanu globalnego polega natomiast na wyznaczaniu wartości parametrów lub predykatów związanych z globalnymi stanami procesów rozproszonych. W asynchronicznym środowisku rozproszonym wyznaczanie stanu globalnego jest trudne i w ogólności niemożliwe bez wstrzymywania przetwarzania.

Problem realizacji zaawansowanych modeli przetwarzania sprowadza się do transparentnej realizacji w środowisku rozproszonym systemu stosowniejszego dla danego zastosowania lub

wygodniejszego z punktu widzenia użytkownika (rozproszona pamięć współdzielona, synchronizm przetwarzania lub komunikacji, przetwarzanie transakcyjne itp.).

Problemy niezawodności i bezpieczeństwa związane są z potrzebą zagwarantowania wymaganego poziomu jakości pracy systemu niezależnie od nieuniknionych błędów, przypadkowych lub celowych działań mogących prowadzić do zniszczenia systemu, czy naruszenia poufności i autentyczności informacji.

Motywy

Mimo prowadzonych od lat intensywnych badań dotyczących wymienionych wyżej problemów, dla wielu z nich nie znaleziono jeszcze w pełni satysfakcjonujących rozwiązań. Znaczenie oraz aktualność problematyki konstrukcji i analizy systemów rozproszonych wynika jednak nie tylko z różnorodności otwartych, ciekawych problemów badawczych lecz również, a może przede wszystkim, z ogromnego rzeczywistego zapotrzebowania na systemy rozproszone w wielu dziedzinach zastosowań, podsycanego dostępnością środków technicznych i sukcesami licznych już rozwiązań praktycznych.

Problemy te sprowadzają się w istocie do odpowiedniego wyznaczenia stanów lokalnych procesów składowych przetwarzania rozproszonego oraz ewentualnie stanów kanałów komunikacyjnych, i utworzenia na ich podstawie stanu globalnego, obejmującego wszystkie składniki systemu. Wyznaczony stan globalny może być dalej analizowany w celu wykrycia jego specyficznych parametrów, definiowanych często w formie predykatów.

Podstawowa trudność tego problemu wynika z właściwości środowiska rozproszonego, a w szczególności z braku globalnego zegara oraz asynchronizmu przetwarzania i komunikacji. Dowolny proces może bowiem łatwo określić jedynie swój stan lokalny, lecz uzyskanie informacji o stanach innych procesów przetwarzania oraz o stanach kanałów wymaga wymiany wiadomości. Wobec asynchronizmu systemu, otrzymane od innych procesów stany lokalne (składowe stanu globalnego) reprezentują w ogólności stany odnoszące się do różnych momentów czasu globalnego z przeszłości. Stąd też ich złożenie może nie mieć żadnego związku z jakimkolwiek stanem globalnym osiągniętym w rzeczywistości przez system. Dalsza analiza takiego złożenia nie ma więc większego znaczenia czy nawet sensu.

Prowadzone przez lata badania wykazały jednak, że w asynchronicznym środowisku rozproszonym można z powodzeniem wyznaczać pewne stany specyficzne (np. tak zwane stany stabilne), jak również pewne aproksymacje stanu globalnego (nazywane obrazami lub konfiguracjami spójnymi), wystarczające w wielu wypadkach.

Rozproszony system informatyczny

Rozproszony system informatyczny obejmuje środowisko przetwarzania rozproszonego (węzły, łącza) oraz procesy rozproszone (zbiory procesów sekwencyjnych realizujących wspólne cele przetwarzania).

Środowisko przetwarzania rozproszonego

Środowisko przetwarzania rozproszonego jest zbiorem autonomicznych jednostek przetwarzających (węzłów), zintegrowanych siecią komunikacyjną (środowiskiem komunikacyjnym, łączami komunikacyjnymi, łączami transmisyjnymi).

Komunikacja w środowisku przetwarzania rozproszonego

W środowisku przetwarzania rozproszonego komunikacja między węzłami możliwa jest tylko przez transmisję pakietów informacji (wiadomości, komunikatów) łączami komunikacyjnymi.

Zegary w środowisku przetwarzania rozproszonego

Jednostki przetwarzające realizują przetwarzanie z prędkością narzucaną przez lokalne zegary. Jeżeli zegary te są niezależne, to mówimy, że węzły działają asynchronicznie. Jeżeli natomiast zegary te są zsynchronizowane, lub istnieje wspólny zegar globalny dla wszystkich węzłów, to mówimy, że węzły działają synchronicznie.

Definicja węzła

Jednostka przetwarzająca $N_i \in \mathcal{N}$ (węzeł) jest elementem środowiska przetwarzania rozproszonego obejmującym:

- procesor,
- pamięć lokalną,
- interfejs komunikacyjny.

Procesor wykonuje automatycznie program zapisany w pamięci lokalnej, gdzie pamiętane są również dane. Interfejs komunikacyjny umożliwia węzłom dostęp do łączy, i tym samym wzajemną wymianę informacji - komunikatów (ang. *message passing*, *message exchange*), oraz komunikację z użytkownikiem.

Łącze komunikacyjne

Łącze komunikacyjne jest elementem pozwalającym na transmisję informacji między interfejsami komunikacyjnymi odległych węzłów. Wyróżnia się łącza jedno i dwukierunkowe.

Bufory łącza

Wyposażone są one na ogół w bufory, których rozmiar określany jest jako pojemność łącza (ang. *link capacity*). Jeżeli łącze nie posiada buforów (jego pojemność jest równa zero), to mówimy o łączu niebuforowanym, w przeciwnym razie - o buforowanym. Dla ułatwienia analizy często przyjmuje się, że pojemność łączy jest nieskończona.

Kolejność odbierania komunikatów

Zwykle, kolejność odbierania komunikatów przesyłanych między węzłami jest zgodna z kolejnością ich wysłania. Jeśli warunek ten jest spełniony, to łącze nazywamy łączem FIFO (ang. *First-In-First-Out*), w przeciwnym razie - nonFIFO.

Niezawodność łącza

Łącza mogą gwarantować również, w sposób niewidoczny dla użytkownika, że żadna wiadomość nie jest tracona, zwielokrotniana (duplikowana) lub zmieniana - są to tzw. łącza niezawodne (ang. *reliable, lossless, duplicate free, error free, uncorrupted, no spurious*).

Czas transmisji w łączu niezawodnym

Czas transmisji w łączu niezawodnym (ang. *transmission delay, in-transit time*) może być ograniczony lub jedynie określony jako skończony lecz nieprzewidywalny. W pierwszym przypadku mówimy o transmisji synchronicznej lub z czasem deterministycznie ograniczonym (w szczególności równym zero), a w drugim - o transmisji asynchronicznej lub z czasem niedeterministycznym.

Struktura środowiska przetwarzania

Struktura środowiska przetwarzania rozproszonego jest często przedstawiana jako graf:

$$\mathcal{G} = \langle \mathcal{V}, \mathcal{A} \rangle$$

w którym wierzchołki grafu $V_i \in \mathcal{V}$ reprezentują jednostki przetwarzające $N_i \in \mathcal{N}$, a krawędzie $(V_i, V_j) \in \mathcal{A}$, $\mathcal{A} \subseteq \mathcal{V} \times \mathcal{V}$, grafu niezorientowanego lub łuki $\langle V_i, V_j \rangle \in \mathcal{A}$ grafu zorientowanego, reprezentują odpowiednio łącza dwu lub jednokierunkowe. Tak zdefiniowany graf odpowiada strukturze środowiska rozproszonego i nazywany jest grafem środowiska lub topologią środowiska.

Przykłady topologii

Struktura środowiska przetwarzania rozproszonego może przybierać najróżniejsze formy, od najprostszych, czyli struktur płaskich w których węzły połączone są szeregowo, poprzez bardziej skomplikowane struktury drzewiaste lub struktury w kształcie gwiazdy. Innym przykładem dość często występującym w praktyce jest struktura pierścienia. Kolejnym przykładem są struktury w których zakłada się iż każdy z węzłów połączony jest z każdym innym węzłem biorącym udział w przetwarzaniu. Oprócz struktur dwu wymiarowych popularne też są struktury przestrzenne takie jak sześciany, hiperkostki, torusy i inne. Przykłady różnych typów topologii zostały zaprezentowane na slajdzie.

Skala przetwarzania

Biorąc pod uwagę skalę przetwarzania możemy podzielić systemy rozproszone na następujące grupy:

- Systemy końcowe (ang. end systems)
  stosunkowo mała liczba komponentów dobrze zintegrowanych ze sobą, zaprojektowanych z myślą o wydajnej współpracy
- Klastry (ang.clusters)
  zwiększona skala przetwarzania, konieczność zastosowania nowych algorytmów, zredukowana integracja pomiędzy elementami
- Sieci intranetowe (ang. intranets)
  różnorodność, wiele centrów zarządzania, brak wiedzy o stanie globalnym
- Internet
  brak scentralizowanego zarządzania, rozproszenie geograficzne, międzynarodowy charakterem sieci

Klasy zastosowań

Wyróżniamy następujące klasy zastosowań

- Aplikacje wykorzystujące przetwarzanie rozproszone na wielu jednostkach obliczeniowych (ang. *distributed supercomputing*)
  bardzo duże i skomplikowane obliczeniowo problemy, wymagające dużej mocy CPU, pamięci, itd.
- Aplikacje wymagające dużej przepustowości (ang. *high throughput*)
  technika wykorzystywania zasobów, które są dostępne w celu zwiększenia przepustowości
- Aplikacje „na żądanie" (ang. *on demand*)
  zdalne zasoby zintegrowane z lokalnym przetwarzaniem, często przez ograniczony okres czasu
- Aplikacje intensywnie przetwarzające dane (ang. *data intensive*)
  synteza nowych informacji z dużych wolumenów danych pochodzących z różnych źródeł
- Aplikacje umożliwiające współpracę (ang. *collaborative*)
  wspieranie pracy grupowej umożliwiającej szybsze osiągnięcie założonych celów badawczych.

TOP500 [http://www.top500.org]

- Lista najszybszych maszyn na świecie (aktualizowana 2 razy w roku)
- W czerwcu 2008 roku na I miejscu: IBM BlueGene/L Roadrunner (122400 CPU, 1026 TFlop/s, 98 TB RAM)

  Maszyna ta znajduje się w Los Alamos, US Department of Energy, National Laboratory.
- 5 pierwszych miejsc dla USA (w tym 3 początkowe dla IBM), 41,8 % maszyn z listy to produkty IBM
- Ostatni komputer na liście ma moc przekraczającą 2 TFlop/s !
- NEC Earth Simulator, (35.86 TFlop/s), który był zwycięzcą rankingu przez 5 edycji do roku 2003, zanim został zdetronizowany przez IBM BlueGene/L zajmuje obecnie 10 pozycję
- W 1993 roku pierwszy na liście komputer miał moc szacowaną na 124 GFlop/s.
- 85.4% maszyn z listy TOP500 pracuje pod kontrolą systemu Linux, 5% pod kontrolą różnych wariantów systemu Unix, 1% pod kontrolą systemu Windows

Internet – środowisko rozproszone

1957 – utworzenie agencji ARPA (jako przeciwwaga na wystrzelenie sputnika przez ZSRR), która będzie głównym promotorem ARPANET'u.

1964 – Ukazuje się raport „On Distributed Communication Networks" autorstwa P. Barana z RAND Corp. Propozycja decentralizacji sieci komputerowej, która ma możliwość funkcjonowania w przypadku awarii znacznej ilości węzłów. Raport ten stał się podstawą do utworzenia projektu ARPANET.

1969 – Powstanie ARPAnet'u, sieci czterech komputerów. W 1971 sieć ta liczyła sobie 13 węzłów, a w 1973 roku - już 35. Początkowo sieć ARPAnet zostaje wykorzystywana do komunikacji między naukowcami, przesyłania listów elektronicznych i wspólnej pracy nad projektami. Powstaje pierwszy dokument z serii RFC, napisany przez Steve Crockera.

1970 – Uruchomiony pierwszy serwer FTP.

1971 – Początki poczty elektronicznej. Ray Tomlinson wysyła pierwszą wiadomość elektroniczną.

1972 – Powstaje Telnet, aplikacja pozwalająca na zdalną pracę na odległych komputerach - połączenie się z nimi i uruchamianie programów.

1973 –ARPANET staje się siecią międzynarodową (University College of London w Wielkiej Brytanii i Royal Radar Establishment w Norwegii)

1974 – Po raz pierwszy pojawia się słowo Internet, w opracowaniu badawczym dotyczącym protokołu TCP, napisanym przez Vintona Cerfa (znany jako „ojciec Internetu") i Boba Kahna „A Protocol for Packet Intercommunication".

1977 – TheoryNet łączy pocztą elektroniczną stu naukowców: powstaje pierwsza lista dyskusyjna (mailing list). Powstają protokoły TCP i IP.

1978 – W Chicago powstaje pierwszy BBS (bulletin-board system).

1979 – Powstaje Usenet, tekstowe grupy dyskusyjne - stworzony przez studentów Toma Truscotta, Jima Ellisa i Steve Bellovina.

1982 – Pojawiają się pierwsze uśmieszki (smileys), :) .

1983 – Z sieci ARPANET zostaje wydzielona część wojskowa tworząc MILNET. Hosty i sieci zaczynają używać protokołu TCP/IP. Powstaje właściwy Internet.

1984 – Powstaje NSFNET, sieć coraz szybszych superkomputerów wykorzystywanych do celów naukowych (finansowana przez NSF). Powstaje specyfikacja DNS, NNTP (Network News Transfer Protocol).

1985 – Rejestracja pierwszej domeny komercyjnej - symbolics.com (dla firmy tworzącej programy i sprzęt dla języka programowania Lisp). Powstaje America Online, słynna usługa on-line.

1986 – Hierarchizacja grup dyskusyjnych Usenet (comp.*, news.* i misc.*). W rok później John Gilmore i Brian Reid, niezadowoleni z istniejących hierarchii, tworzą hierarchię alt.* - dziś skupiającą najwięcej grup dyskusyjnych.

1988 –Jarkko Oikarinen tworzy Internet Relay Chat (IRC), system internetowych pogawędek.

1989 – Formalnie przestaje istnieć ARPANET. Internet rozwija się dalej.

1990 – Tim Berners-Lee tworzy World Wide Web, system pozwalający autorom na połączenie słów, zdjęć i dźwięku, początkowo pomyślany dla wsparcia naukowców zajmujących się fizyką w CERN. W maju Polska zostaje przyjęta do EARN, części sieci BITNET- mamy dostęp do sieci.

1991 – Paul Linder i Mark P. McCahil z uniwersytetu w Minessocie opracowali system Gopher. Powstaje Archie usługa wyszukiwawcza. 23 sierpnia przychodzi z Hamburga pierwsza odpowiedź na pocztę elektroniczną wysłaną z Polski. W styczniu liczba użytkowników sieci w Polsce przekracza 2000. 11 kwietnia 1991 roku sieci WAWPOLIP zostaje przyznana klasa adresowa. W końcu sierpnia uruchomione zostaje pierwsze połączenie intenretowe z Warszawy do Kopenhagi, z inicjatywy prof. dr hab. Antoniego Kreczmara, dr Rafała Pietraka i dr Krzysztofa Helera. W akcji uruchomienia połączenia bierze udział również Marcin Gromisz. Cyfronet w Krakowie buduje Internet, korzystając z przemyconego przez COCOM routera CISCO.

1992 – Powstaje Internet Society, organizacja koordynująca rozwój i działanie Internetu. W Polsce oddano do użytku sieć pakietową TP SA pod nazwą Polpak.

1993 – Pojawia się Mosaic, pierwsza graficzna przeglądarka World Wide Web. Tworzy ją zespół: Marc Andreessen, Eric Bina i inni studenci NCSA. Dzieki niej znacznie wzrasta

popularność Internetu i World Wide Web. Przedstawiciel Microsoft stwierdza, że "większość ludzi nigdy nie będzie potrzebować modemów szybszych niż 2400 bps". W Internecie pojawia się Biały Dom. W Polsce powstaje Naukowa i Akademicka Sieć Komputerowa - NASK, jako jednostka badawczo-rozwojowa.

1994 – David Filo i Jerry Yang tworzą Yahoo! Jako spis interesujących ich miejsc w Internecie; 12 kwietnia firma prawnicza Canter & Siegel wysyła do sieci, na sześć tysięcy grup dyskusyjnych, spam - posting promujący jej usługi w loterii pozwoleń na pracę w Stanach; W Polsce rusza program podłączania szkół średnich do Internetu - Internet dla Szkół - w którym działa między innymi Jacek Gajewski. W marcu 1996 Compuserve próbuje pobierać opłaty za wykorzystanie przez programistów formatu GIF, najbardziej popularnego formatu graficznego na stronach internetowych.

1995 – powstaje Netscape Navigator, (posiadając w swoim czasie do 80 procent rynku). W lipcu Microsoft ogłasza wprowadzenie Microsoft Network, MSN, usługi online z oprogramowaniem dostępnym w każdej kopii Windows 95. Rozpoczyna się "wojna przeglądarek".

1996 – Pojawia się system WebTV, brakujące ogniwo pomiędzy Internetem a telewizją. W maju Procter & Gamble staje się pierwszym dużym reklamodawcą internetowym, który zamierza płacić nie za "spojrzenia" (eyeballs), ale za "kliknięcia" (click-throughs) na banner reklamowy.

Internet – liczba użytkowników sieci

Liczba użytkowników sieci Internet [mln]			
1995	2002	2005	? (2015)
45	445.9	1 080	2 000

Źródło: Computer Industry Almanac 01.2006

Internet – dostęp do sieci

Dostęp do sieci Internet wg krajów [mln osób]	
USA	**197.8**
Chiny	**119.5**
Japonia	**86.3**
Indie	**50.6**
Niemcy	**46.3**
Polska	**10.6**

Źródło: Computer Industry Almanac 01.2006

GRID

"*A computational grid is a hardware and software infrastructure that provides dependable, consistent, pervasive, and inexpensive access to high-end computational capabilities*"

Infrastruktura oparta na sprzęcie i oprogramowaniu, która umożliwia niezawodny, spójny, powszechny i niedrogi dostęp do wydajnego przetwarzania.

/ Ian Foster /

Grid – system, który integruje i zarządza zasobami będącymi pod kontrolą różnych domen (od instytucji po system operacyjny), i połączonymi siecią używa standardowych, otwartych protokołów i interfejsów ogólnego przeznaczenia (odkrywania i dostępu do zasobów, autoryzacji, uwierzytelniania) oraz dostarcza usług odpowiedniej jakości.

Grid - skoordynowane, bezpieczne, współdzielenie zasobów, oraz rozwiązywanie problemów w dynamicznych, obejmujących wiele instytucji wirtualnych organizacjach.

1995 – pierwsze eksperymenty

1997 – projekt UNICORE, pierwszy prototyp SETI@Home

1998 – „The Grid – Blueprint for a New Computing Infrastructure" Ian Foster, Carl Kesselman

1999 – Global Grid Forum [http://www.gridforum.org/]

GRID – cechy

- **Usługa wiarygodna:** użytkownicy żądają pewności, że otrzymają przewidywalny, nieprzerwany poziom wydajności dzięki różnym elementom tworzącym GRID.
- **Usługa powszechnie dostępna (wszechobecna):** usługa zawsze powinna być dostępna, niezależnie od tego gdzie znajduje się użytkownik tej usługi.
- **Usługa relatywnie tania (opłacalna):** dostęp do usługi powinien być relatywnie tani, tak by korzystanie z takiej usługi było atrakcyjne także z ekonomicznego punktu widzenia.
- **Usługa spójna:** potrzebny jest standardowy serwis, dostępny poprzez standardowe interfejsy, pracujący ze standardowymi parametrami.

Trzy spojrzenia na GRID:

Użytkownik

Wirtualny komputer, który minimalizuje czas wykonania obliczeń oraz zapewnia dostęp do zasobów

Programista

Zestaw narzędzi i interfejsów zapewniających przezroczysty dostęp do danych

Administrator

Środowisko umożliwiające monitorowanie, administrowanie i bezpieczne używanie rozproszonych zasobów obliczeniowych, dyskowych oraz sieciowych

Czego GRID nie może …

Grid nie może :

- Naruszać bezpieczeństwa poszczególnych jednostek wchodzących w jego skład oraz naruszać ich autonomii
- Powodować konfliktów w działaniu z istniejącym już oprogramowaniem
- Narzucać użytkownikom języków programowania, narzędzi, bibliotek do programowania równoległego, itp.

Co GRID powinien …

Grid powinien :

- Umożliwiać rozproszenie geograficzne zasobów
- Obsługiwać heterogeniczność sprzętową i programową
- Być odporny na zawodny sprzęt
- Pozwalać na dynamikę dostępu do sprzętu
- Zrzeszać różne organizacje (wirtualne) z ich własnymi politykami bezpieczeństwa i dostępu do zasobów
- Być połączony poprzez heterogeniczną sieć
- Korzystać z ogólnie dostępnych, standardowych protokołów i interfejsów

Przykładowe projekty przetwarzania rozproszonego

- SETI: Projekt poszukiwania cywilizacji pozaziemskich
- Cure Cancer: Projekt, którego celem jest stworzenie leku na raka
- Fight Anthrax: Projekt, którego celem było stworzenie leku na wąglika
- Prime Numbers: Poszukiwanie liczb pierwszych
- Distributed.net
- GIMPS
- FreeDB.org
- The Internet Movie Database: baza informacji o filmach
- The Distributed Chess Project
- Wikipedia – project mający na celu zgromadzenie całej ludzkiej wiedzy
- Dmoz – Open Directory Project
- ClimatePrediction.net
- Lifemapper

SETI@HOME „Czy jest tam ktoś ?"

SETI@Home jest największym publicznym projektem związanym z przetwarzaniem rozproszonym. Jego celem jest odpowiedź na pytanie czy istnieje pozaziemska cywilizacja zdolna przesłać nam wiadomość drogą radiową.

W 1998 roku ogłoszono plany realizacji tego projektu. W ciągu następnego roku 400 000 chętnych osób rejestruje się. W 1999 roku udostępniona zostaje pierwsza wersja oprogramowania. Liczba pobrań w pierwszym tygodniu przekracza 200 tys. 26 września 2001 projekt osiągnął on 1 ZettaFLOP (10^{21} operacji zmiennoprzecinkowych), co daje średną moc 71 TeraFLOP/s. Dla porównania w tym samym czasie najszybszy komputer IBM ASC White miał moc obliczeniową 12.3 TeraFLOP/s. Do połowy 2006 roku w projekcie tym wykonano już $6.5*10^{21}$ operacji zmiennoprzecinkowych, obecnie moc obliczeniowa jest szacowana na ok. 528 TeraFLOP/s. Cywilizacji pozaziemskich poszukiwało w szczytowym momencie prawie 5,5 mln ochotników z 226 krajów, obecnie jest to około 2 mln osób.

SETI@HOME : jak to działa ?

Dane z radioteleskopu Arecibo na wyspie Puerto Rico są rejestrowane na taśmach o dużej pojemności. Dziennie na jednej taśmie typu DLT jest zarejestrowanych około 35 GB danych. Ze względu na to, że Arecibo nie dysponuje połączeniem o dużej przepustowości do Internetu, taśmy z danymi muszą być wysyłane do Berkeley tradycyjną pocztą. Na Uniwersytecie w Berkeley dane z taśm są dzielone na małe porcje o objętości 0,25MB nazywane próbkami danych. Próbki są umieszczane na serwerze SETI@Home i udostępniane do analizy uczestnikom projektu na całym świecie za pośrednictwem Internetu.

Sygnały impulsowe to emisje radiowe na pojedynczej częstotliwości, dostatecznie silne aby można było je odróżnić od szumu tła.

Wyłuskanie takich sygnałów z danych reprezentujących emisję radiową wymaga przeprowadzenia: procedury wyznaczania poziomu odniesienia (ang. baseline smoothing) dla mocy emisji radiowej w próbkach danych, eliminacji przesunięć częstotliwości (ang. de-chirping) odbieranej emisji radiowej i obliczenia szybkich transformat Fouriera (ang. Fast Fourier Transform - FFT). Oprócz pojedynczych sygnałów impulsowych SETI@Home poszukuje również ich grup układających się w określone wzorce — takie jak sygnały ciągłe, sygnały pulsujące i sygnały potrójne.

Sygnały odległego nadajnika powinny narastać i zanikać podczas przesuwania się ogniska radioteleskopu po niebie. Moc odbieranego sygnału powinna się zwiększać, a następnie maleć kreśląc tzw. krzywą dzwonową jej rozkładu w czasie (krzywa rozkładu Gaussa). Dopasowanie do krzywej Gaussa jest doskonałym testem na pozaziemskie pochodzenie sygnału i odróżnienie go od sygnałów zakłóceń interferencyjnych pochodzenia ziemskiego ponieważ ich charakterystyka rozkładu mocy daje wartość stałą, a nie zmienną w czasie. Test dopasowania do krzywej Gaussa jest wykonywany dla wszystkich rozdzielczości częstotliwości większych od 0,59Hz.

Nasi Obcy sąsiedzi mogą wysyłać sygnały niekoniecznie jako miłe dla naszego ucha czyste tony, które moglibyśmy wykryć. Jeśli chcieliby wykorzystać dostępną im energię bardziej ekonomicznie to ich sygnały mogą być seriami pulsacji przerywanymi okresami ciszy. Takich powtarzających się pulsacji i sygnałów potrójnych poszukuje się w SETI@Home dla każdej rozdzielczości częstotliwościowej od 0,59Hz.

SETI@HOME – „Work-units”

W projekcie SETI@home analizuje się emisję radiową w paśmie 2,5MHz skupionego wokół częstotliwości 1420MHz. Takie pasmo jest jednak w dalszym ciągu zbyt szerokie do prowadzenia dokładnych analiz i dlatego dzieli się je na 256 węższych pasm, każde o szerokości 10kHz (dokładnie 9766Hz, ale posługujemy się tu okrągłymi liczbami aby czytelniej przedstawić tok rozumowania). Podział jest wykonywany na drodze programowej za pomocą oprogramowania "podzielnika" (ang. *splitter*). Pasma o szerokości 10KHz są już łatwiejsze w analizie. Rejestracja sygnałów wykrywalnych w takich pasmach wymaga szybkości zapisu na poziomie 20 000 bitów na sekundę (20 kbps). 100 sekund zarejestrowanej emisji po 20 000

daje zatem 2 000 000 bitów lub jak kto woli około 0,25MB przy założeniu, że jeden bajt tworzy 8 bitów. Porcję danych o objętości 0,25MB nazywa się "próbką danych" (ang. *work unit*). W tym co jest przesyłane z serwera danych jest jednak sporo dodatkowych informacji poza zarejestrowaną emisją radiową. Powoduje to, że objętość próbek udostępnianych uczestnikom Projektu zwiększa się do około 340kB.

SETI@HOME – Mapa nieba

Za pomocą radioteleskopu w Arecibo można obserwować tylko pewien określony fragment nieba. Widoczna na slajdzie mapa nieba prezentuje ów fragment, a poszczególne kolory oznaczają liczbę przeprowadzonych nasłuchów danego wycinka nieba. Im bardziej kolor jest czerwony tym większa jest ta liczba.

Cure Cancer – Lek na raka

W ciągu ostatnich 50 lat udało się stworzyć ok. 40 leków, które mogą być wykorzystywane do walki z rakiem. Leki te przedłużają życie wielu chorym, często jednak kosztem wielu wyrzeczeń i skutków ubocznych, które są równie dokuczliwe jak sama choroba. Możliwe skutki uboczne są w wielu przypadkach na tyle groźne, że podawanie leku ograniczone musi być do minimalnych dawek. To powoduje z kolej że w ponad 50% przypadkach leki są nieskuteczne. Z drugiej strony terapie te są bardzo kosztowne. Szacuje się że ponad 6% kosztów związanych z leczeniem pochłania walka z rakiem. W skali globalnej jest to ok. 37 mld $ bezpośrednich kosztów medycznych i 11 mld $ kosztów pośrednich związanych np. z niemożliwością pracy. Jak więc widać motywacja prowadzenia badań nad nowymi lekami zwalczającymi raka jest ogromna.

Firma United Device przy współpracy z Wydziałem Chemii na Uniwersytecie Oxfordzkim i Narodową Fundacją do Walki z Rakiem uruchomiła projekt mający pomóc w wynalezieniu skutecznego leku na raka. Projekt ten w zamierzeniu miał przyspieszyć w sposób istotny pierwszą, wstępną fazę opracowania nowego leku, związaną z identyfikacją molekuł, które będą „pasowały" do białek rokujących duże nadzieje w walce z rakiem. Proces dopasowywania można porównać z poszukiwaniem właściwego klucza do zamkniętego zamka. Należy przeanalizować miliony możliwości i tylko jedna z nich okaże się właściwa.

CureCancer – katalog protein

Na tym slajdzie znajdują się przykładowe białka, które zostały wytypowane jako obiecujące elementy nowego leku. Ich wybór uwarunkowany był poprzednimi wynikami badań. Lista potencjalnych kandydatów nie jest oczywiście zamknięta i kolejne pozycje mogą być do niej dodawane w miarę postępowania badań.

Świat walczy z wąglikiem !

W świetle wydarzeń z 11 września 2001 grupa naukowców z uniwersytetu w Oxfordzie przygotowała projekt, którego celem było wsparcie badań nad skutecznym lekiem zwalczającym wąglika.

22 stycznia 2002 roku firma United Devices ogłosiła uruchomienie projektu mającego pomóc w walce z wąglikiem. Udostępniła ona wszystkim chętnym możliwość włączenia się w obliczenia rozproszone na platformie MetaProcesora, którą sama stworzyła. Na apel o pomoc odpowiedziało ok. 1,3 mln osób z całego świata. Po 24 dniach (14 lutego) projekt został zakończony z sukcesem. Celem projektu było przeanalizowanie 3,57 mld molekuł pod kątem dopasowania ich do białek zawartych w toksynie wąglika. Znalezienie pasującego elementu pozwoliłoby na zneutralizowanie tej toksyny, eliminując ją jako skuteczną broń biologiczną. W czasie trwania projektu każda z molekuł była analizowana pod kątem zgodności z odpowiednimi

białkami. Ostatecznie udało się zawęzić listę molekuł do ok. 300.000 (!), co znacznie skraca czas dalszych badań laboratoryjnych. Wysoką jakość wyniku ostatecznego zapewniono stosując 5-krotną redundancję przy sprawdzaniu każdej molekuły. W oficjalnych statystykach twórcy projektu podali że łączny czas przetwarzania jednostki centralnej wynosił w tym projekcie ponad 6989 lat (!)

Google – początki

(Uniwersytet Stanford, Backrub project, 1998):

- Sun Ultra II Dual 200MHz, 256MB of RAM, 3 x 9GB HDD i 6 x 4GB HDD
- 2 x 300 MHz Dual Pentium II, 512MB RAM, 9 x 9GB HDD
- 8 x 9GB HDD (ofiarowany prze IBM).
- 10 x 9GB HDD (własnoręcznie wykonany)

Google – dzisiaj

- Ponad 450,000 serwerów (533 MHz Intel Celeron – 1,4GHz Intel Pentium III)
- Jeden lub więcej 80GB HDD w każdym serwerze
- 2 – 4 GB RAM w każdym węźle
- 5 farm serwerów (Kalifornia, Wirginia, Oregon), dokładnie dane nie są znane.
    - ok. 6000 procesorów
    - 12000 HDD
    - Połączenie ze światem OC-48 (2488Mbit/s)
    - Połączenia pomiędzy farmami OC-12 (622Mbit/s)

Nie wykorzystuje się procesorów najnowszej generacji ze względu na pobór mocy. Celem Google jest nie maksymalizacja wydajności jako takiej lecz maksymalizacja wydajności w kontekście zużywanej energii elektrycznej. Niestety koszty związane z rachunkami za energię elektryczną są ogromne i wynoszą ok. 1-2 mln $ miesięczne !!!

Wraz z rozwojem sieci Internet wzrasta także obciążenie dla serwisu wyszukującego. Na szczęście ceny sprzętu spadają a wydajność rośnie. Dzięki temu nawet przy dwukrotnym zwiększeniu zasobów sieci WWW nie jest konieczne dwukrotne zwiększenie liczby komputerów.

Firma Google stosuje tani sprzęt komputerowy. Źródła takiego podejścia można doszukiwać się w historii firmy. Jej założyciele: Sergey Brin i Larry Page jeszcze jako studenci Uniwersytetu Stanforda, w swoim projekcie używali sprzętu, wycofanego z normalnej eksploatacji w wyniku modernizacji. Sprzęt ten nie gwarantował dużej prędkości, ale był bardzo tani lub nawet darmowy. Dzięki zastosowaniu taniego sprzętu koszty wyszukiwania i utrzymania działalności są stosunkowo niskie, a zyski generowane przez np. wyświetlanie reklam są bardzo duże. Kluczem do sukcesu jest oprogramowanie, które pozwala stosować taki tani i zawodny sprzęt. Klastry zbudowane są z nie markowych serwerów 1U i 2U umieszczonych w szafach typu rack. Każdy serwer posiada normalny procesor x86 oraz zwykły dysk IDE. Jego awaryjność też jest na poziomie zwykłego komputera PC, co oznacza że średnio po upływie 3 lat się zepsuje. O ile w przypadku komputerów domowych jedna awaria na 3 lata jest do zaakceptowania, o tyle dla Google stanowi ona poważny problem. W klastrze składającym się z tysiąca komputerów średnio jeden serwer dziennie się psuje. Dlatego oprogramowanie zostało tak napisane, żeby zawsze brać pod uwagę możliwość awarii każdego komponentu, który jest natychmiast omijany.

Google w liczbach …

- 112 międzynarodowych domen
- 26 mld indeksowanych dokumentów
- 300 mln żądań dziennie
- 10000 pracowników zatrudnionych na pełen etat
- 2,9 mld zapytań
- ~300 Teraflops - moc obliczeniowa
- 100 dok/s jest pobieranych przez roboty wyszukujące

Główne zadania realizowane przez Google

- Aktualizacja zawartości
- Indeksowanie przechowywanej zawartości
- Obsługa żądań użytkowników

Schemat architektury Google

Serwer URL - Pobiera lokalizatory URL z indeksu dokumentów i przesyła je do automatów skanujących sieć.

Roboty skanujące (ang. *crawlers*) - Pobierają dokumenty z listy i przesyłają na serwer składujący, zaimplementowane w języku python, zwykle działa ich kilka (3-4),utrzymują ok. 300 aktywnych połączeń jednocześnie, są w stanie pobrać ok. 100 dokumentów/s.

Serwer składujący (ang. *Store server*) - Kompresuje przychodzące dokumenty, przydziela im unikalne identyfikatory (docID) oraz zapisuje w repozytorium.

Indekser - Odczytuje dokumenty z repozytorium i analizuje je w poszukiwaniu słów, dla każdego słowa tworzona jest struktura tzw. “word hit”

- “word hit” - przechowuje dane o:
- lokalizacji słowa w dokumencie
- rozmiar czcionki(względny)
- wielkość liter
- plain hits (zawartość treści dokumentu)
- fancy hits (adres URL, tzw. *anchor*, metaznaczniki, tytuł)

URL Resolver - Analizuje pliki 'anchor text' zapisane przez indekser, względne lokalizatory URL przekształcane są w bezwzględne, tworzy bazę danych odnośników (pary elementów docID)

Algorytm PageRankTM - Internet oparty na demokratycznych zasadach, ranking strony pokazuje jej przydatność a nie jedynie zawartość słów kluczowych, intuicyjne uzasadnienie - model “losowego” użytkownika sieci

Sorter - Tworzy tzw. indeks odwrócony (ang. *inverted index*) porządkujący trafienia 'hit' wg wordID, końcowy leksykon Program 'Dump Lexicon' indeksu odwrócony + indeks wygenerowany przez Indexer = leksykon użytkownika

Lexicon - Zorganizowany zarówno w postaci listy jak i tablicy haszującej, zawiera ok. 20 mln słów kluczowych.

Google Web Server - Łączenie wyników z opisem na podstawie serwerów dokumentów i formatowania wyników, sugestie (moduł sprawdzania poprawności pisowni), reklamy

Google zindeksował jak dotąd ponad 4 miliardy stron WWW, które zajmują średnio ok. 10 KB. Łącznie daje to 40 Terabajtów danych, które muszą być średnio 1000 razy w ciągu sekundy przeszukiwane i na ich podstawie wygenerowane muszą zostać wyniki zwrócone w ciągu ułamka sekundy do użytkownika.

Publikowanie treści w Internecie nie jest ustandaryzowane, dlatego proces wyszukiwania też musi być bardzo zróżnicowany i „wrażliwy" na wiele dodatkowych informacji i wskazówek. Aby zindeksować daną stronę, Google analizuje wszystkie linki pomiędzy dokumentami znajdującymi się na witrynie. Każdy link jest potencjalnym źródłem dodatkowych informacji o stronie. Tekst linku pozwoli zorientować się co znajduje się na tej stronie. Jeżeli wiadomo co znajduje się na stronie na której jest link, to pozwala to też zorientować się jaka jest jakość strony do której ten link prowadzi. Takie podejście do oceny stron internetowych jest podstawą algorytmu Page-Rank, który jest podstawowym źródłem sukcesu wyszukiwarki Google. Algorytm ten Page-Rank nie bierze pod uwagę tylko liczby linków, ale też ich jakość i wagę. W rezultacie, wyświetlane są strony, które z dużym prawdopodobieństwem będą tym czego oczekuje użytkownik.

Stosowanie algorytmu Page-Rank dla każdego wyszukiwania i każdej strony jest oczywiście niemożliwe. Dlatego cały proces jest podzielony na kilka etapów i rozdzielony na różne serwery. Gdy system dostaje zapytanie od użytkownika, wędruje ono najpierw do serwerów indeksowych na których znajduje się skatalogowana zawartość całej sieci WWW. Indeks stanowi przyporządkowanie poszczególnych słów kluczowych do dokumentów, które je zawierają. Przy podawaniu wyników Google bierze pod uwagę jeszcze dodatkowe informacje takie jak miejsce występowania słowa kluczowego np. nagłówek czy stopka, pogrubienie itp. Każdy serwer indeksowy zawiera tylko pewien fragment wiedzy na temat sieci WWW. Na jednym komputerze, a tym bardziej na tanich maszynach stosowanych w Google, indeks całej sieci by się po prostu nie zmieścił. Tak więc cały indeks sieci jest rozdzielony na wiele serwerów i zapytanie jest przesyłane jednocześnie do wielu komputerów, przy czym każdy z nich przeszukuje jedynie swój zbiór danych. Po otrzymaniu zapytanie Google wylicza ok. 1000 najlepszych wyników do których przypisuje tzw. Document-ID, czyli identyfikator dokumentu. Następnie identyfikatory te wędrują do serwerów dokumentów na których znajdują się kopie przeszukiwanych przez Google stron. Dzięki temu oprócz listy adresów Google wyświetla także tytuły oraz część tekstu znajdującego się w dokumencie. Także w tym wypadku każdy serwer zawiera tylko pewien podzbiór danych. W ostatnim etapie wyszukiwania, wyniki wędrują do Ad-serwerów, czyli serwerów reklamowych, które do listy stron dodają reklamy, stanowiące podstawowe źródło przychodów firmy. Gotowa lista wyników wraz z reklamami wysyłana jest do przeglądarki internauty.

Podstawowym zabezpieczeniem przed utratą danych jest ich replikacja. Każdy serwer, który zawiera dane, posiada nawet 10 kopii. Wydaje się że jest to drogie rozwiązanie, ale przy takim obciążeniu serwery i tak muszą być zwielokrotniane, żeby obsłużyć dużą liczbę zapytań. Gdy zatem awarii ulegnie jeden serwer to wydajność usługi spada o 10%. Z punktu widzenia użytkownika nie jest to prawie zauważalne. Awaria taka przy poprawnym rozdziale obciążenia, może być skutecznie maskowana do czasu jej usunięcia. Obecnie Google posiada nie 10, ale nawet 50 kopii każdego serwera. Firma tworzy kopie serwerów, kopie zbiorów serwerów oraz kopie centrów obliczeniowych, które rozsiane są na całym świecie. Od lutego 2000 wyszukiwarka nie miała ani jednej poważnej awarii. Awaria sprzed 5 lat wydarzyła się, gdy Google miało tylko jedno centrum obliczeniowe w którym popsuł się główny switch. Przez pół godziny wyszukiwanie z wykorzystaniem serwisu Google nie było mozliwe. Teraz wszystkie dane posiadają swoje kopie rozsiane w różnych centrach obliczeniowych. Szefowie firmy twierdzą że utrata danych z jednego centrum nie stanowi problemu, gdyż jego uruchomienie od nowa trwa nie dłużej niż 3 dni. Dane składowane na serwerze zapisywane są na dyskach przy

pomocy systemu GFS (Google File System). Pojedynczy blok ma aż 64 MB wielkości. Każdy blok zapisywany jest na trzech różnych serwerach znajdujących się w różnych szafach serwerowych podpiętych do różnych przełączników. Takie rozwiązanie gwarantuje, że awaria lub błąd zapisu/odczytu danych nie będzie miał wpływu na jakość wyników. W sumie Google posiada ponad 30 klastrów z systemem GFS. Jeden klaster może składać się nawet z 2000 serwerów i magazynować Petabajty danych. Każdy klaster ma wydajność zapisu i odczytu na poziomie ok. 2 Gbit/s.

Aby uzmysłowić sobie skalę przetwarzania i ilość gromadzonych danych wystarczy powiedzieć że na tym poziomie nawet błędy zapisu wynikające ze specyfikacji dysków twardych mogą być problemem. Gwarantują one bowiem poziom błędów na poziomie $1:10^{-15}$ bitów co oznacza że jeden na 10^{15} bitów może zostać zapisany błędnie, a oprogramowanie sterujące dysku twardego tego nie wykryje. Przy danych gromadzonych w PB takie wartości mają już duże znaczenie, dlatego system plików GFS posiada dodatkowe mechanizmy weryfikacji.