

OPIS ROZWIĄZANIA

Opracowaliśmy aplikację w języku java, która rozwiązuje ten problem.

Konstrukcja sieci: Tworzymy logiczną topologię sieci przypominającą drzewo. To rozwiązanie cechuje się wysoką podatnością na nieaktywne komputery - prowadzą do rozszczepienia sieci. Nasza usprawniająca modyfikacja polega na tym, że w każdym wierzchołku drzewa nie znajduje się jeden komputer, a mesh, którego rozmiar jest niewielki w porównaniu do całej sieci. Dzięki temu rozspójnienie będzie bardzo mało prawdopodobne. Ponadto, gdy jakaś informacja ma być przesłana w górę lub dół drzewa, komputer odbierający jest wybierany losowo spośród mesha, przez co obciążenie na każdej maszynie znacząco maleje, gdyż każde zapytanie przechodzące przez wierzchołek odwiedza tylko jeden komputer.

Krótki opis wydajności - jeśli założymy, że każdy mesh ma rozmiar 50 (a możemy spokojnie je powiększyć), a komputer ma średnio 20% na posiadanie włączonej aplikacji, to szansa na to, że zostaniemy odcięci od reszty drzewa wynosi około 1:100000.

Zapytania w sieci oraz odpowiedzi: Komputer może zapytać komputery o posiadanie jakiegoś pliku. Wysyła w tym celu pakiet we wszystkich kierunkach z pewnym ograniczeniem odległości. komputery rekurencyjnie wysyłają pakiet dalej, a potem zbierają odpowiedź i stopniowo wraca ona do nadawcy. Jeśli nadawcy spodoba się któryś z plików, to może nawiązać połączenie bezpośrednio z komputerem posiadającym ten plik (dzięki danym uzyskanym z zapytania) i pobrać go sobie bez pośredników.

Podłączanie do sieci: Komputer, który zna maszynę w sieci może zapytać ją o to, czy może się do niej podłączyć. Odbiorca wysyła zapytanie do korzenia sieci i stamtąd znajduje dobrą pozycję dla nowego wierzchołka i przesyła mu odpowiednie dane do nawiązania połączenia z resztą sieci.

Lokalnie: Aplikacja ma dostęp do folderu, który jest udostępniany z resztą sieci. Użytkownik może do niego przekopiować dane, którymi chciałby się podzielić. Aplikacja przed każdym zapytaniem sprawdza, jakie pliki są dostępne i wysyła odpowiedź z najnowszymi danymi.