

Prace Naukowe Centrum Obliczeniowego
Politechniki Wrocławskiej

Seria: Konferencje

5

3

Sieci komputerowe
Teoria, technika, zastosowania

Chęć

Wrocław 1987

A 4947 I/3



PRACE NAUKOWE POLITECHNIKI WROCLAWSKIEJ

Scientific Papers of the Computer Centre
of the Technical University of Wrocław

No 5

No 5

Conferences

No 3

1987

Computer networks
Theory, technology, applications

Prace Naukowe Centrum Obliczeniowego
Politechniki Wrocławskiej

5

Seria:
Konferencje

3



Sieci komputerowe

Teoria, technika, zastosowania

Wrocław, 23 – 25 września 1987



Wydawnictwo Politechniki Wrocławskiej · Wrocław 1987

Redaktor naczelny

Jerzy CIEKOT

Komitety programowy konferencji

Jacek BAŃKOWSKI, Jerzy BATTEK, (przewodniczący),
Tadeusz BATYCKI, Daniel BEM, Eugeniusz BILSKI,
Jerzy BROMIRSKI, Marian BUDKA, Wojciech CELLARY,
Czesław DANIŁOWICZ, Zbigniew FRYŻLEWICZ, Zbigniew HUZAR,
Piotr KOCIATKIEWICZ, Witold KOMOROWSKI (redaktor naukowy),
Jerzy MARSZAŁEK, Dominik RUTKOWSKI, Wojciech ZAMOJSKI

Referaty zakwalifikował do druku Komitet programowy konferencji

© Copyright by Wydawnictwo Politechniki Wrocławskiej, Wrocław 1987

WYDAWNICTWO POLITECHNIKI WROCŁAWSKIEJ

Wybrzeże Wyspiańskiego 27, 50-370 Wrocław

ISSN 0860-1615

Nakład 300+ 70 egz. Ark. wyd. 20,75. Ark. druk. 18%. Papier offset. kl. III, 70 g, B1.

Oddano do druku w czerwcu 1987 r. Druk ukończono w lipcu 1987 r.

Zakład Graficzny Politechniki Wrocławskiej. Zam. 4834/87. Cena zł 335.-

Sieci komputerowe, zarówno rozległe jak i lokalne, są od kilkunastu lat przedmiotem narastającego zainteresowania badaczy konstruktorów i producentów sprzętu komputerowego. Wielodostępne systemy przetwarzania informacji obsługiwane przez wiele połączonych w sieć komputerów, znalazły na świecie liczne i efektywne zastosowania w nauce, technice i administracji.

W Polsce, prace nad budową sieci komputerowych zapoczątkowane w 1978 roku na Politechnice Wrocławskiej, przyniosły liczne konkretne rozwiązania sprzętowe i programowe, które umożliwiły uruchomienie w 1983 roku na sprzęcie krajowym, z zachowaniem standardów ISO OSI, pierwszej w kraju Międzyuczelnianej Sieci Komputerowej (MSK), łączącej trzy ośrodki naukowe. W 1986 roku uruchomiono w Instytucie Komputerowych Systemów Automatyki i Pomiarów i WZ ELWRO Sieć Komputerową Jednolitego Systemu (SKJS), odpowiadającą również standardom ISO OSI, a mającą charakter produktu o uniwersalnym charakterze zastosowań, powielarnego w różnych konfiguracjach sprzętowych. W kilku ośrodkach informatycznych kraju są prowadzone również prace nad lokalnymi sieciami mikrokomputerowymi, rozważanymi jako systemy autonomiczne lub współdziałające z sieciami rozległymi.

Konferencja, której materiały przedstawiamy, poświęcona jest zagadnieniom naukowym i technicznym oraz zastosowaniom w latach 1985-1986. Z uwagi na dość długi czas potrzebny na przygotowanie materiałów konferencyjnych, zostaną na konferencji przedstawione referaty uzupełniające, uwzględniające aktualne wyniki prac nad zagadnieniami sieci komputerowych.

Mamy nadzieję, że konferencja ta stanie się, podobnie jak poprzednia, formą konferencyjnej wymiany myśli naukowej w tej dziedzinie informatyki.

Komitet Programowy i Organizacyjny Konferencji



Edward BIELEMINIK^x
Józef LEWOC^{xx}
Andrzej STANISZ^x

sieć komputerowa,
transmisja danych,
MSK, KASK

PODSIEĆ TRANSMISJI DANYCH SIECI MSK/KASK

W pracy przedstawiono założenia, rozwiązania techniczne i programowe węzła podsieci transmisji danych eksperymentalnej sieci komputerowej MSK. Wskazano możliwości wykorzystania tych rozwiązań w sieci KASK. Scharakteryzowano aktualne tendencje w dziedzinie sieci transmisji danych w świecie, wskazano prawdopodobny kierunek rozwoju sieci KASK po zbudowaniu powszechnej sieci transmisji danych w Polsce.

1. WSTĘP

Prace nad budową podsieci transmisji danych (podsieci komunikacyjnej) Międzyuczelnianej Sieci Komputerowej (MSK) zostały zainicjowane w połowie 1979 roku. Przed wykonawcami postawiono wówczas następujące zadania:

- budowa i wdrożenie do eksploatacji 3-węzłowej podsieci komunikacyjnej,
- przygotowanie specjalistów projektowania i budowy sieci transmisji danych,
- opracowanie narzędzi i metod badawczo-pomiarowych sieci komputerowych.

Prace nad oprogramowaniem węzłów podsieci komunikacyjnej prowadzono równolegle w Politechnice Wrocławskiej i Politechnice Śląskiej, dokonując niezbędnych uzgodnień, aby zapewnić współpracę obu wersji węzłów. Przyjęte wówczas założenia są spójne z założeniami określonymi dla MSK[3]. Założenia te można sprowadzić do następujących zagadnień:

^xPolitechnika Wrocławska, Centrum Obliczeniowe
^{xx}Uniwersytet Wrocławski, Centrum Obliczeniowe

- wykorzystanie sprzętu krajowego lub krajów RWPG,
- oparcie się na międzynarodowych standardach i normach,
- budowa od podstaw całego niezbędnego oprogramowania,
- zapewnienie łatwej rozbudowy sieci.

Przyjęto także, że liczba linii, jaką węzeł powinien obsłużyć, nie może być mniejsza niż 6, i że w pierwszym etapie będą to linie o szybkości 2400 bit/s.

Główną funkcją węzła jest odbiór i nadawanie informacji od (do) współpracujących abonentów i innych węzłów podsieci. Realizacja tej funkcji jest zazwyczaj połączona z koniecznością spełniania wielu czynności dodatkowych, jak:

- buforowanie danych,
- wybór drogi w podsieci,
- przeprowadzenie niezbędnych akcji korekcyjnych,
- sterowanie i pomiary,
- współpraca z operatorem węzła.

2. PODSIEĆ KOMUNIKACYJNA SIECI MSK

2.1. Struktura podsieci

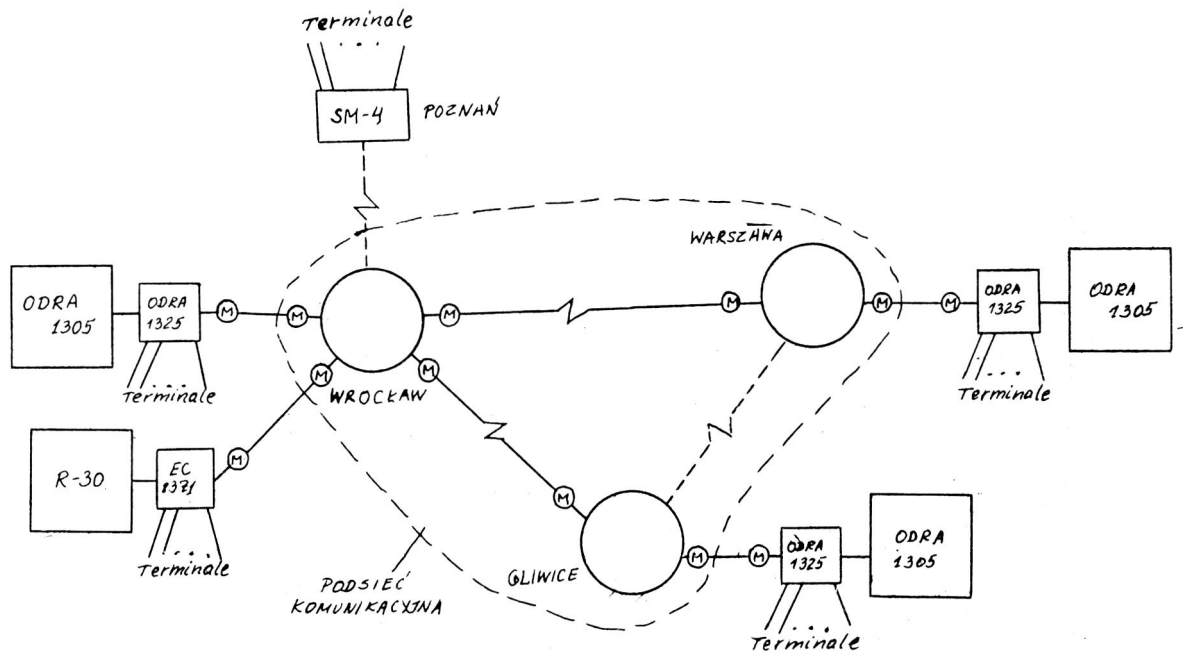
Konfigurację podsieci komunikacyjnej przedstawiono na rys. 1. Liniami przerywanymi zaznaczono połączenia nie eksploatowane na bieżąco. Koncentrator terminali znajduje się w fazie testowania. Węzły podsieci są połączone liniami telefonicznymi. Stosowana szybkość transmisji - 2400 bit/s.

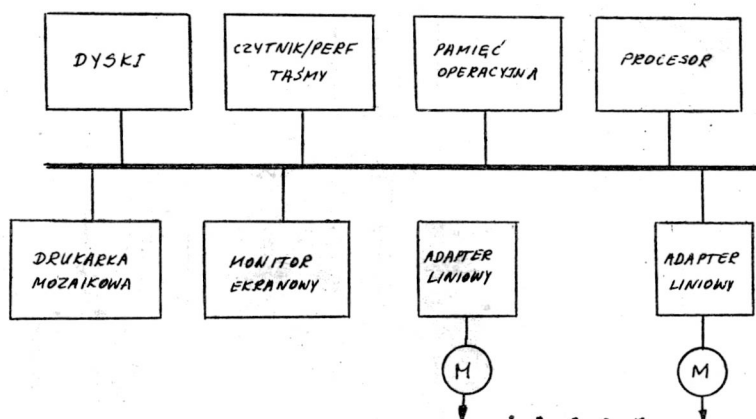
Terminale są przyłączone do sieci za pomocą procesorów czołowych (ang. front-end processor) lub poprzez koncentrator. Komputery obliczeniowe ODRA 1305 i R-32 są włączone do sieci także poprzez procesory czołowe. Mogą one być równocześnie dostępne z lokalnych terminali.

Sprzętową konfigurację węzła tworzy minikomputer SM-3 z następującymi modułami (rys.2):

- procesor GM 1301,
- pamięć operacyjna 28 k słów,
- pamięć dyskowa (2.5 Mbyte),
- monitor ekranowy,
- czytnik taśmy dziurkowanej (perforator taśmy-opcjonalnie),
- drukarka mozaikowa (opcjonalnie),
- adaptery liniowe ALS-11.

rys. 1. Konfiguracja podsięci komunikacyjnej ISK
Fig. 1. Configuration of the ISK communication subnetwork





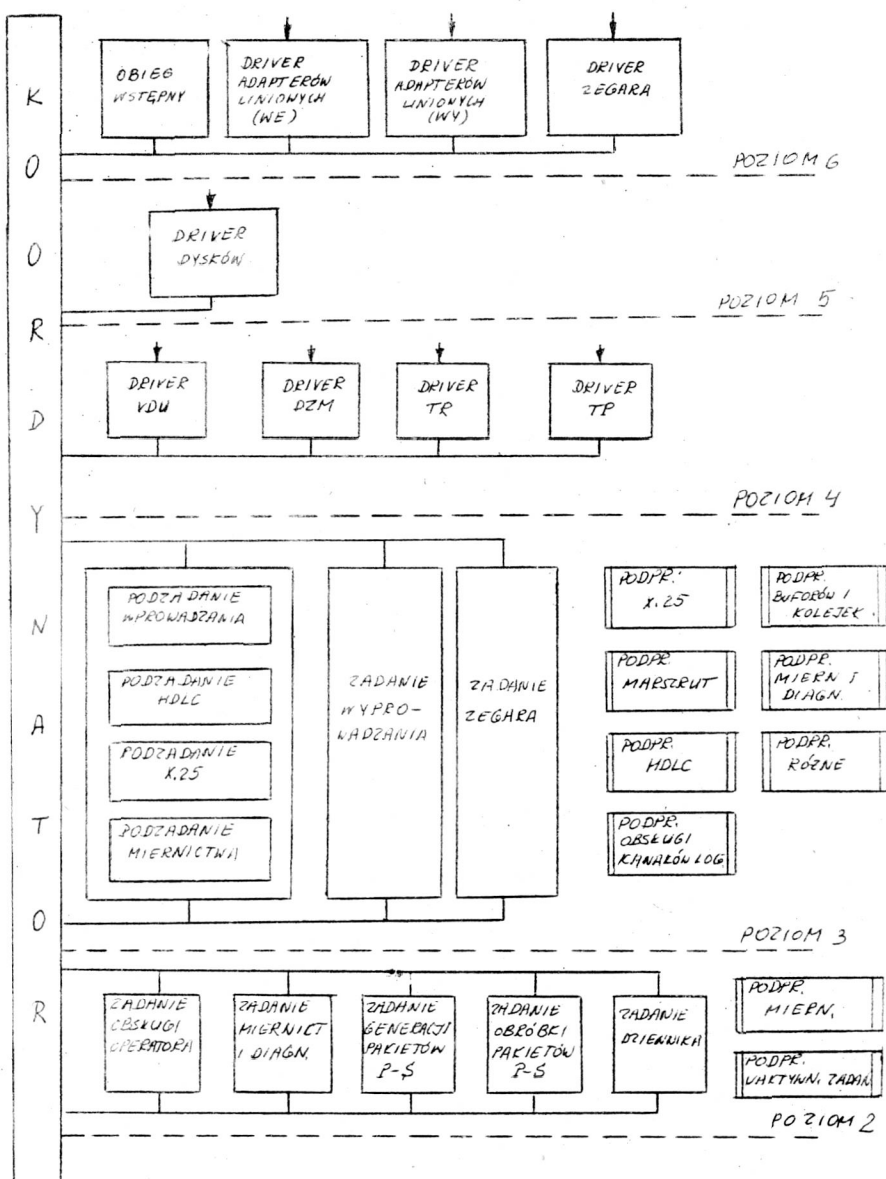
Rys.2. Struktura sprzętowa węzła
Fig.2. Node hardware structure

Adaptory liniowe ALS-11 są kompatybilne z adapterami DUP-11 firmy DEC. Zostały one wykonane w Politechnice Wrocławskiej. Istotnym ograniczeniem tych adapterów jest brak bezpośredniego dostępu do pamięci (DMA), co wymaga obsługi przerw na poziomie znakowym.

Dyski są używane do celów pomocniczych - organizacji restartu węzła oraz zbierania przez system pomiarowo-diagnostyczny podsieci-danych dotyczących pracy węzła i podsieci.

Oprogramowanie węzła podsieci komunikacyjnej [1,2,6] dla minikomputera SM-3 zbudowano jako samodzielny program, rezydujący całkowicie w pamięci operacyjnej. Można w nim wyróżnić (rys.3) część tworzącą prosty system operacyjny (koordynator, programy sterujące urządzeniami, obsługa operatora) oraz część użytkową (odbiór, przetwarzanie i nadawanie danych, generowanie i przetwarzanie informacji pomiarowo-sterujących itp.).

Funkcje węzła są realizowane w modułach pracujących na różnych poziomach priorytetu, zgodnie z architekturą procesorów rodziny SM, przy czym działania uwarunkowane czasowo (wymagające krótkich czasów reakcji) zlokalizowano na wyższych poziomach priorytetu (np. obsługa przerw adapterów liniowych i urządzeń zewnętrznych), natomiast pozostałe - o słabszych uwarunkowaniach czasowych - na niższych. Poziomy priorytetu 6,5 i 4 przeznaczone dla obsługi przerw zewnętrznych, poziomy 3 i 2 - do przetwarzania o różnym stopniu pilności.



Rys.3. Structura oprogramowania węzła
Fig.3. Node software structure

W strukturze programu wyodrębniono zadania kilku typów, będące zamkniętymi częściami spełniającymi określoną funkcję. Zadania są zwykle uruchamiane przez koordynator, niekiedy - przerwaniem zewnętrznym, a po zakończeniu pracy oddają sterowanie do koordynatora. Zadania współpracują ze sobą przez koordynator, mogą się wzajemnie uaktywniać, przekazywać dane oraz korzystać z zestawu podprogramów.

Program węzła zajmuje około 22 k słów pamięci operacyjnej, pozostawiając około 6 k słów na struktury danych (opisy linii, opisy połączeń logicznych itp.) oraz bufor.

Biorąc pod uwagę potrzebę rozszerzenia funkcji wybranego węzła o zadania pomiarowo-sterujące podsieci, opracowano wersję nakładkową systemu operacyjnego węzła. W realizacji tej zadania pomocnicze wykonywane na poziomie 2 rezydują na dysku, co pozwala na znaczną rozbudowę oprogramowania węzła, szczególnie o funkcje pomiarowo-sterujące. Wersja taka jest jednak bardziej zawodna i jak dotychczas była wykorzystywana jedynie eksperymentalnie. Docelowo dla węzłów pomiarowo-sterujących oraz o dużym ruchu przewidywano zastosowanie minikomputera SM-4.

3.2. Usługi

Protokół zewnętrzny i wewnętrzny, zastosowany w podsieci komunikacyjnej jest oparty na zaleceniach CCITT [4,5] -X.21 bis na poziomie fizycznym i X.25 na poziomach liniowym i pakietowym.

Jako protokół międzywęzłowy (wewnętrzny) wykorzystano zmodyfikowaną wersję protokołu X.25. W nagłówku ramki przesyłany jest dodatkowo adres DTE (ang. Terminal Equipment) nadawcy - 8 bajtów. Ułatwia to działanie odpowiednich procedur w węźle, ale zabiera 8 znaków z pasma przepustowości linii.

Dodatkowe elementy w ruchu międzywęzłowym stanowią pakiety pomiarowo-diagnostyczne. Są one identyfikowane w nagłówku pakietu za pomocą pola GFI (ang. General Format Identifier), równego 1111 i rozróżniane między sobą w oparciu o rozszerzenie pola GFI. Pakiety pomiarowo-diagnostyczne są przesyłane na zasadzie datagramów. Celem uniknięcia zapętlenia, dopuszczalna liczba przejść międzywęzłowych jest ograniczona.

Usługi podstawowe podsieci na styku do DTE wynikają z protokołu X.25. W aktualnej wersji podsieci wdrożono następujące opcje:

- połączenia wirtualne (stałe i czasowe),
- potwierdzenia typu końcowego (ang. end-to-end),
- standardowa długość pola danych pakietu - 128 oktetów,
- numeracja ramek i pakietów modulo 8,

negocjacja rozmiaru okna.

Dla ułatwienia obsługi podsieci opracowano wewnętrzny system pomiarowo-diagnostyczny pozwalający na rejestrację ważniejszych zdarzeń, a także wprowadzono zestaw pakietów pomiarowo-diagnostycznych jako usługi dodatkowe.

System pomiarowo-sterujący (SPS) ma strukturę modułową, która pozwala na wygenerowanie pewnego podzbioru modułów tego systemu. Funkcje spełniane przez SPS można podzielić na lokalne i rozproszone.

System lokalny zamyka się całkowicie w ramach węzła. Głównym jego zadaniem jest zbieranie danych o pracy węzła, składanie ich na dysku oraz monitorowanie. Przetwarzanie pamiętanych na dysku danych pomiarowych wykonywane jest w trybie pośrednim. W systemie lokalnym nie ma mechanizmów wymiany informacji między węzłami, natomiast przetwarzanie danych pomiarowych daje statystyczny obraz pracy węzła. Zebrane i zapamiętane w systemie lokalnym dane tworzą bazę informacyjną, wykorzystywaną dalej w systemie rozproszonym. Baza ta powstaje w oparciu o dwa mechanizmy: rejestrację zdarzeń i rejestrację stanów.

Rejestracja zdarzeń polega na odnotowaniu faktu wystąpienia określonego zdarzenia w pracy węzła. Przykładami zdarzeń istotnych dla pracy węzła są: nawiązanie połączenia, wysłanie określonego typu pakietu, przyjęcie ranki, wyłączenie linii, restart węzła itp. Dobór zdarzeń do rejestracji uwarunkowany jest potrzebami wynikającymi zarówno z eksploatacji sieci MSK, jak i z prac badawczych prowadzonych nad podsiecią.

Najprostszym sposobem rejestracji jest zwiększenie o jedną licznika, przyporządkowanego danemu zdarzeniu, w momencie wystąpienia zdarzenia. W pewnych wypadkach, oprócz zliczania, rejestruje się dodatkowe atrybuty zdarzeń, co przy natychmiastowym przetwarzaniu daje opis zdarzenia w postaci danych statystycznych, takich jak wartość średnia czy maksymalna parametru. Liczbę przetwarzanych parametrów oraz liczbę danych statystycznych ogranicza czas przetwarzania, który musi tu być bardzo krótki.

Rejestracja stanów polega na okresowym notowaniu chwilowej wartości określonych parametrów, charakteryzujących stan węzła. Wartości te przechowywane są w pamięci do chwili następnego pomiaru. Niektóre parametry podlegają bieżącemu przetwarzaniu, dzięki czemu w każdej chwili dostępna jest ich wartość średnia i maksymalna. Moment rejestracji wyznacza zegar, lub zadanie operatora. Częstotliwość rejestracji zależy od szybkości zmian wartości parametru.

System rozproszony pozwala na wymianę informacji pomiarowo-sterującej pomiędzy węzłami. Mechanizmem wymiany są pakiety pomiarowo-sterujące. W każdy węzeł wbudowane są moduły umożliwiające tworzenie nadawanie i odbiór pakietów. Pakiety inicjowane są przez operatora dowolnego węzła, żądającego informacji z podsieci. W tym celu zdefiniowany został prosty język poleceń, pozwalający na tworzenie i wysyłanie w podsieć pakietów. Odpowiedzi na wysyłane pakiety monitorowane są w węźle generującym pakiet.

Struktura pakietów pomiarowo-sterujących została zaprojektowana zgodnie z pakietami protokołu X.25 [1], obowiązującego w sieci MSK. Różnią się one kodem GFI, który ma postać 1111 dla pakietów systemu SPS. Nagłówek pakietu, poza kodem GFI oraz typem pakietu zajmującego drugą połowę oktetu, zawiera adres odbiorcy, adres nadawcy, czas wysyłania pakietu oraz znacznik odbicia. Znacznik ten przyjmuje wartość 1 w pakietach nadawanych (pierwotnych), a wartość 0 w pakietach odpowiedzi. Poniżej nagłówka jest pole danych, którego rozmiar i znaczenie zależy od typu pakietu. Długość pakietu pomiarowego nie może być większa niż maksymalna długość standardowego pakietu w MSK.

W obecnie eksploatowanej wersji węzła wdrożono następujące pakiety pomiarowo-sterujące: echo, śledzenia drogi, drogi wymuszonej i sonda.

Bazując na zdobytych doświadczeniach opracowano nową wersję SPS. Zasadniczo nowa jakość tego rozwiązania polega na wyróżnieniu w podsieci węzła pełniącego funkcję centrum pomiarowego (CP), obok normalnych funkcji węzła. Węzeł CP ma możliwość realizacji szeregu dodatkowych funkcji w relacji "nadrzędny-podrzędny". Są to m.in. synchronizacja czasu w podsieci, zbieranie informacji o stanie podsieci, zdalna ingerencja w pracę poszczególnych węzłów.

Podstawowymi elementami SPS są pakiety pomiarowo-sterujące oraz dziennik systemu. Pakiety służą do komunikowania się CP z podsiecią, jak również poszczególnych węzłów pomiędzy sobą. W dzienniku gromadzone są informacje o pracy węzłów i podsieci. Jest on prowadzony przez CP. Poszczególne węzły mogą prowadzić tzw. dzienniki lokalne, do których zapisuje się informacje o pracy danego węzła.

W stosunku do wcześniejszej wersji SPS struktura pakietów pomiarowo-sterujących uległa jedynie nieznacznej modyfikacji. Poszczególne pakiety rozróżniane są przez kod grupy i kod rozszerzenia. Pozwala to na włączanie/wyłączanie całych grup pakietów. Wprowadzono następujące grupy pakietów:

- pakiet synchronizacji czasu,
- pakiety diagnostyczne podsieci,
- pakiety pobierania danych,

- pakiety modyfikacji kodu,
- pakiety sztucznego ruchu,
- pakiety dialogu operatorskiego,
- pakiety autonomiczne.

Źródłem, inicjującym generowanie pakietów mogą być: operator, zegar lub określone zdarzenie w węźle. Niektóre pakiety (np. modyfikacji kodu) są zastrzeżone dla CP.

Wyżej opisana wersja SPS została sprawdzona jedynie w warunkach laboratoryjnych. Do jej wdrożenia nie doszło z powodu ograniczeń pamięci minikomputera SM-3, a także MERA-60.

2.3. Obsługa operatorska

Obsługa operatorska węzła powinna być ograniczona do minimum. Jednak na etapie wstępnej eksploatacji sieci jest ona niezbędna. Do podstawowych czynności operatora należy załadunek węzła, obsługa stanów awaryjnych i zakończenie pracy. Do dyspozycji operatora przewidziano kilkanaście poleceń ułatwiających współpracę z węzłem. Umożliwiają one wykonanie takich czynności, jak przydział urządzeń zewnętrznych, zmiany daty i czasu, monitorowanie stanu węzła, wysyłanie pakietów pomiarowo-sterujących itp.

2.4. Parametry techniczne węzła

Do podstawowych parametrów ilościowych węzła należy przepustowość i opóźnienie tranzytowe. Z dokonanej analizy oraz pomiarów wynika, że czas przetwarzania ramki w węźle jest rzędu kilku milisekund i jest minimalny w porównaniu z czasem transmisji (zwłaszcza przy szybkości transmisji 2400-9600 bit/s). Opóźnienie w węźle może odgrywać istotną rolę w pobliżu jego nasycenia (tj. w warunkach tworzenia się kolejki do nadajnika linii).

Przepustowość węzła jest wyrażana w bit/s lub w ramkach na sekundę. Z wartości przepustowości wynika liczba linii, jaką węzeł może obsłużyć. Obliczono szacunkowo przepustowość węzła dla minikomputerów SM-3 i SM-4, pracujących z adapterami liniowymi ALS-11 i ALS-11M. (Adapter ALS-11M ma mechanizm DMA, natomiast ALS-11 wymaga obsługi przerw znakowych). Obliczeń dokonano wychodząc z warunku niegubienia przerw w najbardziej niekorzystnych okolicznościach (przesyłanie ciągu krótkich ramek). Z obliczeń wynika, że węzeł może obsłużyć do 8 (dla SM-3/ALS-11) i do 20 (dla SM-4/ALS-11M) linii przy szybkości 2400 bit/s.

2.5. Uzyskane doświadczenia

Pamięć komunikacyjna w konfiguracji z rys. 1 zaczęła pracę w końcu 1983 roku. Wcześniej przeprowadzono szereg doświadczeń nad pracą w warunkach laboratoryjnych węzłów zainstalowanych w jednym pomieszczeniu. Opracowano specjalne narzędziapomiarowo-diagnostyczne dla celów badawczych. Zasadniczo można wyróżnić dwa rodzaje prowadzonych badań. Pierwszy kierunek to badania jakościowe, dotyczące poprawności zastosowanych rozwiązań, drugi - badania ilościowe wybranych charakterystyk węzła i podsieci.

Badania jakościowe prowadzone w dwóch wariantach:

- za pomocą specjalnie wykonanego Modułu Intensywnego Testowania (MIT),
- w warunkach eksploatacyjnych.

W pierwszym wypadku testowany był, w warunkach intensywnego ruchu, węzeł uzupełniony o moduły MIT (uproszczone DTE), w drugim - podsieć w warunkach zbliżonych do rzeczywistej eksploatacji.

Także badania ilościowe węzła i podsieci prowadzone były w dwóch wariantach:

- za pomocą Systemu Intensywnego Testowania (SITWA),
- za pomocą Modułu Operacyjnego-Pomiarowego (MOP).

System SITWA opracowano, wychodząc z symulacyjnej metody testowania węzła. Z kolei MOP został wykonany w oparciu o MIT. Istotną różnicą między tymi systemami polega na tym, że SITWA stanowi uzupełnienie oprogramowania węzła o moduły symulujące ruch i moduły pomiarowe, natomiast MOP stanowi rodzaj DTE zlokalizowanego poza węzłem.

Przeprowadzone badania i bieżąca eksploatacja podsieci potwierdziły słuszność przyjętych założeń i poprawność zastosowanych rozwiązań. Pozwoliły one na określenie właściwych procedur techniczno-organizacyjnych eksploatacji i opracowanie dokumentacji eksploatacyjnej. Sformułowano także szereg wniosków dotyczących weryfikacji niektórych procedur, szczególnie tych związanych z obsługą, diagnostyką i sterowaniem pracą węzłów i podsieci.

3. PODSIEĆ KOMUNIKACYJNA SIECI KASK

3.1. Sieć KASK

Sieć KASK ma być bezpośrednią kontynuacją i rozwinięciem sieci MSK. Przewiduje się rozwój ilościowy i jakościowy sieci w następujących kierunkach:

- zwiększenie liczby komputerów obliczeniowych (hostów), przez dołączenie nowych ośrodków dysponujących komputerami ODRA 1305, R-32, R-34, R-60;
- dołączanie mini- i mikrokomputerów (SM-4, IBM PC i kompatybilne) jako urządzeń DDE, współpracujących z siecią wg. Zalecenia X.25 (terminali pakietowych- mini- i mikrohostów);
- łączenie z sieciami lokalnymi, które będą pojawiać się w środowisku akademickim, poprzez urządzenie typu "gate-way";
- rozwój podsieci transmisji danych, przez rozbudowę istniejących węzłów i instalacje nowych w dalszych ośrodkach;
- rozszerzenie zasobów sieciowych (głównie baz danych);
- pełne wdrożenie planowanych usług sieciowych (w tym transferu zbiorów) i wprowadzenie nowych.

Realizacja tych planów wiąże się z zakupem dodatkowego sprzętu (pamięci dyskowe do istniejących komputerów obliczeniowych, dodatkowe urządzenia dla dołączania nowych komputerów - komputer ODRA 1325 i adapter UAL-C1 dla dołączenia komputera ODRA 1305, procesor transmisji danych EC 8571 dla dołączenia komputera jednolitego systemu, nowe węzły, modemy, terminale itp.), co wymaga sporych nakładów finansowych. Dodatkowe środki finansowe muszą być przewidziane na koszty eksploatacyjne.

3.2. Podsieć komunikacyjna

Rozwój podsieci transmisji danych sprowadza się do instalacji nowych węzłów, opartych na rozwiązaniach sprawdzonych i eksploatowanych dotychczas w sieci MSK.

Węzeł na minikomputerze SM-3 ma stosunkowo rozbudowane oprogramowanie (zwłaszcza w zakresie diagnostyki i sterowania siecią), bazuje jednak na adapterach liniowych, wykonanych w ograniczonej ilości i przestarzałej obecnie konstrukcji. Nie udało się doprowadzić do uruchamiania produkcji tych adapterów w większej ilości. Również krajowy producent minikomputerów SM (ERA Warszawa) nie wdrożył do produkcji tego urządzenia (mimo opracowania prototypu SM-DUP), prawdopodobnie ze względu na przewidywany mały zbył). Tak więc mimo produkcji w kraju minikomputera SM-1300, będącego unowocześnioną wersją SM-3, nie ma szans na instalowanie nowych węzłów na tym minikomputerze.

W ramach sieci MSK prowadzono również prace nad przeniesieniem oprogramowania węzła na minikomputer SM-4, licząc na zwiększenie mocy węzła dzięki większej pojemności pamięci operacyjnej (oprogramowanie węzła na SM-3 w maksymalnym wariancie z trudem mieści się w dostępnej na nim pamięci i ogranicza parametry węzła, tj. liczbę linii, pojem-

ność pamięci buforowej, liczbę możliwych kanałów logicznych itp.), większej szybkości procesora i zainstalowaniu nowych adapterów ALS11-M, pracujących na zasadzie DMA. Jedną z wersji oprogramowania węzła (bez rozbudowanych funkcji diagnostyki i sterowania siecią) udało się eksperymentalnie uruchomić na minikomputerze SM-4 z adapterami starego typu, ale wersja ta nigdy nie była eksploatowana. Ograniczenia finansowe i trudności wykonawcze spowodowały, że wykonano ze sporym opóźnieniem jedynie 3 egzemplarze adapterów nowego typu. W rezultacie prace te zakończyły się niepowodzeniem i wariant węzła na SM-4 również nie może być brany pod uwagę przy budowie sieci KASK.

Węzeł na minikomputerze MERA-60 (CM 1620), jakkolwiek uboższy pod względem oprogramowania (nie zaimplementowano na nim pełnego zakresu funkcji diagnostyki i sterowania), opracowany był w ścisłych kontaktach z producentem minikomputera, z myślą o seryjnej produkcji. Dotyczy to zwłaszcza adapterów liniowych, będących jedyną niestandardową częścią konfiguracji węzła. Dzięki temu węzeł na minikomputerze MERA-60 najbardziej nadaje się do powielania w sieci KASK, jakkolwiek oznaczałoby to rezygnację z funkcji diagnostyczno-sterujących (realizowanych w tej wersji węzła w ograniczonym zakresie), istotnych dla utrzymywania sprawności podsieci, zwłaszcza przy większej liczbie węzłów.

Istnieje jeszcze możliwość rozbudowy sieci przez wykorzystanie rozwiązania, opracowanego w ramach prac nad siecią SKJS/2: procesor transmisji danych EC 8371, związany z komputerem obliczeniowym R-32, może spełniać także funkcje węzła. Jednakże połączenie go z istniejącymi węzłami wymaga dodatkowych prac programowych, niezbędnych dla dostosowania protokołu wewnątrzsieciowego (międzywęzłowego).

Istotnym oprogramowaniem rozbudowy sieci transmisji danych w sieci KASK jest możliwość uzyskania nowych łącz stałych (dzierżawionych) pomiędzy ośrodkami akademickimi. Resort łączności, dysponując ograniczoną liczbą takich łącz, nie jest zainteresowany udostępnianiem nowych łącz klientom. Taka polityka może w poważnym stopniu zablokować rozwój ilościowy sieci, a także zmienić jego kierunek, wymuszając rozbudowę w obrębie poszczególnych ośrodków akademickich (gdzie łatwiej o łącza lokalne), równocześnie osłabiając więzi pomiędzy różnymi miastami.

4. TENDENCJE ROZWOJOWE

Śledząc historię sieci komputerowych w świecie i obecne tendencje, można z dużą dozą prawdopodobieństwa stwierdzić, że eksperymentalne sieci komputerowe typu MSK/KASK z czasem muszą ustąpić miejsca sieciom, bazującym na powszechnej sieci transmisji danych z komutacją pakietów, tworzonej i administrowanej przez resort łączności. Za takim rozwiązaniem przemawia szereg czynników m.in. stały wzrost zapotrzebowania na usługi transmisji danych, wywołany postępami elektroniki, gwałtownym rozwojem techniki komputerowej, upowszechnieniem komputerów w różnych dziedzinach gospodarki narodowej itp.

Zadaniem powszechnej sieci transmisji danych jest zaoferowanie łatwo dostępnych środków dla tworzenia połączeń i transmisji danych pomiędzy różnymi urządzeniami, rozmieszczonymi w różnych punktach kraju. Rozbudowane sieci tego typu oferują możliwość podłączenia m.in. komputerów oraz terminali asynchronicznych i synchronicznych, pracujących zgodnie z przyjętymi na świecie standardami (CCITT X.25, X.3, X.26, X.29). Urządzenia te mogą być podłączane przez łącza trwałe, dzierżawione, możliwy jest także dostęp poprzez sieć telefoniczną i teleksową. Oferuje się różne szybkości transmisji i rodzaje styków abonenckich. Sama sieć transmisji danych nie świadczy na ogół żadnych usług poza transmisją danych - funkcje te spełniają urządzenia końcowe dołączone do sieci.

Współczesne sieci transmisji danych buduje się, korzystając z najnowszej technologii, jako urządzenia wielomikroprocesorowe. Rozwiązania takie cechuje:

- daleko posunięta modularność, co ułatwia budowę węzłów (central) o różnej wielkości, oraz obsługę i naprawy a także modernizację;
- bardzo wysoka niezawodność pracy, uzyskiwana dzięki stosowaniu specjalnych rozwiązań (np. dublowaniu sprzętu automatycznej diagnostyce, zdalnej sygnalizacji awarii);
- wysokie parametry techniczne (np. liczba linii abonenckich w węźle rzędu 500 i więcej, przepustowość węzła rzędu 1000 pakietów/s, liczba połączeń logicznych około 5000 itp.);
- najnowsze tendencje światowe zmierzają do integracji usług transmisyjnych poprzez cyfrową transmisję różnego typu informacji (danych, głosu, obrazów).

W pierwszym etapie buduje się sieci hybrydowe, służące do przesyłania mowy i danych. W tym zakresie osiągnięto już znaczny postęp i wiele firm oferuje gotowe produkty, w tym zintegrowane terminale

i tzw. MBX (Private Branch Exchange). Powszechna stała się możliwość dostępu do sieci komputerowych z sieci telefonicznych i telexowych.

Sieci hybrydowe są etapem przejściowym do budowy w pełni zintegrowanych sieci, tzw. Integrated Services Digital Networks (ISDN). Sieci takie wejdą do eksploatacji z początkiem lat 90-tych. Zapewnią one szeroki wachlarz usług w zakresie przesyłania danych, mowy, tekstu, grafiki, obrazów nieruchomych i ruchomych. Użytkownik będzie posiadał dostęp do tych usług poprzez ograniczony zestaw standardowych, wielofunkcyjnych interfejsów.

W parze z rozwojem usług sieciowych postępuje rozwój w zakresie linii transmisyjnych, modemów, węzłów i terminali. Coraz powszechniejsze zastosowanie znajdują łącza satelitarne i światłowodowe. Na łączach światłowodowych osiągana jest szybkość 400 Mbit/s. Cechy szczególne tych łącz to niski koszt jednostkowy, duża pojemność, niewrażliwość na zakłócenia zewnętrzne.

Opracowuje się nowe metody modulacji. Znaczne postępy osiągnięto w zakresie kodowania i upakowywania (kompresji) danych. Powoduje to stałą obniżkę kosztów jednostkowych łącz, np. dla linii telefonicznych, przenoszących cyfrowy sygnał mowy o 15% rocznie.

W zakresie węzłów obserwuje się budowę wielomikroprocesorowych central przełączających o przepustowości do 2000 pakietów/s. Dzięki daleko idącej modularyzacji istnieje możliwość tworzenia różnych konfiguracji węzłów, od najprostszych do bardzo złożonych, rozproszonych terytorialnie, np. firma Telematics Int.Ltd. oferuje rodzinę produktów do budowy sieci w postaci programowalnych węzłów wieloprocessorowych, o przepustowości 100-800 pakietów/s, z protokołami X.25, PAD, a także własnymi, z centralnym sterowaniem i zarządzaniem pracą sieci.

Godna odnotowania jest duża aktywność w zakresie urządzeń pomocniczych do budowy, uruchamiania i testowania pracy sieci. Bardzo popularne są testery i analizatory przystosowane do badania całej gamy protokołów.

5. UWAGI KOŃCOWE

Rozwiązania techniczne sieci MSK/KASK trudno nazwać nowoczesnymi; pod względem sprzętowym można je porównać do eksperymentalnych sieci zachodnich z pierwszej połowy lat siedemdziesiątych. Parametry techniczne sieci (np. liczba węzłów, liczba linii, szybkości transmisji, liczba możliwych połączeń logicznych itp.) są bardzo niskie, jeśli porównać je z istniejącymi obecnie sieciami. Jednak zaimplementowane protokoły są nowsze i opierają się na dokumentach ISO i CCITT z początków lat osiemdziesiątych. Stwarza to szansę przeniesienia części rozwią-

zacji (oprogramowania funkcjonującego w komputerach obliczeniowych), na nową bazę techniczną, tj. na przyszłą powszechną sieć transmisji danych, która zapewne powstanie w Polsce. Jest to przedsięwzięcie konieczne. Z dostępnych informacji wynika, że resort łączności prowadzi od pewnego czasu prace w tym kierunku. Ze względu na niezbędny zakres prac nie należy jednak oczekiwać udostępnienia usług tej sieci przed rokiem 1990. Do tej pory sieć KASK musi bazować na własnej sieci transmisji danych; w przyszłości możliwe będzie połączenie wszystkich zasobów sieci KASK do powszechnej sieci transmisji danych, co stwarza perspektywę jej rozbudowy.

6. LITERATURA

- [1] Bieleninik E., Hudyma E., Kosmulska-Bochenek E., Lewoc J., Stanisław A., Koncepcja systemu pomiarowo-sterującego podsieci komunikacyjnej Międzyuczelnianej Sieci Komputerowej, Raport CO PWr., Seria SPR 3/83, Wrocław 1983.
- [2] Bieleninik E., Hudyma E., Kosmulska-Bochenek E., Lewoc J., Stanisław A., Podsieć komunikacyjna Międzyuczelnianej Sieci Komputerowej, Informatyka 11-12/85.
- [3] Bilski B., Projekt techniczno-funkcjonalny Międzyuczelnianej Sieci Komputerowej - cz.I, ICT i CO PWr, Raport 15/82, Wrocław 1982.
- [4] CCITT Recommendation X.25, Yellow Book, Vol.VIII, Fasc.VIII.2, November 1980.
- [5] Propozycja norm dla podsieci komunikacyjnej Międzyuczelnianej Sieci Komputerowej, Raport CO PWr., Seria SPR, 6/84, Wrocław 1984.
- [6] Zagadnienia oprogramowania węzła podsieci komunikacyjnej sieci komputerowej MSK, Prace Naukowe CO PWr., Seria Studia i Materiały 2/1983, Wrocław 1983.

COMMUNICATION SUBNETWORK OF THE MSK/KASK NETWORKS

In the paper the hardware and software solutions of the data transmission node of the Interuniversity Computer Network MSK are presented. The possibility of utilization those solutions in the KASK network is discussed. Actual world trends in data communication and tendencies in KASK developments have been described.

СЕТЬ ПЕРЕДАЧИ ДАННЫХ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ МСК/КАСК

В статье рассматриваются положения и технические и программные решения узла сети передачи данных экспериментальной вычислительной сети МСК. Указана возможность использования этих решений в сети КАСК. Представлены актуальные тенденции развития сетей передачи данных и указаны возможные направления развития сети КАСК с учетом внедрения в Польше почтовой сети передачи данных.

Eugeniusz BILSKI*

Sieć transmisji danych,
standardy

USŁUGI I STANDARDY SIECI TRANSMISJI DANYCH

Istotną częścią składową sieci komputerowych są sieci transmisji danych. Użytkownik ma do dyspozycji usługi i udogodnienia zależne od rodzaju sieci transmisji danych. Ponadto urządzenia końcowe użytkownika muszą spełniać standardy, które również są zależne od rodzaju sieci. W referacie przedstawiono charakterystyki użytkowe oraz przegląd standardów stosowanych w różnych rodzajach sieci transmisji danych.

1. WPROWADZENIE

Równolegle z burzliwym rozwojem sprzętu informatyki i jego oprogramowania, obserwuje się w ostatnich latach również intensywny rozwój sieci komputerowych. Stanowiska pracy wyposażone w mikrokomputery osobiste, muszą funkcjonować w oparciu o jednolite bazy informacyjne oraz powinny posiadać możliwość komunikacji z innymi stanowiskami pracy. Jednolitość baz jak i problemy komunikacji należy widzieć zarówno w skali przedsiębiorstwa lub instytucji, jak i w skali regionu, resortu, a docelowo również w skali międzynarodowej. Stąd wynika naturalna potrzeba budowy sieci komputerowych lokalnych, regionalnych, krajowych, a także potrzeba połączeń międzynarodowych. Lokalne sieci komputerowe są wprawdzie sprawą wewnętrzną przedsiębiorstwa lub instytucji, ale potrzeba ich łączenia z sieciami regionalnymi, wymaga uwzględnienia w fazie ich projektowania sposobów takiego łączenia - wykracza więc poza lokalne rany; problemy te są przedmiotem osobnych referatów i nie będą dalej omawiane. Natomiast sprawą dotyczącą wszystkich użytkowników sieci komputerowych, są sieci transmisji danych w rozległych sieciach komputerowych (regionalnych i krajowych). Użytkowników interesują praktycznie tylko usługi jakie może realizować sieć transmisji danych; nie interesują ich na ogół szczegóły budowy takiej sieci. Dlatego w referacie ograniczono rozważania do charakterystyk użytkowych różnych rodzajów sieci transmisji danych oraz wymagań jakie muszą spełniać

* Instytut Cybernetyki Technicznej Politechniki Wrocławskiej

urządzenia końcowe użytkowników, aby mogły one być włączone do tych sieci.

W referacie podane zostały charakterystyki dzierżawionych linii telefonicznych i zbudowanych w oparciu o nie, sieci transmisji danych z komutacją pakietów (PSFDN), sieci transmisji danych z komutacją kanałów (CSFDN), komutowanych sieci telefonicznych (PSTN) oraz sieci zintegrowanych (ISDN). Dla każdego rodzaju sieci podany został komplet standardów jakie muszą spełniać urządzenia końcowe użytkowników aby mogły być do tej sieci włączone.

2. CHARAKTERYSTYKI UŻYTKOWE SIECI TRANSMISJI DANYCH

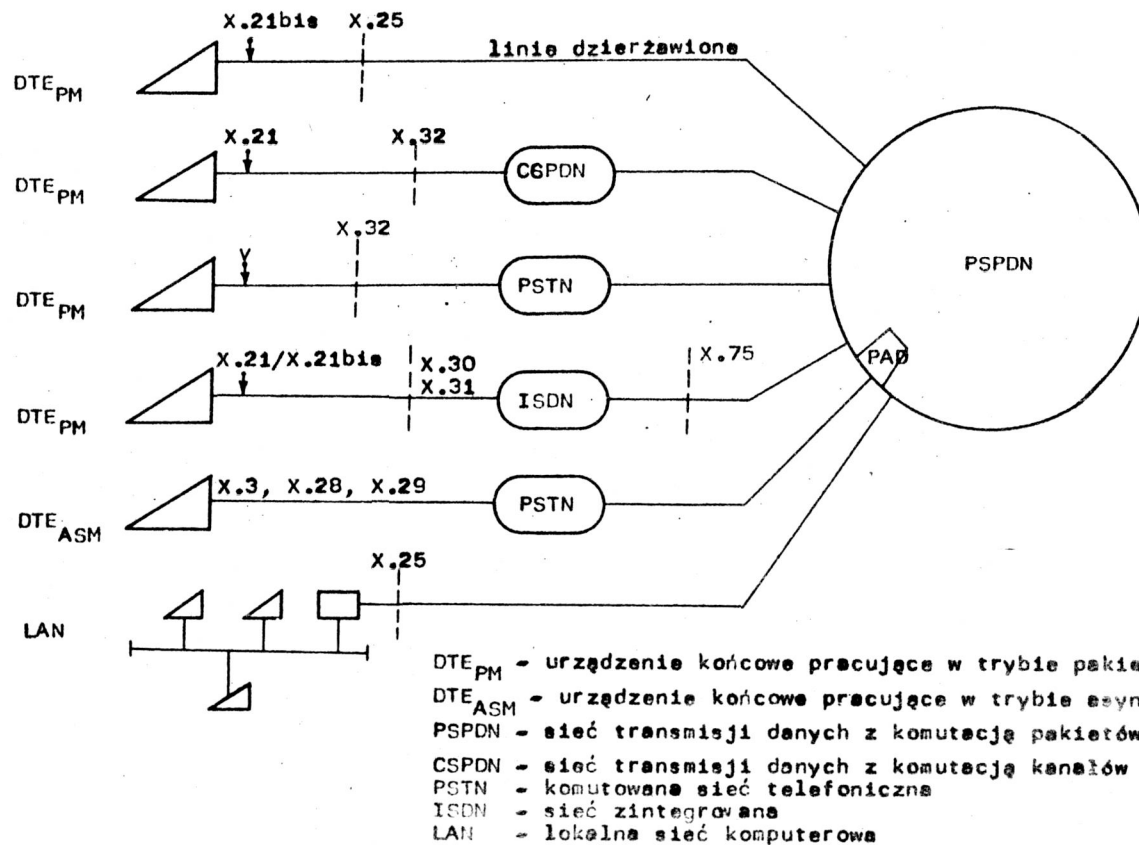
Już przy budowie systemów wielodostępnych (komputer obliczeniowy obsługujący grupę terminali), na początku lat sześćdziesiątych, wystąpił problem wierności przesyłania informacji; zdawano sobie sprawę z tego, że wierność ta musi być absolutna - komputery na poziomie procesów aplikacyjnych są bezradne wobec błędnie przesłanego nawet pojedynczego bitu. Rozważania na ten temat doprowadziły do techniki podziału, przesyłanej informacji, na bloki o stałej długości oraz zastosowania mechanizmów kontroli parzystości poprzecznej i wzdłużnej. Przesyłanie informacji odbywało się zgodnie z protokołami zorientowanymi znakowo. Protokoły te są do dziś stosowane w komunikacji terminal - komputer obliczeniowy, np. ISO-7 w systemach Odra 1300. Mechanizmy te okazały się mało skuteczne przy większych rozmiarach bloków przesyłanej informacji oraz przy przesyłaniu informacji w relacji komputer - komputer. Radykalną poprawę sytuacji uzyskano po wprowadzeniu mechanizmu nadmiarowych kodów cyklicznych (CRC). W oparciu o ten mechanizm powstały protokoły liniowe zorientowane bitowo, np.: SDLC, HDLC i LAP B, które zapewniają przesyłanie informacji z elementową stopą błędu 10^{-9} , wystarczającą obecnie do większości zastosowań sieci komputerowych. Tak niską stopę błędu uzyskuje się dzięki utrzymaniu podziału przesyłanej informacji na bloki - zwane ramkami - sprawdzaniu poprawności przesłania ramki i retransmisji ramek przesłanych z błędami. Protokoły te posiadają ponadto procedury sterowania przepływem, które dostosowują tempo nadawania ramek w urządzeniu nadawczym do aktualnych możliwości odbiorczych urządzenia odbierającego. Zwiększanie liczby urządzeń współpracujących w ramach sieci komputerowej, doprowadziło do zaprojektowania protokołów sieciowych nadbudowanych na protokołach liniowych, umożliwiających współpracę dowolnych urządzeń w sieci. Jeżeli wypadkowa stopa błędu sieci transmisji danych jest niezadawalająca, to wówczas stosuje się jeszcze jeden stopień mechanizmu CRC w warstwie transportowej (4-ta klasa protokołu transportowego). W ten sposób powstały sieci komputerowe

z pakietową techniką przesyłania informacji. Z uwagi na wymaganą wysoką wierność przesyłania informacji oraz opisane wyżej skuteczne sposoby jej osiągania, technika pakietowa jest obecnie kanoniczną techniką stosowaną we wszystkich rodzajach sieci komputerowych.

W początkowej fazie rozwoju rozległych sieci komputerowych (na przełomie lat 1960/1970), były one budowane w oparciu o istniejące środki łączności; były to na ogół dzierżawione linie telefoniczne - rzadziej komutowane linie telefoniczne. W miarę wzrostu liczby użytkowników oraz potrzeby coraz szybszej transmisji, rozwiązania takie stawały się niewystarczające. Dalszy rozwój sieci komputerowych był niemożliwy bez zbudowania odrębnych publicznych sieci transmisji danych. Na początku lat siedemdziesiątych rozpoczęto budowę sieci z komutacją kanałów przeznaczonych wyłącznie do transmisji danych. W 1973 r. w Green Book CCITT, ukazała się rekomendacja X.21, określająca interfejs pomiędzy urządzeniami komputerowymi użytkowników a publiczną siecią transmisji danych z komutacją kanałów (CSPDN). Zastosowanie elektronicznych central komutacji kanałów powodowało, że znacznemu skróceniu uległy czasy nawiązywania połączeń oraz zmniejszył się poziom zakłóceń. Prace nad pakietową techniką przesyłania informacji doprowadziły do publicznej sieci transmisji danych z komutacją pakietów (PSPDN). W 1977 r. w Orange Book CCITT, ukazała się rekomendacja X.25, określająca interfejs pomiędzy urządzeniami komputerowymi użytkowników (DTE) a siecią PSPDN (DCE). Kraje, które wcześniej rozpoczęły budowę CSPDN (REN, Japonia), dokończyły ich budowę i przekazały do publicznej eksploatacji. Natomiast wszystkie kraje rozwinięte zbudowały u siebie sieci PSPDN. Obecnie w krajach rozwiniętych trwają prace nad budową sieci zintegrowanych (ISDN), które w oparciu o impulsową technikę transmisji (PCM), służą do przesyłania mowy, danych, obrazów oraz filmów. Tak więc rozważając sieci transmisji danych, należy uwzględnić wszystkie wymienione wyżej rodzaje sieci (rys. 1). Przed podaniem charakterystyk poszczególnych rodzajów sieci, warto porównać ich podstawowe parametry użytkowe, są one podane w Tablicy 1.

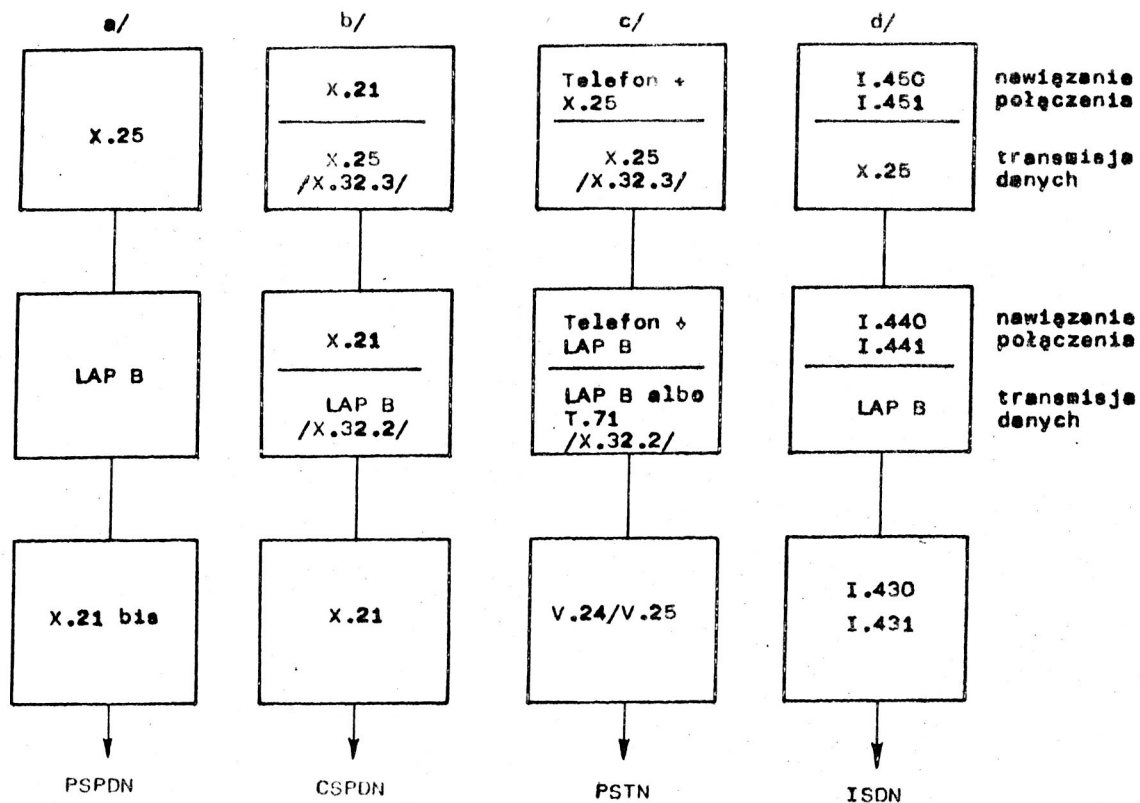
Dla użytkowników ważny jest jeszcze jeden parametr, a mianowicie - koszt przesyłania informacji. Dla poszczególnych rodzajów sieci, koszty te są różne w każdym kraju; wszędzie jednak najdroższe jest przesyłanie poprzez linie dzierżawione, a najtańsze - poprzez komutowane sieci telefoniczne.

Ponadto użytkownicy powinni wiedzieć jakie standardy musi spełniać ich urządzenie końcowe, aby możliwa była współpraca z danym rodzajem sieci transmisji danych; objaśnia to rys. 2.



Rys. 1. Łączenie terminali do PSPDN, poprzez różne rodzaje sieci transmisji danych

Fig. 1. Connection DTE to PSPDN via different kinds of data transmission networks



Rys. 2. Standardy urządzeń końcowych /DTE/ współpracujących z różnymi sieciami transmisji danych

Fig. 2. Standards of DTE's, connected to different data transmission network

Tablica 1. Szybkość transmisji i stopa błędów dla różnych rodzajów sieci transmisji danych.

Table 1. Data signalling rate and error rate in different data transmission networks.

Lp.	Rodzaj sieci	Szybkość transmisji b/s	Elementowa stopa błędów
1.	Telefoniczne linie dzierżawione	do 48 000	ca 10^{-6}
2.	Komutowana sieć telefoniczna	do 4800	$10^{-3} + 10^{-4}$
3.	Sieć transmisji danych z komutacją kanałów	do 9600	ca 10^{-5}
4.	Sieć transmisji danych z komutacją pakietów	do 48 000	ca 10^{-9}

2.1. Dzierżawione linie telefoniczne

Jak wynika z tablicy 1, posiadają one dobre parametry eksploatacyjne; szczególnie ważna jest duża maksymalna szybkość transmisji. Obecnie są one stosowane powszechnie w sieciach komputerowych z komutacją pakietów PSPDN, do realizacji następujących połączeń:

- pomiędzy węzłami komutacji pakietów,
 - komputer obliczeniowy - węzeł,
 - sieciowy koncentrator terminali - węzeł
- oraz bardzo często do realizacji połączeń:
- terminal - sieciowy koncentrator terminali,
 - terminal - komputer obliczeniowy.

Ponieważ w pierwszej grupie połączeń stosowana jest wyłącznie transmisja dwukierunkowa, niezbędne jest dzierżawienie 2-ch par przewodów. Technika komutacji pakietów umożliwia jednoczesną realizację, na jednym 4-przewodowym połączeniu, do 4096 połączeń logicznych; stwarza to możliwość dobrego wykorzystania łącza, dzięki jednoczesnej obsłudze wielu użytkowników. Szybkość transmisji 48 kb/s, wymaga dzierżawienia całej grupy pierwotnej (12 kanałów).

Urządzenia końcowe użytkowników (DTE_{PM}), którymi mogą być komputery obliczeniowe, koncentratory terminali oraz terminale pracujące w trybie pakietowym, muszą spełniać rekomendacje podane na rys. 2a.

X.21 bis [1], stanowi podzbiór V.24 i DTE_{PM} może współpracować z modemami serii V [2], podanymi w tablicy 2. Na poziomie liniowym stosowany jest LAP B a na sieciowym - X.25 [1].

W sieci z komutacją pakietów mogą być realizowane dwa podstawowe rodzaje usług:

- przełączane połączenia logiczne (VC),

Tablica 2. Rodzaje modemów serii V

Table 2. V-series type of modems

Typ modemu	Szybkość transmisji b/s	Tryb pracy	Rodzaj sieci
V.21	do 300, start/stop	dupleks	telefoniczna komutowana
V.22	1200/600 synchroniczna	dupleks	telefoniczna komutowana /oraz dzierżawiona 2-przewodowa/
V.22 bis	2400/1200	dupleks	telefoniczna komutowana /oraz dzierżawiona 2-przewodowa/
V.23	600/1200 synchroniczna lub asynchroniczna	półdupleks albo dupleks	telefoniczna komutowana albo dzierżawiona 4-przewodowa
V.26	2400	dupleks albo półdupleks	dzierżawiona 4-przewodowa
V.26 bis	2400/1200	półdupleks	telefoniczna komutowana
V.26 ter	2400	dupleks	telefoniczna komutowana /oraz dzierżawiona 2-przewodowa/
V.27	4800	dupleks albo półdupleks	dzierżawiona 4-przewodowa
V.27 bis	4800/2400	dupleks albo półdupleks	dzierżawiona 4-przewodowa
V.27 ter	4800/2400	półdupleks	telefoniczna komutowana
V.29	9600	dupleks	dzierżawiona 4-przewodowa
V.32	9600/4800	dupleks	telefoniczna komutowana /oraz telefoniczna dzierżawiona

- stałe połączenia logiczne PVC.

W trakcie korzystania z przełączanych połączeń logicznych realizowane są następujące usługi:

- nawiązywanie połączenia,
- wykrywanie i poprawa błędów transmisji,
- sterowanie przepływem,
- rozłączanie połączenia.

W stałych połączeniach logicznych nie stosuje się nawiązywania i rozłączania połączenia.

Oprócz usług, które są obligatoryjne, sieć transmisji danych z komutacją pakietów, powinna realizować udogodnienia, które są opcjonalne. Rekomendacja X.2 [3], wymienia dla połączeń typu VC sieci pakietowej następujące podstawowe udogodnienia, które powinny być realizowane w skali międzynarodowej:

- negocjacja parametrów sterowania przepływem,
- negocjacja klasy przepustowości,
- wywołania przychodzące zabronione,
- wywołania wychodzące zabronione,
- kanał logiczny jednokierunkowy - wychodzący,
- zamknięta grupa użytkowników.

2.2. Sieci transmisji danych z komutacją kanałów

Usługi CSPDN polegają na ustanawianiu połączenia pomiędzy urządzeniami końcowymi pracującymi w trybie pakietowym, jego utrzymywaniu oraz rozłączaniu połączenia. Usługi te realizowane są zgodnie z rekomendacją X.21 [1]. W fazie transmisji danych mogą być stosowane procedury IAP B i X.25 (rys. 2b), podobnie jak w sieci PSPDN; jest to niezbędne z uwagi na zbyt wysoką stopę błędów występującą w CSPDN. Tak więc za poprawne przesłanie informacji odpowiadają komunikujące się urządzenia końcowe użytkowników. Stosowana jest transmisja dwukierunkowa. Urządzenia końcowo wyposażane są w autowzywaki, które automatycznie nawiązują wymagane połączenie. Z podstawowych udogodnień, które w CSPDN powinny być realizowane w skali międzynarodowej, rekomendacja X.2, wymienia tylko zamkniętą grupę użytkowników.

W 1985 r., w Red Book, ukazała się po raz pierwszy rekomendacja X.32 [1], określająca interfejs pomiędzy DTE i DCE, dla terminali pakietowych włączanych do PSPDN poprzez CSPDN albo PSTN (rys. 1). Rekomendacja ta jest w początkowej fazie opracowania; dotychczas opracowane zostały tylko propozycje identyfikacji i rejestracji zgłaszających się terminali; pozostałe problemy są "for further study". W szczególności zapowiedziano propozycje procedur dla poziomu liniowego i sieciowego; należy się spodziewać, że na poziomie liniowym pozostanie IAP B, dlatego że

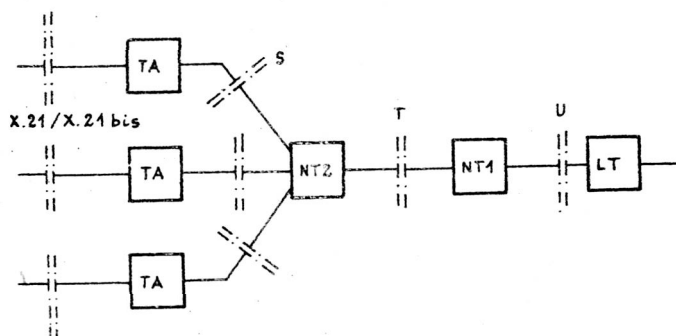
obecnie nie ma nic lepszego; natomiast na poziomie sieciowym mogą być realizowane procedury X.25 jak na rys. 2b, albo uproszczone procedury sieciowe, jak w terminalach TELETEX'u.

2.3. Komutowana sieć telefoniczna

Usługi PSTN polegają na ustanawianiu połączenia, jego utrzymaniu oraz rozłączeniu. Nawiązywanie połączenia odbywa się dwustopniowo, najpierw wg numeru telefonu a następnie pakietem call request wg X.25. Nawiązywanie połączenia wg numeru telefonu może się odbywać ręcznie lub przy pomocy autowzywaka zgodnego z V.25 [2]. Przesyłanie danych może się odbywać zgodnie z procedurami X.25; na poziomie liniowym przewiduje się również możliwość stosowania transmisji półdupleksowej, zgodnie z rekomendacją T.71 [4]. Jak wynika z tablicy 1 sieć PSTN posiada najgorsze parametry użytkowe a dodatkowo mogą występować przerwy w połączeniach. W jednak w przyszłości należy przewidywać jej wykorzystanie w sieciach komputerowych. Nie do pomyślenia jest bowiem doprowadzenie wszędzie gdzie potrzeba, sieci transmisji danych ze stykiem X.25 czy X.21; byłoby to równoznaczne z powtórным zbudowaniem następnych dwóch (z uwagi na duplex) sieci telefonicznych. Należy przewidywać dostęp do sieci także małych zakładów pracy, małych miejscowości czy wsi a przede wszystkim terminali domowych. To chyba z myślą o nich rozpoczęto opracowywanie, wspomnianej już, rekomendacji X.32 z uwzględnieniem opcji transmisji półdupleksowej. Terminale włączane są do sieci poprzez modemy, podane w tablicy 2. Przy okazji należy odnotować rozpoczęcie produkcji modemów akustycznych, umożliwiających realizację sprzężenia terminala z aparatem telefonicznym; modem taki został również opracowany w kraju (modem MAK650, produkowany przez Spółdzielnię "ART" w Szczecinie). Oprócz terminali pracujących w trybie pakietowym, przewiduje się również włączanie do sieci terminali asynchronicznych (DTE_{ASM} - na rys. 1); mogą one być włączane do PSPDN poprzez PSTN lub CSPDN oraz urządzenia przejścia z/na format pakietowy (PAD), które są częścią PSPDN. Współpraca terminali asynchronicznych z PSPDN odbywa się zgodnie ze standardami X.3 [3], X.28 i X.29 [1]. Coraz częściej słyszy się opinie, że proste terminale asynchroniczne o małej szybkości transmisji będą zanikały na korzyść terminali inteligentnych budowanych w oparciu o coraz doskonalsze mikrokomputery osobiste; w związku z tym pod znakiem zapytania stawia się celowość uwzględniania PAD-ów w budowanych obecnie sieciach komputerowych.

2.4. Sieć zintegrowana

Podobnie jak w sieciach CSPDN i PSTN, sieć ISDN realizuje usługi nawiązywania połączenia, utrzymania połączenia na czas transmisji danych oraz rozłączania połączenia, w oparciu o techniki komutacji kanałów. Procedury realizujące te usługi zawarte są w rekomendacjach serii I, przedstawionych na rys. 2d. W fazie transmisji danych mogą być stosowane procedury LAP B oraz procedury sieciowe X.25. Jeżeli terminal posiada "stare" styki X.21/X.21 bis, to jego podłączenie do ISDN odbywa się tak jak przedstawiono na rys. 3.



- TA - adapter terminala,
 NT - zakończenie sieci,
 LT - zakończenie linii,
 S - podstawowy interfejs dostępu do ISDN (2x64 + 16) kb/s,
 T - pierwotny interfejs dostępu do ISDN 2.048 Mb/s.

Rys. 3. Sposób włączenia DTE do ISDN
 Fig. 3. Support of DTE by an ISDN

Szczegółowe rozważania na temat dołączania tego typu terminali do ISDN zawarte są w rekomendacji X.30 [1], która ukazała się po raz pierwszy w Red Book w 1985 r. W tym samym czasie powstała rekomendacja X.31 [1], która zawiera opis włączania terminala pracującego w trybie pakietowym wg X.25 do PSPDN poprzez ISDN (rys. 1). Zgodnie z X.31, interfejs pomiędzy ISDN i PSPDN ma być realizowany wg X.75 [5]. Problemy współdziałania ISDN z X.25 są obecnie w fazie opracowywania; propozycje rozwiązań omawiane były między innymi na Kongresie IFIP we wrześniu 1986 r. i są obszernie przedstawione w referatach [6] i [7].

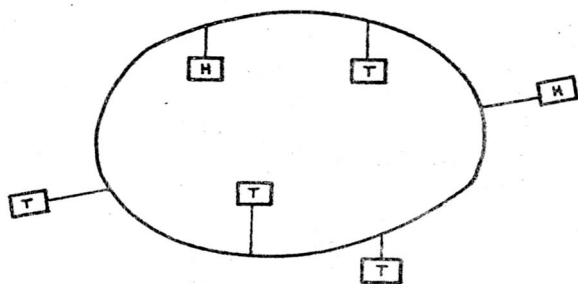
W dniach 9-12 czerwca 1986 r. odbyła się w Londynie międzynarodowa konferencja poświęcona sieciom zintegrowanym. Z referatów wynika, że w krajach rozwiniętych, do 1990 r., połowa przesyłanych informacji, będzie przenoszona poprzez łącza ISDN.

2.5. łączenie lokalnych sieci komputerowych LAN do PSPDN

Interfejsy terminali (DTE) do sieci lokalnych są przedmiotem osobnych referatów i nie będą dalej omawiane. W tym miejscu, należy tylko zwrócić uwagę, że włączanie lokalnych sieci komputerowych do rozległych sieci komputerowych, odbywa się za pomocą interfejsu X.25 (rys. 1). Z uwagi na to, że w sieciach lokalnych może pracować jednocześnie wielu użytkowników, nie stosuje się łączenia LAN do PSPDN poprzez PSTN i CSPDN.

3. SIECI METROPOLITALNE

Dotychczas, rozległe sieci komputerowe na terenach dużych aglomeracji miejskich, budowane są w topologii typu gwiazda. Urządzenia końcowe użytkowników komunikują się ze sobą poprzez centrale komutacji pakietów (węzeł lub węzły). Połączenia urządzeń z centralą realizowane są obecnie na liniach dzierżawionych a w przyszłości również na komutowanych liniach telefonicznych. Taka sieć metropolitalna ma na ogół połączenia z centralami na terenie innych miast, stanowiąc część rozległej sieci krajowej. Ostatnio rozważa się możliwość budowy sieci metropolitalnych typu backbone (kręgosłupowych) (rys. 4). Są to pierścienie obejmujące swoim zasięgiem całe miasto, do których dołączone są komputery obliczeniowe dużej mocy (H) oraz urządzenia terminalowe użytkowników (T).



Rys. 4. Sieć metropolitalna typu backbone
Fig. 4. Metropolitan area network backbone type

Pierścień jest szerokopasmowym kablem koncentrycznym lub światłowodem. W nośniku takim funkcjonuje jednocześnie kilka szybkich kanałów, np.: 10 kanałów po 10 Mb/s każdy. Jest to więc rodzaj sieci lokalnej typu token ring, posiadającej duże rozmiary. Sieć taka, poprzez odpowiedni adapter może być połączona z krajową rozległą siecią komputerową PSPDN.

4. JAKOŚĆ POŁĄCZEŃ SIECIOWYCH

W praktyce mogą występować duże różnice w jakości połączeń sieciowych, uzyskiwane w różnych rodzajach sieci transmisji danych lub w różnych wykonaniach sieci tego samego rodzaju. Jakość połączenia sieciowego oceniana jest wypadkową stopą błędu sieci transmisji danych (DCE), a więc wynikająca zarówno z jakości linii łączących, jak i pośredniczących w połączeniu węzłów lub/i central komutacji, a także częstotliwością upadków połączenia, a więc częstotliwością rozłączeń lub zerowań. Parametry liczbowe dotyczące jakości połączeń sieciowych podane są w X.213 oraz ISO/DIS8348. Ponadto wprowadzony został przez ISO podział połączeń sieciowych na trzy następujące typy:

- A - połączenia sieciowe z akceptowalną stopą błędu i akceptowalną częstością występowania, sygnalizowanych rozłączeń lub zerowań;
- B - połączenia sieciowe z akceptowalną stopą błędu i nieakceptowalną częstością sygnalizowanych rozłączeń lub zerowań;
- C - połączenia sieciowe z nieakceptowalną przez użytkownika stopą błędu.

Urządzenia końcowe użytkowników (DTE), mogą być w znacznym stopniu uniezależnione od jakości połączeń sieciowych. Mechanizmy uniezależnienia mogą być implementowane w DTE, w protokole transportowym, znajdującym się na 4-tym poziomie Modelu Odniesienia. W ramach prac ISO, opracowano protokół transportowy zawierający 5 klas (0÷4); szczegóły można znaleźć np. w [8]. W dokumencie tym zalecane jest stosowanie, w zależności od typu wykorzystywanego połączenia sieciowego, następujących klas protokołu:

- typ A - klasa 0 lub 2;
- typ B - klasa 1 lub 3;
- typ C - klasa 4.

Klasy protokołu 1, 3 i 4 zawierają mechanizmy utrzymywania połączenia transportowego (pomiędzy urządzeniami końcowymi), podczas upadku połączenia sieciowego oraz samoczynnego uruchamiania procedur nawiązywania nowego połączenia sieciowego. Klasa 4 posiada ponadto procedury CRC, które podobnie jak na poziomie liniowym, wykrywają i korygują błędy występujące w czasie transmisji. Warto zwrócić uwagę, że w przypadku sieci transmisji danych złej jakości (połączenia sieciowe typu C), sieć komputerowa funkcjonuje, ale czasy przesyłania informacji, zwłaszcza transfery dużych zbiorów, mogą być wielokrotnie dłuższe niż w sieciach transmisji danych dobrej jakości.

5. WNIOSKI

Na podstawie powyższych rozważań można sformułować następujące wnioski:

- 1 W sieciach komputerowych, podstawową techniką przesyłania informacji jest technika pakietowa; wynika to z wymaganej bardzo wysokiej wierności przesyłania informacji pomiędzy urządzeniami komputerowymi.
- 2 Sieci transmisji danych z komutacją pakietów PSPDN są i w najbliższych latach będą podstawą do budowy sieci komputerowych. W szczególności urządzenia końcowe z interfejsem X.25, będą podstawowym sprzętem w sieciach komputerowych.
- 3 Docelowo, powszechną siecią transmisji danych dla sieci komputerowych, będą sieci zintegrowane (ISDN), służące jednocześnie do przesyłania danych, głosu, obrazów oraz filmów.
- 4 Docelowo należy przewidywać konieczność łączenia terminali pracujących w trybie pakietowym do komputerów obliczeniowych, poprzez komutowaną sieć telefoniczną (PSTN).

6. LITERATURA

- [1] CCITT Red Book, Vol. VIII, Fascicle VIII.3 - Data communication networks: interfaces. Recommendations X.20 - X.32. Geneva 1985.
- [2] CCITT Red Book, Vol. VIII, Fascicle VIII.1 - Data communication over the telephone network. Series V Recommendations. Geneva 1985.
- [3] CCITT Red Book, Vol. VIII, Fascicle VIII.2 - Data communication networks: services and facilities. Recommendation X.1 - X.15. Geneva 1985.
- [4] CCITT Red Book, Vol. VII, Fascicle VII.3 - Terminal equipment and protocols for telematic services. Series T Recommendations. Geneva 1985.
- [5] CCITT Red Book, Vol. VIII, Fascicle VIII.4 - Data communication networks: transmission, signaling and switching, network aspects, maintenance and administrative arrangements. Recommendations X.40 - X.181. Geneva 1985.
- [6] Eckhart Raubold - ISDN and X.25. Invited Paper. Materiały IFIP 1986.
- [7] Shuji Tomita - ISDN and X.25. Response Paper. Materiały IFIP 1986.
- [8] Sieci komputerowe - usługi, protokoły, modele. Część 1. Mat.Konf., Wrocław 1985.

SERVICES AND STANDARDS OF DATA TRANSMISSION NETWORKS

In the paper review of the services and standards of different kinds of data transmission networks is presented. All the problems from users point of view are described. In particular PSPDN, CSDN, PSTN and ISDN, and connections between these networks are presented. Also the standards users DTE's in different cases of connections are indicated. Finally some conclusions for present projects of computer networks are suggested.

УСЛУГИ И СТАНДАРТЫ СЕТЕЙ ПЕРЕДАЧИ ДАННЫХ

В статье представлен актуальный просмотр услуг и стандартов для разного типа сетей передачи данных. Эти проблемы в статье представлены с точки зрения пользователей. В частности представлены свойства сетей передачи данных с коммутацией пакетов, коммутацией каналов, коммутированных телефонных сетей, интегрированных сетей и связей между ними. Описаны также стандарты терминалов. В заключении внесены предположения относительно проектов новых сетей ЕВМ.

lokalne sieci komputerowe
metody dostępu
projektowanie

Leszek BORZEMSKI^x, Adam GRZECH^x, Andrzej KASPRZAK^x

PROJEKTOWANIE LOKALNYCH SIECI KOMPUTEROWYCH, WYBÓR METODY DOSTĘPU DO MEDIUM TRANSMISYJNEGO

W pracy przedstawiono zagadnienie wyboru metody dostępu do medium transmisyjnego lokalnej sieci komputerowej. Projektowanie lokalnej sieci komputerowej, poprzedzane analizą wymagań użytkowników, prowadzi do takiej konstrukcji sieci, która jest, według pewnego kryterium, najlepsza dla klasy zadań generowanych przez użytkowników sieci. Analiza wymagań użytkowników obejmuje zarówno jakościowe jak i ilościowe wymagania. Rezultatem tej pierwszej są przede wszystkim wymagania na system przetwarzania danych, podczas gdy wynikiem analizy wymagań ilościowych są parametry rozwiązywania w określonej klasie rozwiązań sieciowych. System automatycznego przetwarzania danych dla przewidywanych zastosowań projektowanej sieci lokalnej jest uzależniony od algorytmów przetwarzania zadań użytkowników, metod podziału zasobów sieciowych i algorytmów przydziału zasobów sieciowych. Dla ustalonych algorytmów przetwarzania zadań oraz metod podziału zasobów sieciowych, badany jest związek między algorytmami przydziału zasobów sieciowych a metodami przydziału zasobów komunikacyjnych oraz zagadnienie ich wzajemnego dopasowania. Przyjmując, że projektowana lokalna sieć komputerowa jest siecią magistralową a kanał transmisyjny pracuje w trybie rozgłoszeniowym, przeanalizowano przydatność znanych algorytmów dostępu do medium transmisyjnego do realizacji typowych transakcji przydziału zasobów sieciowych.

1. WPROWADZENIE

Rozproszony system komputerowy to system złożony z wielu autonomicznych jednostek przetwarzających ogólnego i specjalnego przeznaczenia, połączonych systemem komunikacyjnym. System komunikacyjny umożliwia w tym przypadku nie tylko komunikację pomiędzy poszczególnymi jednostkami systemu ale także podział i przydział zasobów systemu. W rezultacie, zadania lokalne mogą być przekazywane i przetwarzane w różnych (odległych) jednostkach systemu. Przyczyną przekazywania zadań może być konieczność skorzystania z zasobów niedostępnych lokalnie lub możliwość wykonania zadania w korzystniejszych warunkach (np. w krótszym czasie) [24].

^x Instytut Sterowania i Techniki Systemów, Politechnika Wrocławska,
Wrocław

Rozwój technik mikrokomputerowych i telekomunikacyjnych oraz potrzeba przybliżenia zasobów komputerowych do miejsc powstawania i wykorzystywania informacji wpłynęły na rozwój systemów wieloprocesorowych: systemów wielodostępnych i sieci komputerowych. Obydwa wymienione rozwiązania są, w pewnym sensie, skrajnymi realizacjami systemów wieloprocesorowych: pierwsze są orientowane na przetwarzanie scentralizowane podczas gdy drugie na przetwarzanie rozproszone tj. obsługę zadań użytkowych i dostęp do zasobów komputerowych w systemie rozproszonego przetwarzania danych [4,11].

Rozwój sieci komputerowych, będący następstwem postępu technologii i rozumienia korzyści wynikających ze stosowania przetwarzania rozproszonego, doprowadził do wyodrębnienia dwóch grup rozwiązań sieciowych: rozległych (zdalnych, publicznych) sieci komputerowych i lokalnych sieci komputerowych (m.in. [16]). Podstawą takiego podziału jest uniwersalność i specjalizacja odpowiednio rozległych i lokalnych sieci oraz różne mechanizmy wykorzystywane do organizacji komunikacji między zasobami sprzętowo-programowymi komputerów sieci. Podczas gdy pod względem własności warstw wyższych architektury logicznej brak jest istotnych różnic w obydwu wymienionych grupach rozwiązań, to dla lokalnych sieci komputerowych charakterystyczne jest znaczne uproszczenie systemu komunikacyjnego w porównaniu z systemem komunikacyjnym sieci rozległych. Z tego też względu, podstawą do podziału sieci lokalnych są charakterystyczne dla ich systemów komunikacyjnych: struktura topologiczna, metody dostępu do medium transmisyjnego oraz medium transmisyjne wykorzystywane w sieci [5,7,13,24].

O jakości dostarczanych przez lokalną sieć komputerową usług, podobnie jak i w przypadku rozległych sieci komputerowych, decyduje stopień dopasowania systemu komunikacyjnego sieci do rodzaju zadań realizowanych w sieci oraz wymagań użytkowników związanych z generowanymi przez nich zadaniami użytkowymi.

Rodzaj zadań obsługiwanych przez lokalną sieć komputerową oraz wymagania użytkowników (jakościowe i ilościowe) związane z obsługiwanymi zadaniami, decydują o algorytmach automatycznego przetwarzania i metodach dostępu do zasobów [3]. Zarówno algorytmy automatycznego przetwarzania jak i metody dostępu do zasobów (metody rozmieszczenia zasobów i metody ich udostępniania), tworząc system automatycznego przetwarzania, określają wymagania pod adresem systemu komunikacyjnego sieci. Wymagania te mogą być spełnione w lokalnej sieci komputerowej poprzez dobór struktury topologicznej sieci, metody dostępu do medium i medium transmisyjnego. Przyjmując, że o doborze struktury, metody dostępu oraz medium decydują w głównej mierze odpowiednio: wymagania niezawodnościowe i technologiczne, wymagania jakościowe systemu automatycznego

przetwarzania oraz wymagania ilościowe użytkowników, istotne jest zadanie doboru odpowiedniej metody dostępu do medium transmisyjnego wykorzystywanego w lokalnej sieci komputerowej do systemu automatycznego przetwarzania będącego rezultatem analizy wymagań użytkowników. Zadanie to jest elementem znacznie ogólniejszego problemu dopasowania systemu komunikacyjnego lokalnej sieci komputerowej do rodzaju wykonywanych zadań i wymagań związanych z wykonywanymi zadaniami. Zadanie to, jak i wymienione wcześniej bardziej szczegółowe zadanie może być sformułowane dwójako:

- dla danego systemu komunikacyjnego sieci taka organizacja obsługi zadań, która maksymalizuje stopień spełnienia wymagań użytkownika, lub
- dla danych zadań i ustalonych wymagań użytkowników taki system komunikacyjny który, dla przyjętego kryterium, jest optymalny w założonej grupie dopuszczalnych rozwiązań.

Zasadność tak sformułowanych zadań (dobór systemu komunikacyjnego lub jego poszczególnych elementów dla pewnej grupy zadań) w przypadku lokalnych sieci komputerowych wynika z tego, że realizacja zadań w lokalnej sieci komputerowej jest bezpośrednio uzależniona od metod dostępu a tym samym od metod przydziału zasobów komunikacyjnych. Podobne zadania, w przypadku rozległych sieci komputerowych, są znacznie bardziej złożone gdyż stosowane tam mechanizmy sterowania przepływem na różnych poziomach, metody doboru tras, złożone mechanizmy potwierdzeń, itd., tworząc złożony system sterowania komunikacją, utrudniają ekstrakcję związków jakości usług z poszczególnymi, stosowanymi w sieci mechanizmami.

Prace nad doбором odpowiednich metod dostępu do systemu komunikacyjnego lokalnej sieci komputerowej, dostosowanych do klasy zadań realizowanych w sieciowym systemie rozproszonego przetwarzania danych, prowadzone są w związku z poszukiwaniem algorytmów sieciowych dla zadań identyfikacji i rozpoznawania [1] w pracach nad wykorzystaniem lokalnej sieci komputerowej NETEX [2] do wspomagania prac badawczych i dydaktyki.

2. LOKALNE SIECI KOMPUTEROWE

Jakkolwiek brak jest jednoznacznych i powszechnie uzawanych definicji lokalnych sieci komputerowych, to istnieje szereg ich własności charakterystycznych, spośród których eksponowane są między innymi następujące: wspólnie wykorzystywane medium transmisyjne, duże szybkości transmisji, niska stopa błędów, geograficzne (przestrzenne) rozproszenie zasobów, itd.

Istotą tych definicji jest odnoszenie własności charakterystycznych dla lokalnych sieci komputerowych do rozwiązań właściwych dla sieci dużych.

Porównując te dwie grupy rozwiązań sieciowych, dla lokalnych systemów sieciowych charakterystyczne są [19]:

- duża szybkość transmisji (do 10 Mbit/s) i niska stopa błędów (10^{-6}),
- bardzo prosty mechanizm potwierdzeń - w większości przypadków, potwierdzenie pojedynczych pakietów (wiadomości),
- krótkie wiadomości z krótkimi nagłówkami oraz brak konieczności segmentowania,
- minimalne wymagania na sterowanie siecią, przepływem i przeciążeniami jako konsekwencja bardzo prostej struktury topologicznej,
- brak konieczności wyznaczania tras, jako następstwo wykorzystywania jednego medium transmisyjnego,
- małe opóźnienia wnoszone przez system transmisyjny z powodu małych odległości i dużych szybkości transmisji, oraz
- bardzo proste mechanizmy adresowania z powodu prostej struktury topologicznej.

Możliwość odrzucenia konieczności stosowania złożonych procedur sterowania i zarządzania systemem komunikacyjnym, charakterystycznych dla rozległych sieci komputerowych, powoduje, że system komunikacyjny lokalnej sieci komputerowej jest stosunkowo prosty i do opisu jego architektury wystarcza opis jego struktury topologicznej, metody dostępu do medium transmisyjnego i samego medium transmisyjnego [22].

Trzy, wymienione wyżej elementy opisu, są także podstawą do podziałów dokonywanych w grupie lokalnych sieci komputerowych.

3.PROJEKTOWANIE LOKALNYCH SIECI KOMPUTEROWYCH

Projektowanie dowolnego systemu komputerowego, w tym także sieci komputerowych, jest zawsze związane z koniecznością dokonywania wyboru pomiędzy różnymi rozwiązaniami, zaspakajającymi różnym kosztem w różnym stopniu wymagania użytkowników. Z powodu mnogości alternatywnych rozwiązań, konieczne jest zdyscyplinowane postępowanie, pozwalające na precyzyjne zdefiniowanie zadania projektowania i otrzymanie rozwiązania tego zadania. Proces projektowania lokalnej sieci komputerowej można podzielić na następujące etapy [3]:

1. Analiza wymagań, celem której jest kodyfikacja żądań i oczekiwań pod adresem sieci, pozwalająca na zdefiniowanie zadania projektowania oraz wybór różnych alternatyw rozwiązania. Analiza ta dotyczy tak istniejących jak i przewidywanych wymagań.
2. Formułowanie kryterium wyboru, celem uzyskania "wspólnego mianownika" umożliwiającego porównywanie alternatywnych rozwiązań.
3. Wybór rozwiązań alternatywnych wśród rozwiązań dopuszczalnych.
4. Iluściowy opis otrzymanych rozwiązań alternatywnych.
5. Wybór rozwiązania spośród różnych dopuszczalnych rozwiązań.

Spółród pięciu wyżej wymienionych etapów, najistotniejszy jest etap pierwszy, gdyż pozostałe etapy, bazując na rezultatach etapu pierwszego służą praktycznie do weryfikacji i oceny rozwiązań wygenerowanych na podstawie przeprowadzonej analizy wymagań. Etap pierwszy jest także najtrudniejszy, gdyż w przeciwieństwie do czterech dalszych, których realizacja może być w znacznym stopniu oparta na metodach formalnych (analitycznych lub/i symulacyjnych) [6,10,23], tu muszą być wzięte pod uwagę także wymagania niewymierne.

Analiza wymagań, obejmująca żądania (wymagania które muszą być spełnione) i oczekiwania (wymagania które mogą być spełnione) dotyczy wymagań jakościowych i ilościowych.

Rezultatem analizy wymagań jakościowych i ilościowych są odpowiednio: kierunek poszukiwania rozwiązania oraz obszar rozwiązań.

O kierunku poszukiwania rozwiązania zadania projektowania decydują:

- podstawowe atrybuty rozwiązania których spełnienie jest niezbędne aby rozwiązanie mogło być przyjęte jako dopuszczalne (klasy użytkowników, rodzaje zasobów, rozwiązania technologiczne, ograniczenia ekonomiczne, wymagane dostosowanie do istniejących standartów, itp.)
- pożądane cechy automatycznego systemu przetwarzania danych (scentralizowany, rozproszony, hierarchiczny, itd.), oraz
- dopuszczalne lub przewidywane zmiany zastosowań, technologii, standartów, możliwe do przewidzenia zmiany wymagań w następstwie zastosowania określonego rozwiązania, itd).

Określony przez wymienione wyżej grupy kierunek poszukiwania rozwiązania jest uszczegóławiany w wyniku analizy wymagań ilościowych. Stopień uszczegółowienia zależy od możliwości ilościowego oszacowania:

- liczby i lokalizacji zasobów sprzętowych (istniejących i przewidywanych),
- charakterystyk strumieni danych w sieci,
- charakterystyk jakościowych usług, oraz
- charakterystyk sprzęgów sprzętowych i funkcjonalnych.

Dwupoziomowa, hierarchiczna analiza wymagań użytkowników jest konsekwencją tego, że zmiana wymagań jakościowych może w znacznej mierze zmienić klasę w której poszukiwane są rozwiązania, podczas gdy zmiana wymagań ilościowych powoduje zmianę parametrów w klasie rozwiązań (np. zmianę parametrów pojedynczych i sumarycznego strumieni danych generowanych przez użytkowników).

W analizie wymagań jakościowych najistotniejszym elementem jest określenie klasy automatycznego systemu przetwarzania danych. O ile bowiem podstawowe atrybuty rozwiązań i dopuszczalne oraz przewidywane zmiany stanowią zbiór ograniczeń które muszą lub powinny być spełnione, o tyle przyjęcie określonego systemu automatycznego przetwa-

rzania ma decydujący wpływ na organizację przetwarzania i komunikacji w sieci, a dalej na charakterystyki ilościowe projektowanego systemu. O systemie automatycznego przetwarzania danych decydują (Rys. 1):

- algorytmy przetwarzania danych,
- podział zasobów, oraz
- algorytmy dostępu do zasobów.

Algorytmy przetwarzania danych, określone są przez przewidywane zastosowania sieci tzn. przez rodzaje zadań użytkowników. Charakterystyka algorytmów polega na sprecyzowaniu ich możliwych dekompozycji, możliwych realizacji równoległych w następstwie przyjętej dekompozycji i wymagań na realizację zadania (komunikacja między procesami) [12]. Wymienione wyżej trzy elementy są wzajemnie związane w ten sposób, że im głębsza dekompozycja zadania, tym większa ilość możliwych operacji równoległych i tym większe zapotrzebowanie na transmisję między odpowiednimi procesami. Realizacja określonych algorytmów przetwarzania danych wymaga udostępnienia określonych zasobów, głównie procesorów i pamięci ale także zasobów transmisyjnych w przypadku przestrzennego rozproszenia tych pierwszych. Ponieważ w środowisku sieciowym wiele algorytmów (korzystających ze wspólnych zasobów) może być wykonywanych jednocześnie, wymagane jest istnienie algorytmów dostępu do wspólnych zasobów, uwzględniających wzajemne relacje pomiędzy różnymi zadaniami oraz relacje pomiędzy częściami tego samego zadania. Zadaniem tych algorytmów jest rozstrzygnięcie sytuacji konfliktowych powstających w rezultacie ubiegania się o te same zasoby, w tym samym czasie, przez więcej niż jeden procesor. W sytuacji gdy dostęp do zasobów odbywa się poprzez odpowiednie korzystanie z systemu komunikacyjnego, algorytmy dostępu do zasobów jednoznacznie określają sposoby korzystania z systemu komunikacyjnego.

W lokalnych sieciach komputerowych, których system komunikacyjny oparty jest na jednym, wspólnie wykorzystywanym medium transmisyjnym, sposoby korzystania z systemu komunikacyjnego sprowadzają się w rzeczywistości do metod przydziału jednego kanału komunikacyjnego poszczególnym zadaniam użytkownika.

Tak więc zadanie przydziału zasobów w lokalnej sieci komputerowej sprowadza się do przydziału dwóch kategorii zasobów: zasobów sieciowych (rozumianych jako zbiór procesorów, pamięci, urządzeń specjalizowanych, oprogramowania, itd) umożliwiających bezpośrednią realizację przetwarzania danych (według odpowiednich algorytmów przetwarzania) oraz zasobów komunikacyjnych, umożliwiających pośrednio realizację przetwarzania poprzez przekazywanie wyników i danych niezbędnych na poszczególnych etapach realizacji algorytmów. Zasoby komunikacyjne wykorzystywane są także do przekazywania informacji dla celów synchronizacji

poszczególnych operacji przetwarzania. Dla zbiorów algorytmów przetwarzania danych, dla których został ustalony podział zasobów sieciowych istotne jest to, w jakim stopniu mechanizm dostępu do systemu komunikacyjnego pełni funkcje synchronizacyjne dla realizowanych algorytmów. O przydatności mechanizmu dostępu do systemu komunikacyjnego dla określonego zbioru algorytmów, realizowanego w lokalnej sieci komputerowej, decyduje ich wzajemne dopasowanie. W skrajnych przypadkach mogą to być następujące sytuacje:

- brak dopasowania tzn. wszelkie funkcje synchronizacyjne realizowane są poza systemem komunikacyjnym, a zadania systemu komunikacyjnego ograniczone są do przenoszenia nadawanych informacji,
- pełne dopasowanie tzn. stosowane metody dostępu do systemu komunikacyjnego umożliwiają nie tylko przenoszenie nadawanych informacji ale również wybór i realizację komunikacji między procesami o odpowiednich cechach (funkcje synchronizacyjne).

Jak zwykle w takich sytuacjach wszystkie rozwiązania pośrednie pomiędzy skrajnymi, włącznie z tymi ostatnimi, mogą być stosowane ale konsekwencją mniejszego lub większego dopasowania są odpowiednio niższa lub wyższa jakość dostarczanych usług oraz mniej lub bardziej efektywne wykorzystanie zasobów komunikacyjnych sieci.

Za jak największym dopasowaniem przemawia także fakt, że z uwagi na stosunkowo niewielką moc obliczeniową instalowanych w sieciach lokalnych mikro lub/i mikrokomputerów, jedną z naczelnych zasad w trakcie projektowania jest dążenie do jak najmniejszego obciążania stacji sieci funkcjami komunikacyjnymi.

Reasumując, można stwierdzić, że w przypadku lokalnych sieci komputerowych, jako rozproszonego systemu przetwarzania danych, celem analizy wymagań jakościowych jest określenie cech automatycznego systemu przetwarzania a następnie odpowiedniej metody dostępu do zasobów komunikacyjnych (Rys. 1).

4. ANALIZA WYMAGAŃ JAKOŚCIOWYCH NA SYSTEM KOMUNIKACJI LOKALNEJ SIECI KOMPUTEROWEJ

Typowe zastosowania lokalnych sieci komputerowych [3,22] wymagają udostępnienia użytkownikom następujących możliwości [3,5] :

- zbieranie danych z procesów rzeczywistych i symulacyjnych,
- konwersja danych,
- transmisja danych z miejsca ich zbierania lub/i przechowywania do miejsca ich przetwarzania lub/i przechowywania,
- rozproszone przetwarzanie (przetwarzanie lokalne, scentralizowane lub hierarchiczne), oraz
- przechowywanie danych i programów przetwarzania.

Zastosowania lokalnych sieci komputerowych obejmują systemy automatyzacji prac biurowych, zarówno administracyjnych, jak i komputerowego wspomaganie projektowania, w bankach, centralach handlowych, instytutach badawczych i biurach projektowo-konstrukcyjnych, systemy rozszerzonego wielodostępu w centrach obliczeniowych wyposażonych w duży system komputerowy, systemy zbierania i przetwarzania danych w czasie rzeczywistym, systemy sterowania procesami przemysłowymi, transportem, itd. Mimo różnych zastosowań, większość z wymienionych wyżej usług jest typowa dla wszystkich lokalnych sieci komputerowych, co najwyżej różnią się proporcje pomiędzy wykorzystywanymi typami usług [3]. Analizując wymienione, typowe usługi lokalnych sieci komputerowych, można wyodrębnić zbiór elementarnych transakcji związanych z udostępnianiem określonych zasobów za pośrednictwem systemu komunikacyjnego sieci. Są to następujące transakcje:

1. $1 \rightarrow 1$ tzn. transmisja informacji typu punkt-punkt. Transakcja ta jest charakterystyczna dla systemu w którym nadawca informacji posiada dokładną informację o adresie odbiorcy.
2. $1 \rightarrow m$ tzn. transmisja informacji w trybie rozgłoszeniowym, w trakcie której informacja generowana przez nadawcę przeznaczona jest dla więcej niż jednego odbiorcy (transakcja charakterystyczna dla poczty elektronicznej).
3. $1 \text{ z } n(d) \rightarrow 1$ tzn. transmisja informacji przez uporządkowany zbiór źródeł do pojedynczych adresatów (może być do tego samego) realizowana po kolei, zgodnie z ustalonym dla źródeł porządkiem (transakcja charakterystyczna dla systemu zbierania informacji, w którym istotna jest kolejność nadawania przez źródła).
4. $1 \text{ z } n(p) \rightarrow 1$ tzn. transmisja informacji przez zbiór źródeł do pojedynczych adresatów z tym, że w przypadku konfliktu źródeł wybierane jest źródło o największej wartości cechy charakterystycznej (transakcja charakterystyczna dla systemów zbierania informacji w których żądania generowane przez źródła mają różne priorytety a także dla systemów w których o jeden rodzaj zasobów może ubiegać się szereg procesów o różnych priorytetach). Istotą tej transakcji jest to, że przydział zasobów komunikacyjnych jest poprzedzany wyborem jednego ze źródeł wtedy gdy więcej niż jedno generuje żądanie transmisji.
5. $1 \rightarrow 1 \text{ z } m(d)$ tzn. transmisja informacji przez jedno źródło do jednego z ujęć należących do uporządkowanego zbioru ujęć. Transmisje realizowane są po kolei, zgodnie z uporządkowaniem zbioru ujęć. Dla tej transakcji istotne jest uporządkowanie zbioru adresatów.

6. $1 \rightarrow 1$ z $m(p)$ tzn. transmisja informacji przez pojedyncze źródło do jednego z ujść, będącego elementem zbioru ujść, z tym, że o wyborze tego ostatniego decyduje stan ujść (transakcja charakterystyczna dla systemu w którym pojedyncze źródło ubiega się o dostęp do zbioru jednakowych zasobów, z tym, że zajętość tych zasobów stwarza konieczność wyboru np. zasobu wolnego lub mało obciążonego).
7. 1 z $n(d) \rightarrow 1$ z $m(d)$ tzn. transmisja informacji typu punkt-punkt, z tym, że istotne jest zachowanie kolejności par źródło-ujście biorących udział w transmisji (transakcja charakterystyczna dla sekwencyjnych algorytmów przetwarzania).
8. 1 z $n(d) \rightarrow 1$ z $m(p)$ tzn. transmisja informacji z zachowaniem uporządkowania po stronie źródeł, podczas gdy ujście wybierane jest każdorazowo ze zbioru ujść na podstawie bieżącego ich stanu.
9. 1 z $n(p) \rightarrow 1$ z $m(d)$ tzn. informacje transmitowane są przez źródła do zbioru ujść udostępnianych w pewnym, założonym porządku. Wybór po stronie źródła odbywa się na podstawie porównywania jakości źródła w przypadku konfliktu pomiędzy źródłami.
10. 1 z $n(p) \rightarrow 1$ z $m(p)$ tzn. informacje transmitowane typu punkt do punktu, z tym, że po stronie źródeł konieczny jest wybór w przypadku konfliktu, a po stronie ujść wybór jest uzależniony od stanu poszczególnych odbiorców. Transakcja to jest charakterystyczna dla systemu w którym wiele jednakowych zasobów udostępnianych jest jednocześnie źródłom żądań mogących mieć różne priorytety.

Wymienione transakcje charakteryzuje różny sposób przydziału zasobów sieciowych i komunikacyjnych, pozwalający na ich klasyfikację. I tak dla transakcji (1) i (2) charakterystyczne jest to, że podział zasobów sieciowych jednoznacznie determinuje przydział zasobów sieciowych (tzn. pary źródło-ujście lub źródło-ujścia są ustalane ostatecznie w trakcie podziału zasobów) i jedynym zasobem rzeczywiście przydzielanym w czasie realizacji transakcji są zasoby transmisyjne. Realizacjom transakcji (3) i (4) towarzyszy przydzielanie zasobów sieciowych i komunikacyjnych. Konieczność przydzielania zasobów sieciowych wynika z tego, że podział zasobów określa tu klasę użytkowników mogących korzystać z danego zasobu, natomiast sposób korzystania przez użytkowników danej klasy z danego zasobu jest rozstrzygany przez algorytm przydziału. W pierwszym przypadku, (3), zadaniem algorytmu przydziału jest nadzorowanie wcześniej ustalonego porządku, podczas gdy w drugim, (4), jest to rzeczywisty przydział "na bieżąco". Przydział zasobów transmisyjnych jest tutaj poprzedzany przydziałem zasobów sieciowych po stronie źródeł transmisji.

Realizacja transakcji (5) i (6) przebiega podobnie do realizacji transakcji (3) i (4), z tym, że podział zasobów sieciowych definiuje grupę zasobów dla danego użytkownika.

Warunki realizacji transakcji (7)-(10) są o tyle inne od wcześniejszych że tutaj w wyniku podziału zasobów ustalane są grupy użytkowników korzystających z grup zasobów sieciowych i stąd bardziej rozbudowane zadania, konieczne do spełnienia przez algorytmy przydziału.

O ile w transakcjach (3) i (4) oraz (5) i (6) przydział zasobów transmisyjnych poprzedzany jest przydziałem zasobów sieciowych odpowiednio po stronie źródeł i ujść, to w transakcjach (7)-(10) przydział zasobów sieciowych konieczny jest tak po stronie źródeł jak i ujść. W transakcjach (5)-(10) przydział zasobów po stronie ujść może być realizowany przed lub po przydziale zasobów komunikacyjnych. To czy zasoby sieciowe po stronie ujść przydzielane są przed czy po przydziale zasobów komunikacyjnych, nie pozostaje bez wpływu na organizację systemu komunikacji: w pierwszym przypadku, po niezbędnych negocjacjach, transmisja ma charakter transmisji punkt-punkt; w drugim przypadku, po stronie ujść musi istnieć centrum decyzyjne wybierające dla odebranej informacji ostatecznego adresata (transmisja dwuetapowa).

Wybór odpowiedniego sposobu przydziału zasobów komunikacyjnych, będący jednym z zadań projektowania lokalnej sieci komputerowej, może być zatem związany z poszukiwaniem takiego sposobu przydziału zasobów komunikacyjnych, który jednocześnie mógłby rozwiązywać zadania przydziału zasobów sieciowych. Wiąże się to z koniecznością odpowiedzi na następujące pytania:

Jakie możliwości w tym zakresie posiadają znane metody dostępu stosowane w lokalnych sieciach komputerowych?,

Czy istnieje możliwość skonstruowania takiej metody dostępu która rozwiąże w pełni zadanie przydziału zasobów sieciowych?, oraz

Jakie metody dostępu powinny być zastosowane do realizacji wybranej transakcji wtedy gdy nie istnieje metoda dostępu bezpośrednio rozwiązująca wymagania danej transakcji?

W dalszych rozważaniach przyjmujemy, że system komunikacyjny lokalnej sieci komputerowej jest oparty na wielodostępnej magistrali pracującej w trybie rozgłoszeniowym.

5. METODY DOSTĘPU A REALIZACJA WYBRANYCH TRANSAKCJI

W celu zilustrowania możliwości wykorzystania znanych metod dostępu do rozwiązywania zadań przydziału zasobów sieciowych, założymy, że wykorzystywane są następujące metody dostępu do medium transmisyjnego:

- CSMA/CD (ang. Carrier Sense Multiple Access with Collision Detection),
- przekazywanie znacznika (ang. token-bus), oraz

- TDMA (ang. Time Division Multiple Access).

Metody te zostały wybrane spośród wielu znanych metod dostępu [7, 14, 15, 17, 20 i 21], jako najczęściej wykorzystywane w istniejących instalacjach sieciowych. Przydatność tych metod do rozwiązywania typowych zadań przydziału zasobów ilustruje Tabela 1. Wybrane metody reprezentują trzy grupy metod dostępu tj. metody losowe (ang. contention), metody wyboru (ang. selection) i metody rezerwacji (ang. reservation) [14]. (W Tabeli 1 "x" oznacza, że dana metoda dostępu w pełni spełnia wymagania określonej transakcji. Lewa i prawa kolumna odnoszą się odpowiednio do podstawowego i zmodyfikowanych algorytmów danej metody.)

Transakcja	CSMA/CD		przekazywanie znacznika		TDMA	
1 → 1	x	x	x	x	x	x
1 → m	x	x	x	x		
1 z n(d) → 1		x	x	x	x	x
1 z n(p) → 1		x	x	x		
1 → 1 z m(d)				x		
1 → 1 z m(p)				x		
1 z n(d) → 1 z m(d)				x		
1 z n(d) → 1 z m(p)				x		
1 z n(p) → 1 z m(d)				x		
1 z n(p) → 1 z m(p)				x		

Tabela 1. Przydatność metod dostępu do realizacji transakcji.
Table 1. Access-control methods ability to solve transactions.

Zawartość Tabeli 1, wyraźnie świadczy o małej przydatności znanych metod dostępu do rozwiązywania zadań związanych z poszczególnymi transakcjami. Względnie duża przydatność metody przekazywania znacznika, to rezultat wbudowanego w metodę mechanizmu priorytetowania. Rozwiązywanie transakcji typu (3) przy pomocy tej metody można osiągnąć przez przypisanie wszystkim stacjom z grupy n(d) najwyższego priorytetu, niedostępnego dla innych użytkowników. Obsługa transakcji (4) nie wymaga żadnych zmian w tej metodzie.

Mała przydatność omawianych metod dostępu wynika z tego, że nie dysponują one mechanizmem umożliwiającym wybór po stronie ujęć.

Poprzez modyfikacje algorytmu CSMA/OD tzn. poprzez uzupełnienie podstawowego mechanizmu o mechanizm rezerwacji (np. [17, 18]) lub wyboru (np. [8]) można uzyskać metody rozwiązujące odpowiednio zadania transakcji (3) i (4). Podobnie, dobudowanie do podstawowej, statycznej metody TDMA mechanizmu dynamicznej rezerwacji (np. [14]), daje możliwość obsługi transakcji (4).

Analiza wymienionych wcześniej trzech metod dostępu do kanału transmisyjnego, pracującego w trybie rozgłoszeniowym, prowadzi do wniosku, że do obsługi transakcji (4)-(10) możliwość poszukiwania odpowiednich metod dostępu istnieje tylko w grupie metod wyboru, dla której to grupy charakterystyczne jest przeszukiwanie przed udzieleniem pozwolenia na transmisję. Wprowadzając do metody przekazywania znacznika dodatkowy mechanizm umożliwiający priorytetowanie po stronie ujęć, istnieje możliwość, poprzez realizację dwustopniowej procedury przeszukiwania [9], obsługi transakcji w których przydział zasobów komunikacyjnych jest poprzedzany przydziałem zasobów bądź po stronie ujęć bądź po stronie źródeł i ujęć.

Zadanie doboru odpowiedniej metody dostępu do zasobów komunikacyjnych dla określonych algorytmów przydziału zasobów sieciowych jest z gądnieniem wtórnym do tego w jaki sposób organizowane są algorytmy przydziału zasobów sieciowych.

Algorytmy przydziału zasobów sieciowych można podzielić biorąc za podstawę sposób gromadzenia informacji o stanie zasobów w sieci oraz ilość informacji o stanie zasobów jaką posiadają poszczególni użytkownicy. Algorytmy te można podzielić na scentralizowane i rozproszone, w obydwu grupach dokonując dalszego podziału w zależności od ilości informacji o zasobach (Rys. 2).

Dla poszczególnych, wybranych przypadków charakterystyczne jest to, że:

- przydział zasobów sieciowych w systemie w pełni scentralizowanym oznacza, że istnieje centrum w którym gromadzone są informacje o zasobach i ich stanie a realizacja wszelkich przydziałów odbywa się poprzez korzystanie z usług tego centrum.
- przydział zasobów sieciowych w systemie lokalnie scentralizowanym polega na tym, że istnieją ustalone centra gromadzące informacje o zasobach i ich stanie. Oznacza to, że w sytuacji gdy istnieje jedno centrum, system jest systemem scentralizowanym, natomiast w sytuacji gdy liczba centrów równa jest liczbie użytkowników system jest systemem rozproszonym w którym lokalnie gromadzone są informacje o zasobach i ich stanie jednego użytkownika.
- przydział zasobów w systemie rozproszonym z całkowitą informacją o zasobach i ich stanie oznacza, że całą informację o zasobach posiada źródło żądające dostępu do określonego zasobu, a tym samym wszystkie możliwe żądania mają jednoznaczne adresy.
- przydział zasobów w systemie całkowicie rozproszonym tzn. wtedy gdy informacja o zasobach i ich stanie gromadzona jest tylko w miejscach przechowywania zasobów oznacza, że przyznanie określonych zasobów wymaga przeglądnięcia wszystkich możliwych rozwiązań i wybranie spośród nich tego które zapewnia najwyższą jakość.

Przyjęcie określonego systemu przydziału zasobów (scentralizowanego, rozproszonego, itd.) wpływa na sposób realizacji systemu komunikacyjnego a tym samym decyduje o niezbędnych do spełnienia transakcjach. Te ostatnie z kolei decydują o metodach przydziału zasobów komunikacyjnych - w rozważanym w tej pracy przypadku - o metodach dostępu do kanału transmisyjnego lokalnej sieci komputerowej.

6. PRZYKŁAD

Dla ilustracji zagadnienia dyskutowanego w tej pracy, rozważmy przykład systemu, przedstawionego na Rys. 3, będącego typowym przykładem systemu rozproszonego [12].

Wybór architektury lokalnej sieci komputerowej w przypadku systemu jak na Rys. 3, sprowadza się do wyboru struktury topologicznej, metody dostępu do medium transmisyjnego i medium transmisyjnego.

Założmy, że w omawianym przykładzie zadania realizowane są lokalnie a w razie konieczności skorzystania z zasobów rozmieszczonych w modułach pamięci (1-m), generowane jest żądanie dostępu do pamięci. Obsługa żądania generowanego przez procesor polega na odpowiednim transferze z pamięci do pamięci lokalnej danego procesora. Zakładamy dalej, że zawartość modułów pamięci (1-m) jest taka sama, a żądania generowane przez procesory mają różne priorytety uzależnione od zadań wykonywanych przez poszczególne procesory. Przyjmując także, że żądania procesorów mają priorytet wyższy od żądań pamięci (żądania pamięci mają jednakowy priorytet), typowe dla tego przykładu są następujące transakcje:

- $1 \text{ z } n(p) \rightarrow 1 \text{ z } m(p)$

Transakcja towarzysząca generacjom żądań przez procesory, wtedy gdy spośród kolidujących żądań należy wybrać żądanie o najwyższym priorytecie i przesłać je do modułu pamięci który jest wolny lub najmniej obciążony, oraz

- $1 \rightarrow 1$

Transakcja towarzysząca generacjom żądań wtedy gdy nie występują kolizje oraz generacjom odpowiedzi przez moduły pamięci.

Dla skupienia uwagi, rozważmy możliwe rozwiązania dla obsługi żądań procesorów zakładając różne sposoby gromadzenia informacji o zasobach i ich stanie.

1. Jeżeli w danym systemie istnieje centrum posiadające wszystkie informacje o stanie wszystkich zasobów sieciowych, to realizacja transakcji $1 \text{ z } n(p) \rightarrow 1 \text{ z } m(p)$ sprowadza się do realizacji dwustopniowej procedury:

- transmisja żądania (żądań) do centrum i wybór właściwego, oraz
 - wybór modułu pamięci i retransmisja wcześniej wybranego żądania.
- Oznacza to, że wyżej wymieniona transakcja została zdekomponowana na

dwie prostsze transakcje: $1 \text{ z } n(d) \rightarrow 1$ oraz $1 \rightarrow 1$ (Rys. 4a). Uproszczenie i dekompozycja transakcji początkowej wynika stąd, że konieczność wyboru źródła i ujścia zawarta w transakcji początkowej została przeniesiona z metody dostępu do centrum.

2. Jeżeli w danym systemie założyć istnienie dwóch centrów posiadających odpowiednio wszystkie informacje o stanie procesorów i pamięci, to realizacja omawianej transakcji sprowadza się do realizacji trzy-stopniowej procedury:

- transmisja żądania (zadań) do centrum procesorów i wybór jednego z zadań według wcześniej przyjętego kryterium,
- transmisja wybranego żądania do centrum pamięci, oraz
- wybór modułu pamięci i retransmisja żądania.

Oznacza to, że w tym przypadku transakcja początkowa została zdekomponowana na trzy prostsze transakcje tj. $1 \text{ z } n(d) \rightarrow 1$, $1 \rightarrow 1$ i $1 \rightarrow 1$ (Rys. 4b).

3. Jeżeli w rozważanym systemie każdy z procesorów i modułów pamięci dysponuje całkowitą informacją o stanie zasobów sieciowych, to obsługa żądań sprowadza się do ustalenia przez każdy z procesorów jednoznacznego adresu modułu pamięci a zadaniem metody dostępu do systemu komunikacyjnego jest rozstrzygnięcie o realizowanym żądaniu. oznacza to, że transakcja początkowa jest zastąpiona przez transakcję $1 \text{ z } n(p) \rightarrow 1$. Wybór modułu pamięci do którego adresowane będzie żądanie został przeniesiony z metody dostępu do poszczególnych procesorów generujących żądania (Rys. 4c). Sposób realizacji transakcji $1 \text{ z } n(p) \rightarrow 1$ został szczegółowo omówiony w [24].

4. Jeżeli w omawianym systemie każdy z procesorów i modułów pamięci dysponuje informacją tylko o własnych zasobach, to obsługa żądań musi uwzględnić przegląd wszystkich zasobów i żądań w ramach metody dostępu do systemu komunikacyjnego. Transakcja początkowa, w tym przypadku, nie może być zdekomponowana i zastąpiona przez złożenie kilku prostszych transakcji jak to miało miejsce w omawianych wcześniej przypadkach (Rys. 4d).

5. Jeżeli w danym systemie dostęp do modułów pamięci realizowany jest przez centrum pamięci, to transakcja początkowa może być zrealizowana dwuetapowo w postaci transakcji $1 \text{ z } n(p) \rightarrow 1$ oraz $1 \rightarrow 1$ (Rys. 4e).

Przedstawione wyżej różne możliwości rozwiązywania zadania przysądu zasobów sieciowych w zależności od sposobu gromadzenia informacji o zasobach i ich stanie nie wyczerpują wszystkich możliwych w tym przypadku sposobów dekomponowania transakcji początkowej. Ilustrują one jedynie to, że istnieje wiele sposobów na rozwiązanie tak sformułowanego zadania. Jak z tego wynika pytanie o metodę dostępu jest jedynie częścią bardziej złożonego pytania o architekturę lokalnej

sieci komputerowej przeznaczonej do realizacji określonych zadań. Te ostatnie decydują o typie transakcji, będącej podstawą do poszukiwania struktury topologicznej sieci i metody dostępu.

Z przedstawionego przykładu wynika, że zadanie projektowania lokalnej sieci komputerowej, a dokładnie jej systemu komunikacyjnego, może być postawione dwojako:

- dla przyjętego sposobu gromadzenia informacji o zasobach i ich stanie oraz znanej transakcji początkowej znaleźć strukturę sieci i metodę dostępu umożliwiającą realizację transakcji, lub
- dla znanych: struktury sieci i metody dostępu oraz znanej transakcji początkowej znaleźć taki sposób gromadzenia informacji o zasobach i ich stanie, aby umożliwić realizację transakcji początkowej.

W pierwszym z wyżej przedstawionych zadań, sposób gromadzenia informacji decyduje o możliwych dekompozycjach transakcji początkowej, natomiast w zadaniu drugim posiadane możliwości wyboru wśród metod dostępu mogą decydować o strukturze systemu gromadzenia informacji o zasobach i ich stanie.

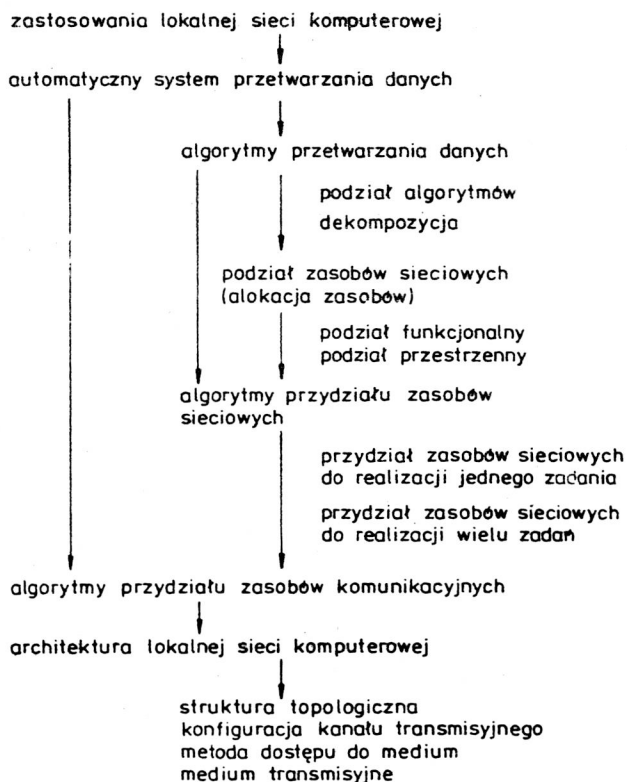
Możliwość realizowania określonej transakcji przez systemy sieciowe o różnej strukturze topologicznej w których wykorzystywane są różne metody dostępu do medium transmisyjnego powoduje konieczność posiadania miary "dobroci" poszczególnych, możliwych rozwiązań. Do tego celu mogą być przydatne powszechnie wykorzystywane miary jakości takie jak średnia przepustowość, średnie opóźnienie, itd., a zadaniem ich byłoby rozstrzygnięcie o przyjęciu lub odrzuceniu danego rozwiązania

7. PODSUMOWANIE

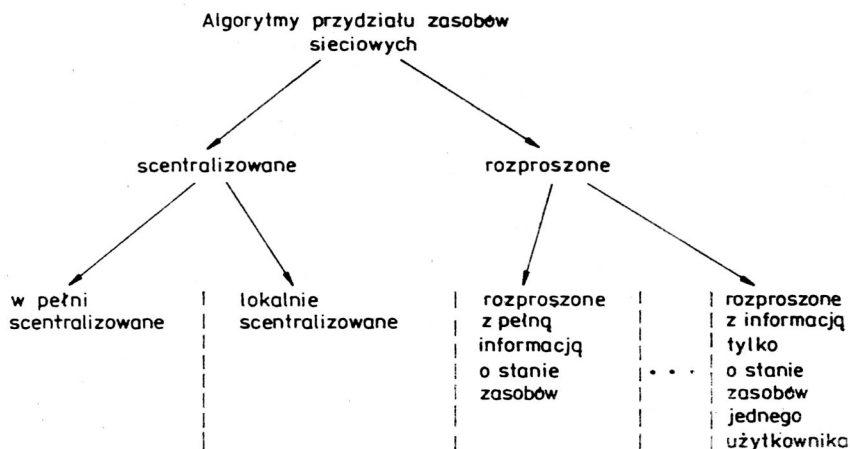
Rozważane w pracy zagadnienie dopasowania metod dostępu do zasobów komunikacyjnych lokalnej sieci komputerowej oraz algorytmów przydziału zasobów sieciowych dla założonego zbioru algorytmów przetwarzania i ustalonego podziału zasobów sieciowych jest związane z wyborem określonego typu lub typów transakcji, realizowanych przez sieć, a następnie wyboru takich metod dostępu które umożliwiają realizację ustalonej transakcji.

Zaproponowano zbiór podstawowych transakcji charakterystycznych dla najczęściej spotykanych zastosowań sieci komputerowych oraz oceniono przydatność powszechnie wykorzystywanych metod dostępu, reprezentujących różne grupy metod, do realizacji poszczególnych transakcji.

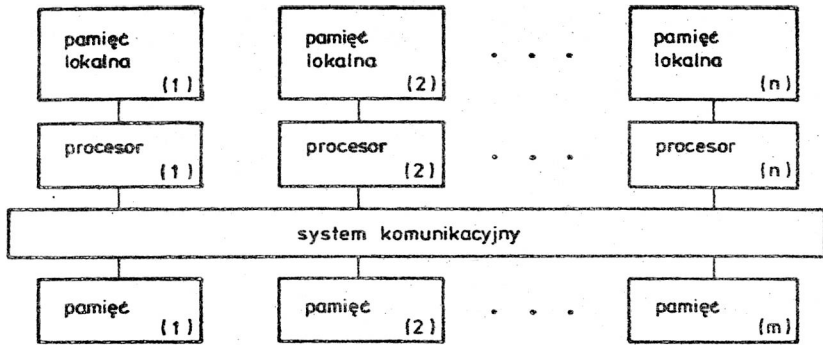
Na podstawie przykładu, wykorzystanego do zilustrowania prowadzonych rozważań, pokazano, że zadanie doboru metody dostępu do kanału transmisyjnego jest częścią bardziej ogólnego zadania doboru architektury lokalnej sieci komputerowej. Możliwość doboru struktury topologicznej i metody dostępu jednocześnie, daje szansę realizacji określonej



Rys. 1. Analiza wymagań jakościowych.
Fig. 1. Qualitative requirements analysis.

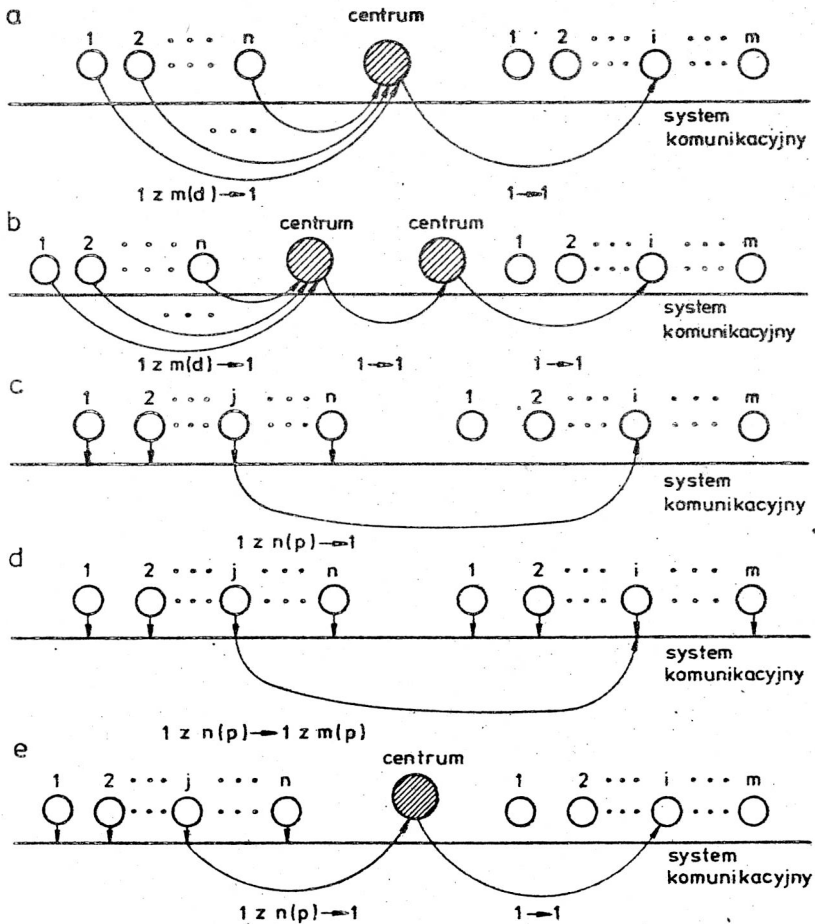


Rys. 2. Podział algorytmów zasobów sieciowych.
Fig. 2. Network resources scheduling algorithms division.



Rys. 3. System rozproszony do połączenia procesorów i pamięci.

Fig. 3. Distributed system for connecting processors and memory.



Rys. 4. Obsługa żądań w omawianym przykładzie.

Fig. 4. Access service in discussed example.

transakcji w prostszy sposób niż wtedy gdy założona jest struktura topologiczna sieci.

Dalsze prace w tym kierunku będą miały na celu znalezienie wskaźników ilościowych umożliwiających porównanie różnych rozwiązań i wybór najlepszego spośród dopuszczalnych.

8. LITERATURA

- [1] Z. Bubnicki, "Algorytmy identyfikacji systemu o strukturze sieciowej dla wybranych przypadków", Raporty Instytutu Sterowania i Techniki Systemów Politechniki Wrocławskiej, Wrocław 1986.
- [2] L. Borzemski, A. Grzech, A. Kasprzak, "Wykorzystanie sieci NETEX w laboratorium dydaktycznym i badawczym", IX Szkoła Mikroprocesorowa, Łódź, październik 1986.
- [3] L. Borzemski, A. Grzech, A. Kasprzak, L. Koszałka, "Development of distributed applications for experiment and education support based on the NETEX local area network", COMNET'85, Budapest.
- [4] L. Borzemski, A. Grzech, A. Kasprzak, "Stan i kierunki rozwoju sieci lokalnych dla potrzeb szkolnictwa wyższego", IX Szkoła Mikroprocesorowa, Łódź, październik 1986.
- [5] M. Budka, "Modele i usługi sieci lokalnych", Sieci komputerowe. Usługi, protokoły, modele. Część 1, Wydawnictwo Politechniki Wrocławskiej, Wrocław 1985.
- [6] P. Bulaban, K. S. Shanmugan, B. W. Stuck, "Computer-aided modelling, analysis and design of communication systems: Introduction and issue overview", IEEE Journal of Selected Areas in Communication, vol. SAC-2, No. 1, January 1984.
- [7] D. D. Clark, K. T. Pograd, D. P. Reed, "An introduction to local area networks", Proceedings of the IEEE, vol. 66, No. 11, November 1978.
- [8] A. Grzech, "Group hybrid access-control discipline for multiaccess channels - a throughput analysis", Raport Instytutu Sterowania i Techniki Systemów Politechniki Wrocławskiej, Wrocław 1986.
- [9] A. Grzech, "Modyfikacje algorytmu przekazywania znacznika", Raport Instytutu Sterowania i Techniki Systemów Politechniki Wrocławskiej, Wrocław 1986.
- [10] J. G. Hottzman, K. S. Shanmugan, G. J. Minden, C. Townsend, V. S. Frost, "Communication system simulation: theory, application and design example", Fourth International Conference on Systems Engineering, Coventry 1985.
- [11] R. E. Kahn, "Resource-sharing computer communication networks", Computer networks: a tutorial, Proceedings of the IEEE, vol. 60, No. 11, November 1972.
- [12] L. Kleinrock, "Distributed systems", Communication of the ACM, vol. 28, November 1985.
- [13] J. M. Kryskov, C. K. Miller, "Local area networks overview - Part 1: definitions and attributes", Computer Design, February 1981.
- [14] E. C. Luczak, "Global bus computer communication techniques", Proceedings of Computer Networking Symposium, December 1978.
- [15] R. M. Metcalfe, D. R. Boggs, "Ethernet: Distributed packet switching for local computer networks", Communications of the ACM, vol. 19, 1976.
- [16] J. K. Peacock, J. W. Wong, E. G. Manining, "Distributed simulation using a network of processors", Computer Networks 3, 1979.
- [17] I. Rubin, "Access control disciplines for multiaccess broadcast channels. Reservation and TDMA schemes", IEEE Transactions on Information Theory, vol. IT-25, September 1979.
- [18] I. Rubin, "Group random-access disciplines for multi-access broadcast channels", IEEE Transactions on Information Theory, vol. IT-24, September 1978.
- [19] N. F. Schneidewind, "Interconnecting local networks to long distance

- networks", Computer, vol. 16, No. 9, September 1983.
- [20] N.C. Strole, "A local communication network based on interconnected token-access rings: A tutorial", IBM Journal of Research Developments, vol. 27, No. 5, September 1983.
 - [21] F.A. Tobagi, W.B. Hunt, "Performance analysis of Carrier Sense Multiple Access with Collision Detection", Computers Networks 4, 1980.
 - [22] C.D. Tsao, "A local area network architecture overview", IEEE Communications Magazine, vol. 22, No. 8, August 1984.
 - [23] R.E. Wachs, M.J. Corrigan, "ARTES: in support C3 network design and evaluation", Proceedings of Computer Networking Symposium, December 1978.
 - [24] B.W. Wah, Jie-Yong Jaung, "Resource scheduling for local computer systems with a multi-access network", IEEE Transactions on Computers, vol. C-34, No. 12, December 1985.

LOCAL AREA NETWORK DESIGN - ACCESS-CONTROL DISCIPLINE ADJUSTMENT PROBLEM

In the paper, the problem of access-control discipline adjustment in local area network designing process is discussed. Assuming that the tasks generated by users are served by distributed computer system, i.e., local area network, both distributed data processing algorithms and network resource allocation determines network resource scheduling algorithms. Because the network resource scheduling is performed via network communication system, the problem which arises is: what kind of communication system access-control discipline should be adjusted for different kinds of network resource scheduling schemes. To answer this question, ten typical transactions that are usually performed in local area networks to meet resource scheduling gains, have been extracted. It was shown that some well-known access-control disciplines (CSMA/CD, token-passing and TDMA) to bus-type broadcast channel have very limited capability to serve the extracted transactions. A simple architecture for connecting processors and memory modules via local area network was used as an example to illustrate the discussed problem. The presented example was also used to show possible decompositions of initial transaction into simple transactions chain which may be solved using known access-control mechanisms.

ПРОЕКТИРОВАНИЕ ЛОКАЛЬНЫХ СЕТЕЙ ЭВМ - ВЫБОР МЕТОДА ДОСТУПА К КОММУНИКАЦИОННОМУ СРЕДСТВУ

В работе представлена проблема выбора метода доступа к коммуникационному средству локальной сети ЭВМ. Если проектированию локальной сети ЭВМ предшествует анализ требований пользователей, можно найти такую конструкцию сети, которая согласно определенному критерию является самой лучшей. Анализ требований пользователей охватывает требования как количественные так и качественные. Результатом качественных требований является прежде всего требование к системе обработки данных, а результатом анализа количественных требований являются параметры решения в определенном классе сетевых решений.

Для определения алгоритмов обработки требований а также методов разделения сетевых ресурсов исследуется связь между алгоритмами приделения сетевых ресурсов и методами приделения коммуникационных ресурсов, а также проблема их взаимного соответствия.

Проблемы представленные авторами работы иллюстрируются примером.

Marian BUDKA ^x
Jerzy WOJTUSZEK ^x

lokalne sieci komputerowe,
rozległe sieci komputerowe,
adaptery między-
sieciowe

WSPÓŁPRACA LOKALNYCH I ROZLEGŁYCH SIECI KOMPUTEROWYCH-ZASADY I SPOSOBY IMPLEMEN- TACJI

W pracy przedstawiono zasady współpracy lokalnych i rozległych sieci komputerowych, na bazie prac standaryzacyjnych prowadzonych na świecie. Zaproponowano rozwiązanie systemu sprzęgającego wg. tych zasad. Dla "dopasowania" protokołów komunikacyjnych sieci lokalnych i rozległych zaproponowano sposób doboru parametrów i opcji protokołów komunikacyjnych w sieciach lokalnych, a w szczególności sposób adresacji NSAP-ów w sieci LAN, przydział numerów kanałów logicznych oraz dobór rozmiaru pakietu w LAN. Przedstawione rozwiązanie będzie wykorzystane dla łączenia sieci LAN/RASH i WAN/KASK.

1. WSTĘP

Podstawowym celem stosowania sieci komputerowych zarówno lokalnych jak i rozległych, jest zapewnienie użytkownikom sieciowym szybkiego i niezawodnego dostępu do zasobów, zazwyczaj zdalnych i dzielonych pomiędzy wielu użytkowników. Dla minimalizacji liczby operacji związanych z fizyczną transmisją w sieciach, dąży się do takiego skonfigurowania lokalnych sieci, aby gros transmisji sieciowych odbywało się w obrębie poszczególnych sieci LAN.

Z uwagi na koszty zainstalowania i utrzymania zasobów, zazwyczaj "właściciel" sieci lokalnej nie jest w stanie zapewnić swoim użytkownikom wszystkich, potrzebnych im zasobów w obrębie swojej sieci LAN. Stąd też potrzeba umożliwienia dostępu do zasobów innych sieci użytkownikom sieci lokalnych.

Odbywa się to bądź przez łączenie szeregu lokalnych sieci komputerowych, bądź też poprzez łączenie sieci LAN z siecią rozległą-WAN.

Idea tworzenia sieci rozległych - WAN obejmuje konieczność zapewnienia "masowego użytkownika" dla efektywnego wykorzystania zasobów zainstalowanych w sieci. Łączenie wielu lokalnych sieci z siecią rozle-

^x Zakład Telekomunikacji Politechniki Śląskiej, Gliwice

głą jest jednym ze sposobów spełnienia tego wymogu.

W niniejszej pracy przedstawiono ogólne zasady współpracy lokalnych i rozległych sieci komputerowych, wyspecyfikowano problemy doboru parametrów i opcji protokołów komunikacyjnych w sieciach lokalnych dla współpracy z sieciami rozległymi X.25 oraz przedstawiono architekturę systemu sprzęgającego sieci lokalne w sieciach WAN. Przedstawione w pracy problemy i propozycje ich rozwiązania przewidywane są do wykorzystania przy łączeniu lokalnych sieci LAN/KASK z Krajową Akademicką Siecią Komputerową KASK oraz z sieciami regionalnymi.

2. ZASADY WSPÓŁPRACY LOKALNYCH I ROZLEGŁYCH SIECI KOMPUTEROWYCH

Łączenie lokalnych i rozległych sieci komputerowych musi zapewnić obok możliwości komunikowania się pomiędzy odpowiednimi komponentami urządzeniowo-programowymi obydwóch typów sieci, wysoką efektywność poszczególnych sieci lokalnych, sieci rozległej oraz wymiany informacji pomiędzy nimi. Różne media transmisyjne i różne zasady transmisji w tych dwóch typach sieci komputerowych prowadzą do konieczności rozstrzygnięcia szeregu kompromisów.

Jednym z podstawowych problemów są różne długości pakietów w lokalnych i rozległych sieciach komputerowych. W sieciach LAN z uwagi na dużą szybkość transmisji, dużą odporność na zakłócenia i 32-bitowe zabezpieczenie kodowe preferowane są bloki o długości do 1512 bajtów. Pozwala to na uzyskanie wysokiej efektywności przesłań.

W kolei w sieciach rozległych, przy niskiej jakości łączy i 16-to bitowym zabezpieczeniu kodowym dopuszczane są do transmisji bloki - pakiety o długości znacznie niższej, zwykle 128 lub 256 bajtów.

Przyjęcie kompromisowe odnośnie długości pakietów w sieciach lokalnych i rozległych musi uwzględnić również złożoność tzw. bram IWU-LAN/WAN X.25 Interworking Unit oraz zagadnienie opóźnień wprowadzanych przez nie.

Podstawowym zadaniem bramy - IWU jest "dopasowanie" zasad komunikacji, w obydwóch typach sieci komputerowych, przy zapewnieniu maksymalnej efektywności pracy sieci. Oznacza to dopasowanie lub konieczność transformacji protokołów komunikacyjnych obowiązujących w sieciach lokalnych i sieci rozległej.

Z istoty mediów transmisyjnych protokołów warstwy fizycznej różni sieci lokalne od rozległych. Różne są również zasady dostępu, ponieważ w sieci LAN istnieje wspólne medium transmisyjne, zaś w sieci WAN - podsieć komunikacyjna pracująca wg zasady stopniowego przesy-

łania (store-and-forward).

Specyfika transmisji w sieciach rozległych oraz w sieciach lokalnych wykorzystujących media szerokopasmowe, pozwala na ujednolicenie zasad komunikacji - protokołów komunikacyjnych w warstwach - powyżej drugiej. Z uwagi na brak podsieci komunikacyjnej w sieci LAN, protokoły poziomu LLC są zbliżone do protokołów poziomu pakietowego. Stąd też w materiałach normatywnych [2] przewiduje się implementację obsługi w IWU trzech dolnych warstw architektury sieci.

Idea łączenia sieci lokalnych z siecią rozległą jest przedstawiona na rys. 2.1 [2], na tle warstwowego modelu sieci LAN i WAN. Zgodnie z tą ideą protokoły warstw od trzeciej w górę w sieciach lokalnych i sieciach rozległych są identyczne.

Trzy podstawowe problemy związane ze współpracą sieci LAN i WAN na poziomie właściwego styku tj. poziomie pakietowym, rozważono w dalszej części pracy. Są to zagadnienia:

- adresacji NSAP-ów w sieci LAN
- przydziału numerów kanałów logicznych
- oraz
- rozmiaru pakietu w sieci LAN.

Dopasowanie zasad obowiązujących w sieciach LAN i WAN na poziomie pakietowym, jest możliwe poprzez dobór odpowiednich parametrów i opcji z zestawu dopuszczalnych. Takie podejście zaprezentowano w niniejszej pracy.

3. DOBÓR PARAMETRÓW I OPCJI PROTOKOŁÓW KOMUNIKACYJNYCH W SIECIACH LOKALNYCH DLA WSPÓŁPRACY Z SIECIAMI ROZLEGŁYMI X.25

3.1. Adresacja NSAP-ów w sieci LAN

Współpraca sieci WAN z siecią LAN wymaga przyjęcia określonych ustaleń odnośnie adresowania NSAP-ów (tj. punktów dostępu do usług warstwy sieciowej) w obu sieciach.

Ustalenia te powinny być zgodne ze standardem [1].

Adres NSAP-u powinien identyfikować dany NSAP w obrębie całej sieci globalnej tj. sieci utworzonej przez połączenie sieci LAN z siecią WAN. Przydzielanie adresów NSAP-ów powinno zatem być koordynowane przez Administrację sieci globalnej.

Sieć LAN stanowi zwykle część sieci globalnej zarządzaną przez oddzielną jednostkę organizacyjną nazywaną dalej Administracją sieci LAN. Aby zapewnić Administracji sieci LAN pewną niezależność od Administracji sieci globalnej w przydzielaniu adresów NSAP-ów, sieć LAN powinna tworzyć domenę adresową w sieci globalnej. Oznacza to zgodnie z [1], że adres każdego NSAP-u sieci LAN składać się będzie

z dwu części (rys.3.1):

IDP (Initial Domain Part) - część określająca domenę adresową w sieci globalnej),

DSP (Domain Specific Part) - część identyfikująca NSAP w obrębie domeny adresowej.

Część IDP dzieli się dalej na:

AFI (Authority and Format Identifier) - część określająca format części IDI oraz syntaktykę części DSP,

IDI (Initial Domain Identifier) - identyfikator domeny adresowej.

Format części IDI i jej zawartość określa Administracja sieci globalnej, natomiast syntaktykę i zawartość części DSP - Administracja sieci LAN.

Pośród kilku zaproponowanych w [1] formatów IDI najważniejszym dla sieci budowanych obecnie w kraju wydaje się format odpowiadający standardowi CCITT X.121.

W formacie tym IDI jest sekwencja cyfr o maksymalnej długości równej 14. Część DSP może być określana za pomocą dwu rodzajów syntaktyk: binarnej i dziesiętnej. Wybór syntaktyki DSP zależy od specyfiki sieci LAN, jednakże wygodniejszą dla większości przypadków wydaje się syntaktyka dziesiętna, w której wartość DSP jest wyrażana za pomocą cyfr dziesiętnych. Liczba tych cyfr nie powinna być większa niż 24. Każdej kombinacji formatu IDI i syntaktyki DSP odpowiada dwucyfrowa wartość części AFI. Dla formatu IDI zgodnego z X.121 i syntaktyki dziesiętnej DSP wartość ta wynosi 36.

Adresy NSAP-ów w sieci WAN powinny być tworzone według tych samych reguł, jednakże może w nich być pomijana część DSP. Dopuszcza się więc istnienie domen adresowych zawierających tylko jeden NSAP.

Odpowiednio zakodowane adresy NSAP-ów przesyłane mogą być w polach AF (Address Field) lub i AEF (Address Extension Facility) pakietów ustanawiających połączenia logiczne w warstwie sieciowej. Rozmiary pól AF mogą okazać się niewystarczające do zakodowania w nich pełnych adresów. W związku z powyższym proponuje się aby w pakietach pomiędzy IWU i stacjami sieci LAN adresy te umieszczone były w polu AEF [2], [3]. Pole AF nie będzie wówczas wykorzystane. Każda cyfra adresu NSAP-u zapisywana będzie w polu AEF na jednym półokciecie - zgodnie z regułami kodowania dziesiętnego przedstawionymi w [1]. Część IDP powinna zostać uzupełniona, o ile jest to konieczne, do maksymalnej liczby cyfr za pomocą zer prowadzących.

Powyższe zasady kodowania adresów NSAP-ów można przyjąć również dla pakietów przesyłanych pomiędzy stacjami w sieci LAN nie będącymi IWU, chociaż można stosować w takich przypadkach adresy skrócone, tj.

bez części IDP. Przyjmuje się wówczas domyślnie, że pominięta część IDP określa domenę odpowiadającą sieci LAN. Brak części IDP w adresie NSAP-u powinien być odpowiednio zaznaczony, np. przez umieszczenie cyfry zero na początku adresu (zgodnie z [1] część AFI nie może zaczynać się od cyfry zero).

Identyczne zasady przesyłania adresów NSAP-ów można zalecić także dla pakietów przesyłanych w sieci WAN. Zlecenie to może być jednak trudne do zrealizowania w praktyce. Sieci WAN są bowiem zazwyczaj sieciami już istniejącymi. Często stosuje się w nich starsze wersje protokołu X.25, w których nie używa się jeszcze pól AEF. W przypadku różnych sposobów przesyłania adresów NSAP-ów w sieciach LAN i WAN zadaniem IWU będzie dokonywanie konwersji danych adresowych w pakietach przesyłanych pomiędzy obiema sieciami.

Rozpatrzony zostanie dalej przykładowo sposób dokonywania takiej konwersji w IWU łączącym sieć LAN z istniejącymi sieciami MSK lub ASK.

W sieciach MSK ASK adresy NSAP-ów przesyłane są w polu AF pakietów ustanawiających połączenia logiczne. Adresy te powstają przez złożenie adresu węzła przełączającego, do którego podłączone jest DTE związane z danym NSAP-em oraz numeru tego DTE.

Adres węzła przełączającego składa się z od 1 do 6 cyfr i można go uważać za identyfikator domeny adresowej obejmującej wszystkie NSAP-y znajdujące się w DTE dołączonych do tego węzła. Numer DTE jest zawsze dwucyfrowy i można go uważać, za część DSP adresu NSAP-u. Część AFI nie jest przesyłana w pakietach - można domyślnie przyjmować, że jej wartość wynosi 36 (format X.121, syntetyka dziesiętna). Z punktu widzenia adresacji NSAP-ów najwygodniejszym sposobem połączenia sieci LAN z siecią MSK ASK jest potraktowanie IWU jako węzła przełączającego posiadającego swój adres, natomiast stacji sieci LAN - jako DTE podłączonych do tego węzła i posiadających dwucyfrowe numery.

Numery DTE przypisane stacjom sieci LAN powinny być równe częściom DSP adresów NSAP-ów mieszczących się w tych stacjach.

Z powyższych ustaleń wynika, że liczba NSAP-ów sieci LAN mogących komunikować się z siecią MSK ASK nie może być większa niż 99 oraz, że części DSP adresów tych NSAP-ów muszą zawsze być dwucyfrowe i wyrażone w syntaktyce dziesiętnej. Adresy innych NSAP-ów w sieci LAN mogą być tworzone bez uwzględniania wymienionych ograniczeń.

Przedstawione rozważania prowadzą do wniosku, że konwersja adresów NSAP-ów dokonywana w IWU polegać będzie na umieszczaniu odpowiednich fragmentów pola AEF pakietów odebranych z sieci LAN w polu AF pakietów wysyłanych do sieci MSK ASK lub na czynności odwrotnej, co ilustruje rys. 3.2.

Jeśli nie przewiduje się łączenia sieci LAN z innego typu sieciami WAN, wówczas w polu AEF zaniechać można zapisywania wartości AFI, a kodowanie części IDI ograniczyć do 6-ciu cyfr.

3.2. Przydział numerów kanałów logicznych

Metoda przydziału numerów kanałów logicznych powinna umożliwiać kojarzenie pakietów odbieranych w danym NSAP-ie z odpowiednimi połączeniami logicznymi wyłącznie na podstawie numerów kanałów logicznych zapisanych w tych pakietach. W sieciach LAN nie istnieje podsieć komunikacyjna zapewniająca przydzielenie danego numeru kanału logicznego tylko jednemu połączeniu dochodzącemu do danego NSAP-u. W związku z powyższym zasady przydziału numerów kanałów logicznych określone w standardzie X.25 nie mogą być w sieciach LAN stosowane.

Sformułowane na wstępie wymaganie spełnia sposób przydzielania numerów kanałów logicznych przedstawiony w [4]. Sposób ten polega na stosowaniu różnych numerów kanałów logicznych w pakietach przesyłanych z DTE Calling do DTE Called i w kierunku odwrotnym. Numery te są negocjowane podczas ustanawiania połączenia logicznego w następujący sposób.

DTE Calling w pakiecie CALL REQUEST umieszcza wartość 0 w polu określającym numer kanału logicznego. Wartość ta pełni rolę identyfikatora omawianego sposobu. W polu udogodnień pakietu CALL REQUEST umieszczona jest z kolei wartość numeru kanału logicznego, która powinna być używana w pakietach wysyłanych do DTE Calling. DTE Called wysyła w odpowiedzi pakiet CALL ACCEPTED, który w polu numeru kanału logicznego ma wpisaną wartość żadaną w pakiecie CALL REQUEST, natomiast w polu udogodnień - wartość numeru kanału logicznego, która powinna być używana w pakietach wysyłanych z DTE Calling. Ilustruje to rys.3.3.

Dodatkową zaletą przedstawionego sposobu jest to, że przydział numerów kanałów logicznych nie zależy od roli jaką przyjęło na siebie dane DTE we współpracy z odległym DTE (rola DTE lub DCE).

3.3. Rozmiar pakietu w sieci LAN

Jako normalny a zarazem domyślny rozmiar pakietu (rozumiany jako maksymalna długość pola danych w pakietach DATA) w sieci LAN proponuje się przyjąć 1024 bajty. Jest to największy rozmiar pakietu zgodny zarówno ze standardem X.25 jak również ze standardem IEEE802. Dopuszcza się jednocześnie negocjowanie rozmiaru pakietu podczas ustanowienia połączenia logicznego przez każdą stację sieci LAN - w tym IWU.

Przyjęcie wyżej podanego rozmiaru pakietu pozwala efektywniej wykorzystać możliwości sieci LAN jednakże wprowadza pewne komplikacje we współpracy sieci LAN z sieciami WAN, w których z reguły stosuje się mniejsze rozmiary pakietów - zazwyczaj 128 bajtów. Wynikają z tego określone konsekwencje odnośnie funkcjonowania IWU. Rozpatrzone zostaną dwa typy IWU:

Typ A - IWU ma możliwość segmentacji i składania pakietów,

Typ B - IWU nie ma możliwości segmentacji i składania pakietów.

IWU typu A

Nie jest konieczne negocjowanie rozmiaru pakietu w sieci LAN. Przyjmuje się domyślnie wartość 1024.

Przesyłaniu pakietów DATA przez IWU towarzyszą procesy segmentacji (dla kierunku z LAN do WAN) i składania (dla kierunku z WAN do LAN). W pakietach tworzonych w wyniku segmentacji odpowiednio ustawione muszą być bity M i D tak, aby możliwe było odtworzenie z tych pakietów jednostki danych usługowych NSDU (pakietu pierwotnego) oraz, aby prawidłowo funkcjonował mechanizm potwierdzeń typu end-to-end. Konsekwencją różnych rozmiarów pakietów na stykach LAN IWU oraz IWU WAN jest różna numeracja pakietów DATA na tych stykach. Wobec powyższego IWU dokonywać musi konwersji parametrów $P(R)$ i $P(S)$ w pakietach DATA, RR, RNR i REJ przesyłanych pomiędzy sieciami LAN i WAN. Przykład posługiwania się bitami M i D oraz określania wartości parametrów $P(R)$ i $P(S)$ w IWU przedstawiono na rys. 3.4. Rysunek ten ilustruje przesyłanie pakietów DATA zarówno z sieci LAN do WAN jak i w kierunku odwrotnym. Zgodnie z rysunkiem, pola danych pakietów DATA wysyłanych z LAN do IWU (kolumna 1) dzielone są na cztery segmenty. Każdy z tych segmentów wysyłany jest do sieci WAN w postaci pojedynczego pakietu DATA (kolumna 2). Pakiety te z wyjątkiem ostatniego mają bit $M=1$. Bit D w pakiecie przesyłającym ostatni segment ma wartość taką samą jak w pakiecie pierwotnym. W pozostałych pakietach bit ten ma wartość zero. Sekwencja pakietów DATA wysyłanych z IWU do WAN odpowiada sekwencji pakietów RR odbieranych z sieci WAN (kolumna 3) potwierdzających odbiór pakietów DATA.

Spśród wspomnianych pakietów RR tylko te, które potwierdzają odbiór ostatniego segmentu pakietu pierwotnego przesyłane są dalej do sieci LAN ze zmienioną wartością parametru $P(R)$ (kolumna 4). W analogiczny sposób rozpatrzeć można przesyłanie pakietów DATA w kierunku odwrotnym. W takim przypadku IWU musi być zdolne do generowania pakietów RR potwierdzających odbiór z sieci WAN każdego pakietu DATA z bitem $D=0$. Pakiety RR potwierdzające odbiór z sieci WAN

pakietów DATA z bitem D=1 są tworzone w IWU z pakietów RR odebranych z sieci LAN.

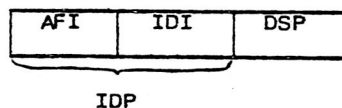
Możliwość segmentacji i składania pakietów w IWU nie będzie wykorzystana, jeżeli maksymalny rozbiór jednostki danych transportowych TPDU będzie równy rozmiarowi pakietu w sieci WAN. Biorąc pod uwagę, że podstawowym rozmiarem TPDU w sieciach WAN jest 128 bajtów stwierdzić można, że stosowanie IWU typu A jest celowe wtedy, gdy umożliwi ono komunikowanie się sieci LAN z inną siecią LAN poprzez sieć WAN (przy założeniu, że w obu sieciach LAN stosuje się duże rozmiary TPDU).

IWU typu B

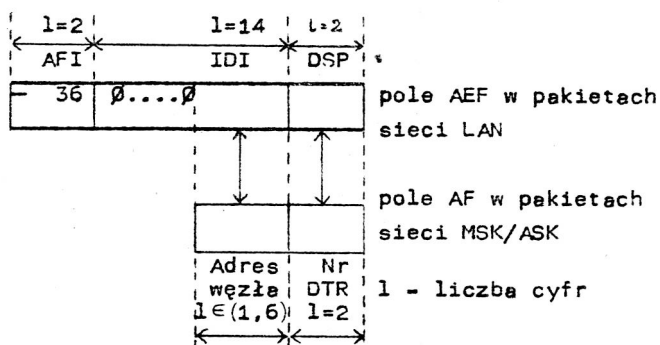
Podczas ustanawiania połączeń logicznych IWU powinno dążyć do wynegocjowania w sieci LAN takiego rozmiaru pakietu jaki jest stosowany w sieci WAN.

Jeżeli połączenie logiczne ustanawiane jest z inicjatywy stacji LAN, wówczas pakiet CALL REQUEST nie musi zawierać jawnego żądania rozmiaru, co oznacza propozycję rozmiaru 1024 bajty. IWU powinno wskazać w pakiecie CALL CONNECTED wysykanym do LAN rozmiar taki jaki przyjęto (lub wynegocjowano) na styku IWU WAN. Podczas ustanowienia połączenia z inicjatywy sieci WAN, IWU powinno wskazać w pakiecie INCOMING CALL przesykanym do LAN rozmiar pakietu żądany przez sieć WAN. W obu przypadkach rozmiary pakietów wskazane przez IWU powinny być obowiązujące dla stacji LAN.

Pakiety DATA a także RR, RNR, REJ przesyłane będą przez IWU prawidłowo transparentnie, tzn. zmieniane będą jedynie numery kanałów logicznych.

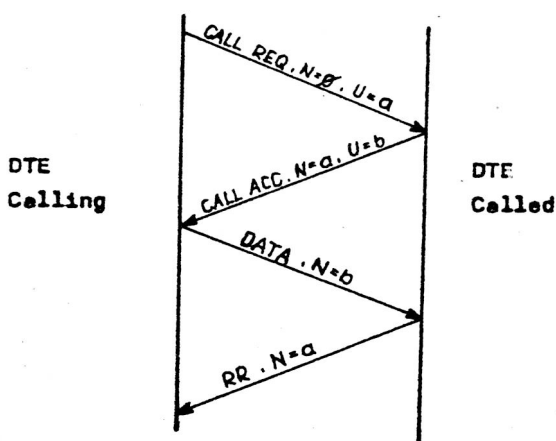


Rys. 3.1. Postać adresu NSAP-u
Fig. 3.1. NSAP address format



Rys.3.2. Zasady określania zawartości pól adresowych pakietów w IWU łączącym sieć LAN z sieciami MSK lub ASK

Fig. 3.2. Address fields in IWU packets



Rys. 3.3. Ilustracja sposobu przydzielania numerów kanałów logicznych; N-pole numeru kanału logicznego, U-pole udogodnień

Fig. 3.3. Assignment of logical channel numbers

1	2	3	4
Pakiety DATA		Pakiety RR	
IWU LAN lub LAN IWU	WAN IWU lub IWU WAN	IWU WAN lub WAN IWU	LAN IWU lub IWU LAN
M=0,D=1 P/s/=0	M=1,D=0 P/s/=0	P/R/=1	
	M=1,D=0 P/s/=1	P/R/=2	
	M=1,D=0 P/s/=2	P/R/=3	
	M=0,D=1 P/s/=3	P/R/=4	P/R/=1
M=0,D=0 P/s/=1	M=1,D=0 P/s/=4	P/R/=5	
	M=1,D=0 P/s/=5	P/R/=6	
	M=1,D=0 P/s/=6	P/R/=7	
	M=0,D=0 P/s/=7	P/R/=0	P/R/=2

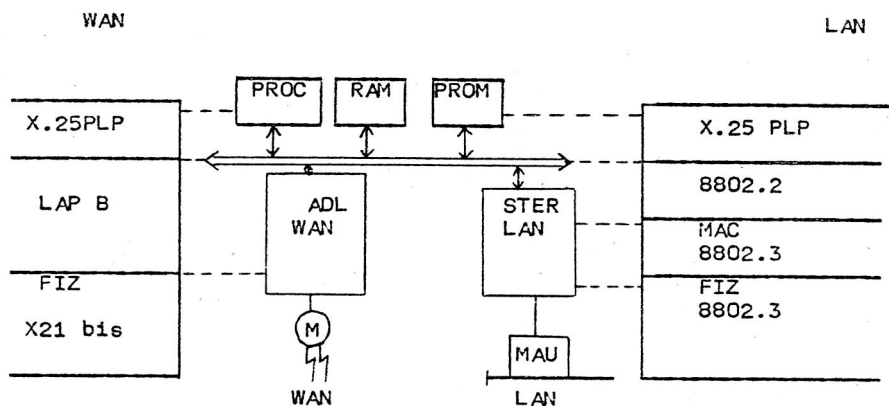
Rys.3.4. Zasady przetwarzania pakietów DATA i RR w IWU typu A
Fig. 3.4. Converting the DATA and RR packets into IWU packets

4. ARCHITEKTURA SYSTEMU SPRZĘGAJĄCEGO SIECI LOKALNE Z SIECIĄ ROZLEGŁĄ

Zgodnie z zasadami współpracy lokalnych i rozległych sieci komputerowych przedstawionymi w pktcie 2, obejmującymi zgodność protokołów komunikacyjnych, ich parametrów i opcji w obydwóch typach sieci, od poziomu trzeciego w górę, architektura systemu sprzęgającego zapewnia możliwości "dopasowania" dolnych warstw o różnych protokołach.

Warstwy fizyczne i łącza danych, a w zasadzie podwarstwa dostępu, są zorientowane na medium transmisyjne i charakterystyczne dla różnych typów sieci.

System sprzęgający IWU ma zapewnić przeźroczystość sprzętową jednego typu sieci dla drugiego typu sieci. Architektura IWU spełniającego w/w wymagania jest przedstawiona na rys. 4.1



Rys.4.1. Architektura systemu sprzęgającego LAN/WAN

Fig. 4.1. Architecture of LAN/WAN interfacing system

5. WNIOSKI KOŃCOWE

Przedstawione w pracy koncepcje rozwiązania systemu sprzęgającego lokalne sieci komputerowe i sieci rozległe odpowiada światowym trendom w tym zakresie, ponieważ zakłada się zgodność protokołów warstw wyższych - powyżej trzeciej, w obydwóch typach sieci. Implikuje to konieczność dopasowania jedynie protokołów warstw dolnych zorientowanych na medium transmisyjne.

Tego typu koncepcja pozwala konstruować systemy sprzęgające jako urządzenia niezbyt skomplikowane i wprowadzające nieznaczne opóźnienia przy wymianie informacji pomiędzy sieciami LAN i WAN.

Przedstawione powyżej rozwiązanie będzie wykorzystane w łączeniu sieci lokalnych LAN KASK z siecią WAN KASK.

6. LITERATURA

- [1] ISO DIS 8348 DAD - Network service definition, addendum 2: Network layer addressing, ISO TC 97, 1986.
- [2] ISO DIS 8878 - Use of X.25 to provide the OSI connection - mode network service, ISO TC 97, 1986.
- [3] ISO DP - Use of the X.25 packet level protocol in local area networks, ISO TC 97, 1985..
- [4] ISO DIS 8208 DAD i - X.25 packet level protocol for data terminal equipment, Addendum 1: Alternative logical channel number allocation.

INTERWORKING LAN AND WAN - PRINCIPLES AND IMPLEMENTATIONS

In the paper, the LAN and WAN interworking principles based on international standards, are presented. A proposal of gateway construction according these principles is described. In order to obtain "compatibilities" of the LAN and WAN protocols, the mean of parameters selection and options of the LAN protocols are presented: in detail the way of addressing NSAP, the way of the numbers of logical channels allocation and the selection of packets size in the LAN is discussed. The presented solution will be applied within LAN KASK and WAN KASK interworking.

СОТРУДНИЧЕСТВО ЛОКАЛЬНЫХ И ШИРОКИХ КОМПЬЮТЕРНЫХ СЕТЕЙ - ПРИНЦИПЫ И СПОСОБЫ ИМПЛЕМЕНТАЦИИ

В работе представлены принципы сотрудничества локальных и широких компьютерных сетей на базе проводимых в мире стандартизованных работ. Предложено решение сцепляющей системы по этим принципам. Для "подбора" протоколов коммуникационных локальных и широких сетей предложен способ отбора параметров и опций коммуникационных протоколов в локальных сетях, особенно способ адресирования NSAP в сети LAN, придача номеров логических каналов и отбор размера пакета в LAN. Представлено решение используют для соединения сетей LAN/KASK и WAN/KASK.

Zbigniew HUZAR ^x
Jan KWIATKOWSKI ^{'x}

sieć komputerowa,
usługi sieciowe,
transfer zadań

ROZPROSZONE PRZETWARZANIE ZADAŃ W SIECI KOMPUTEROWEJ

Jedną z usług świadczonych przez sieć komputerową jest usługa transferu i manipulacji zadaniami (JTM) objęta standaryzacją przez Międzynarodową Organizację ds. Standardów (ISO). W pracy przedstawiono aktualny stan prac standaryzacyjnych nad wymienioną usługą. W pierwszej części pracy podano ogólną strukturę oraz zasady funkcjonowania systemu transferu i manipulacji zadaniami. Następnie określono zakres oraz przeprowadzono klasyfikację usług JTM, koncentrując się na opisie tzw. klasy podstawowej, będącej przedmiotem oddzielnego standardu.

1. WSTĘP

Sieć komputerowa łącząc odległe systemy komputerowe, stanowi tym samym naturalne środowisko rozproszonego przetwarzania zadań. Pojęcie rozproszenia może tu odnosić się do podziału zadania na fragmenty przeznaczone do współbieżnego wykonania w różnych systemach komputerowych, a także może oznaczać, że dane potrzebne do jego realizacji są rozmieszczone w pamięciach różnych systemów oraz że wyniki obliczeń zadania mają być rozesłane do różnych systemów. W rozwoju architektur sieci komputerowych, w tym także w architekturze systemów otwartych ISO, zestaw mechanizmów umożliwiających użytkownikowi rozproszone przetwarzanie zadań został zgrupowany w tzw. usługi transferu i manipulacji zadaniami (ang. - Job Transfer and Manipulation Services - JTM).

Celem artykułu jest przedstawienie zakresu usług JTM oraz przegląd najistotniejszych problemów funkcjonowania systemu rozproszonego przetwarzania zadań. Prace nad zdefiniowaniem JTM są intensywnie prowadzone

^x Politechnika Wrocławska, Centrum Obliczeniowe, Wrocław

przez ISO. Aktualny stan tych prac, chociaż znacznie zaawansowany [12,13], jest jeszcze odległy od pełnego zakończenia. Należy podkreślić, że koncepcja i usługi JTM były wcześniej opracowywane przez pojedyncze kraje zaawansowane w rozwijaniu i stosowaniu sieci komputerowych. Wiele z tych koncepcji, zwłaszcza proponowanych w brytyjskim protokole NIJTM [8,9], który był także podstawą projektu standardu dla sieci MSK [1,2], zostało przejęte przez rozwiązania ISO.

Usługi JTM są pośrednikiem pomiędzy użytkownikiem, który formułuje zadanie sieciowe, a zespołem realizatorów, który może uczestniczyć w realizacji tego zadania. Usługi JTM są ulokowane w warstwie aplikacji modelu referencyjnego OSI [18]; należą one do usług świadczonych przez jednostki tzw. specjalizowanych usług aplikacyjnych (Specialized Application Service Entities - SASE) [16]. Użytkownikiem usług JTM może być użytkownik fizyczny lub program, zaś realizatorem zadań sieciowych systemy operacyjne lokalnych systemów komputerowych. Zarówno użytkownik jak i realizatorzy przetwarzania zadania sieciowego są abstrakcyjnie określani jako tzw. agencje. Standaryzowany zakres usług JTM obejmuje właśnie kontakt systemu JTM z mającymi do niego dostęp agencjami.

Usługi, których system JTM dostarcza agencjom zależą od roli jaką odgrywają w procesie przetwarzania zadania. Dlatego wyróżnia się agencje inicjujące - reprezentujące użytkownika formułującego zadanie sieciowe; agencje źródłowe - reprezentujące te systemy komputerowe bądź ich elementy, które dostarczają dane do zadania; agencje przetwarzające - reprezentujące systemy wykonujące obliczenia na dostarczonych danych; oraz agencje docelowe - reprezentujące te systemy, dla których są przeznaczone wyniki obliczeń zadania. Oczywiście pewien element lokalnego systemu operacyjnego, np. system zbiorów, będzie mógł pełnić rolę różnych agencji, np. agencji źródłowej lub docelowej, w zależności od tego jaką rolę spełnia w przetwarzaniu danego zadania. Agencjami mogą być także aplikacje sieciowe np.: stacja transferu zbiorów może pełnić rolę agencji źródłowej bądź docelowej.

Poziom usług, które system JTM dostarcza, wynika z założenia, że użytkownik:

- wskazuje te systemy komputerowe, które mają uczestniczyć w procesie realizacji zadania i określa porządek oraz sposób ich wykorzy-

stania,

- posiada odpowiednią znajomość własności i funkcji tych systemów, a w szczególności ich języki opisu sterowania zadaniami (job control and response languages).

Obecnie rozróżnia się dwie klasy usług tzw. podstawową (basic class) oraz pełną (full class). Klasa podstawowa jest przedmiotem oddzielnej standaryzacji [12].

2. OGÓLNA STRUKTURA SYSTEMU JTM

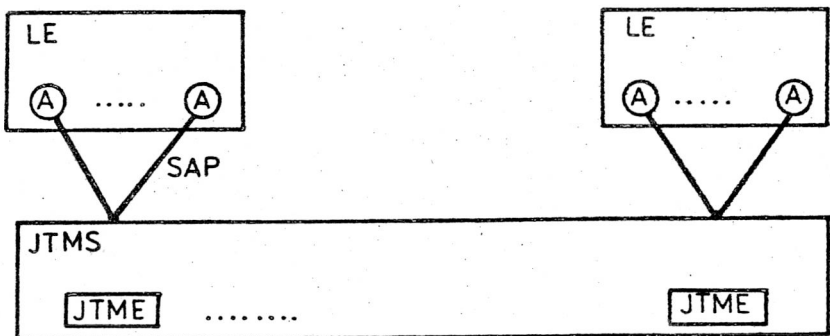
System JTM składa się ze współpracujących ze sobą stacji JTM, posadowionych w oddzielnych systemach komputerowych (rys. 1), których współpraca polega na wzajemnym przekazywaniu do realizacji tzw. opisów zadań. Usługi, których dostarcza system JTM, za pośrednictwem tzw. prymitywów (poleceń) usług, są dostępne agencjom funkcjonującym w tych systemach. Natomiast usługi, z których korzysta, to usługi: systemu diagnostyki i odnowy informacyjnej (CCR - Commitment, Concurrency and Recovery), systemu zarządzania asocjacjami (KCAS - Kernel Common Application Service) oraz usługi warstwy prezentacji. Stacja JTM wykorzystuje wymienione usługi w różnych fazach swojej działalności.

Wykorzystanie usług KCAS służy zakładaniu i kasowaniu asocjacji pomiędzy współpracującymi stacjami JTM. Pojęcie asocjacji stanowi uogólnienie pojęcia sesji. Ustanowienie asocjacji polega, w uproszczeniu, na wynegocjowaniu parametrów połączeń sesyjnych oraz kontekstów prezentacyjnych. Raz ustanowiona asocjacja pozwala na wznawianie kolejnych sesji z powołaniem się na parametry wynegocjowane dla pierwszego z połączeń sesyjnych. Własności ustanowionych asocjacji dotyczą m.in.:

- własności wymaganych przez CCR;
- ustalenia kontekstów prezentacyjnych tzn. ustalenia abstrakcyjnych syntaktyk dla obiektów stanowiących elementy transferu pomiędzy stacjami JTM (abstrakcyjne syntaktyki dla opisów zadań oraz dołączonych do nich dokumentów) oraz obiektów stanowiących elementy CCR;
- adresacji obiektów, które wykorzystuje asocjacja;
- jakości połączeń (stopy niewykrywalnych błędów transmisji, częstotliwości zrywania usług).

Wykorzystanie usług CCR ma służyć zapewnieniu niezawodnego transferu, w tym odpowiedzialności za przesyłane zadanie (w danym momencie czasu za każde zadanie sieciowe ponosi odpowiedzialność dokładnie jeden system komputerowy). Usługi CCR dostarczają też zwrotnej informacji diagnostycznej o niepowodzeniach w kontaktach pomiędzy współpracującymi obiektami. Usługi CCR są, zgodnie ze standardem, wbudowane w prymitywy usług wymieniane pomiędzy stacją JTM a skojarzonymi z nią agencjami oraz w komendzie transferu opisu zadań pomiędzy stacjami JTM.

Współpraca stacji JTM z warstwą prezentacji polega tylko na wykorzystaniu usługi P-DATA, czyli na przesyłaniu elementów transferu w odpowiednim, wcześniej uzgodnionym kontekście prezentacji, poprzez ustanowioną uprzednio asocjację pomiędzy komunikującymi się stacjami JTM. Podstawowa klasa usług JTM wykorzystuje tylko pojedyncze jednokierunkowe prymitywy P-DATA.



LE - lokalne środowisko systemowe (Local System Environment)

JTMS - system usług JTM (JTM System)

A - agencja (Agency)

JTME - stacja JTM (JTM Entity)

SAP - punkt dostępu do usług (Service Access Point)

Rys. 1. Ogólna struktura systemu usług JTM

Fig. 1. The global structure of JTM service

3. ZASADY FUNKCJONOWANIA SYSTEMU JTM

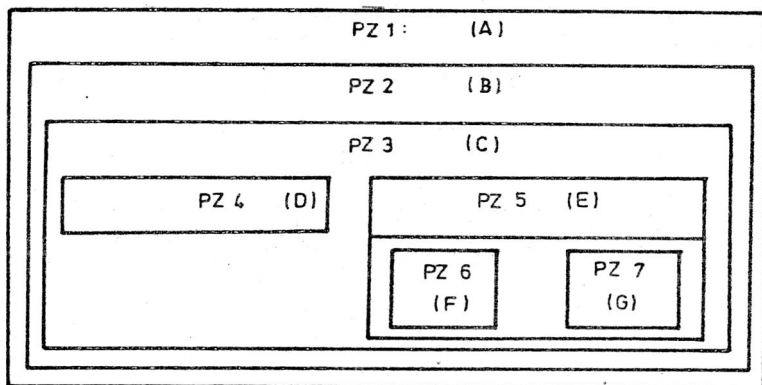
Stacje JTM prowadząc ze sobą współpracę wymieniają opisy przekazywanych sobie do realizacji zadań. Opis zadania składa się zasadniczo z trzech części. Pierwsza część zawiera informacje sterujące, dotyczące całego zadania oraz tego jego fragmentu (podzadania), które ma być wykonane w systemie, do którego jest przesyłane. Druga - zawiera zbiór dokumentów (danych) związanych z wykonywanym zadaniem. Trzecia część zawiera również informację sterującą dotyczącą jednakże tych fragmentów zadania, które mają być wykonywane w dalszej kolejności - są to tzw. szkielety opisów nowych podzadań.

Ogólnie opis zadania (work specification) zawiera informacje potrzebne do:

- identyfikacji zadania i określenia miejsca oraz czasu jego powstania,
- autoryzacji zadania,
- określenia miejsc przesyłania i zawartości raportów z postępów realizacji zadania,
- określenia typu zadania,
- pilności realizacji,
- identyfikacji systemów, które mają uczestniczyć w przetwarzaniu zadania,
- uwarunkowania postępu zadania od zajścia odpowiednich akcji lub warunków,
- określania akcji, które ma podejmować system JTM,
- określania dalszych podzadań danego zadania.

Pierwotny opis zadania sieciowego powstaje w stacji JTM na podstawie informacji dostarczonych przez odpowiednią agencję inicjującą. Sporządzony opis zadania sieciowego stanowi w ogólnym przypadku pewną strukturę drzewiastą odpowiednio zagnieżdżonych w sobie podzadań. Dla ilustracji rozpatrzmy następujące przykładowe zadanie wygenerowane w systemie A:

Należy pobrać odpowiednie dokumenty z systemu B, przesłać je do C w celu ich przetworzenia, a następnie jedną część wyników przesłać do D a drugą do E w celu dalszego przetwarzania, którego wyniki przesłać do F oraz G. Struktura opisu tego zadania ma postać przedstawioną na rys.2.



Rys. 2. Struktura przykładowego zadania.

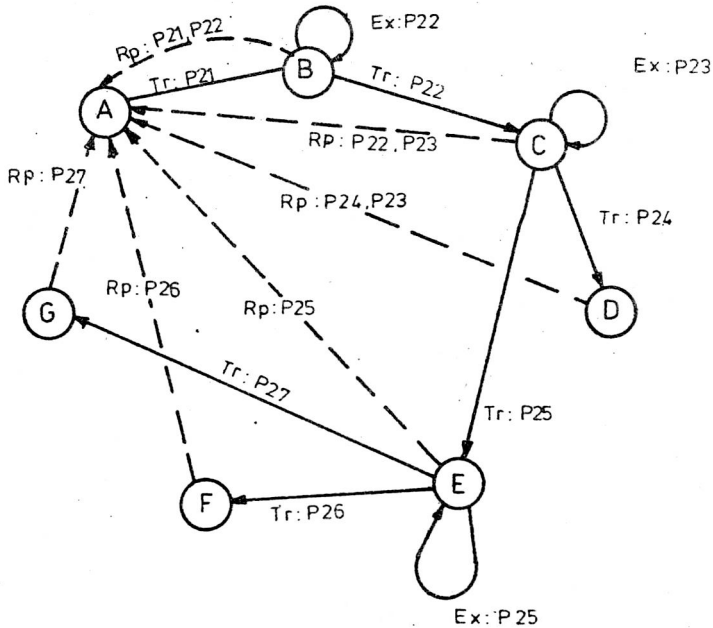
Fig. 2. Example structure of a job

Znaczenie poszczególnych podzadań jest następujące:

- PZ 1: prześlij pierwotny opis z A do B
- PZ 2: pobierz dokumenty z B, dołącz do opisu i prześlij do C
- PZ 3: przetwórz dokumenty w C, część wyników prześlij do D (w postaci pewnego opisu) oraz część do E (w postaci drugiego niezależnego opisu)
- PZ 4: zapamiętaj otrzymane dokumenty w D
- PZ 5: przetwórz otrzymane dokumenty w E, część wyników prześlij do F, a część do G
- PZ 6: zapamiętaj otrzymane dokumenty w F
- PZ 7: zapamiętaj otrzymane dokumenty w G.

Należy zwrócić uwagę, że w opisie zadania sieciowego wraz z opisem konkretnego podzadania, które ma być wykonane w pierwszej kolejności są zawarte w postaci szkieletów opisy dalszych podzadań. Podzadanie PZ 1 po zrealizowaniu, tzn. po przesłaniu całego opisu zadania sieciowego z systemu A do B i po przejęciu przez B odpowiedzialności za dalszą jego realizację, przekształca się w nowe podzadanie PZ 2 itd. Należy zauważyć, że po zakończeniu podzadań PZ 3 pozostają dwa niezależne opisy, które są dalej wykonywane niezależnie od siebie. Ponieważ są one fragmentem tego samego zadania ich transfery nazywa się transferami stowarzyszonymi.

Zadanie sieciowe zostanie zakończone, gdy zakończone zostaną wszystkie wygenerowane przez nie podzadania.



- A, ..., G - systemy (systems)
 Tr - transfer zadania (job transfer)
 Ex - wykonanie zadania (job execution)
 Rp - raport z wykonania zadania (report)

Rys. 3. Schemat akcji związanych z realizacją przykładowego zadania sieciowego

Fig. 3. Example of a job execution.

Zadanie w trakcie swej realizacji, podobnie jak w przypadku przetwarzania zadań w klasycznym systemie wielodostępnym, może generować sprawozdanie o swym postępie. Raportowane mogą być zdarzenia o dokonanych transferach zadania pomiędzy systemami, skompletowaniu danych potrzebnych do przetwarzania, wygenerowaniu nowych podzadań, błędach itp. Wszystkie raporty przyjmują postać podobną do opisu zadania - raporty są traktowane jako zadania, które nie wymagają żadnego przetwarzania, a

tylko przesłania we wskazane miejsce lub miejsca sieci. Wyróżnia się dwa rodzaje raportów: jeden to ten, który definiuje agencja inicjująca, a drugi to ten, który niezależnie może wprowadzić dla własnych potrzeb stacja JTM przyjmująca zadanie do wykonania. Rys. 3 obrazuje sytuację nawiązującą do omawianego przykładu i pokazuje możliwe drogi przesyłania opisów zadań i raportów z jego realizacji pomiędzy systemami uczestniczącymi w realizacji zadania. Poziom raportowania określający raportowane zdarzenia jest definiowany przez użytkownika i może być przez niego zmieniany w zależności od postępów w realizacji zadania sieciowego. Podstawowa klasa usług dopuszcza tylko pierwszy z wymienionych poziomów raportowania.

Również, podobnie jak w przypadku klasycznych systemów wielodostępnych, użytkownik może z własnej inicjatywy kierować zapytania o stan realizacji swego zadania, bądź dokonywać pewnych jego modyfikacji. Takie zapytania lub zadania zmian dotyczące uprzednio zdefiniowanych zadań można formułować w postaci opisu zadań sterujących. Są one wprowadzane do systemu JTM, tak jak poprzednio omawiane zadania, a ich wykonanie polega na dotarciu do tych systemów komputerowych, które aktualnie zawierają zadania mające podlegać modyfikacji, dokonaniu tych modyfikacji i wygenerowaniu raportów z ich realizacji.

W trakcie przetwarzania zadania mogą powstawać różnego rodzaju błędy. Pierwsze mogą zostać wykryte w momencie przyjmowania zadania zgłaszanego przez agencję inicjującą. Wykrycie takich błędów przez stację JTM, przed przyjęciem przez nią odpowiedzialności za wykonanie zadania, powoduje odpowiednią sygnalizację i odnowę akceptacji zadania. Następne błędy mogą powstać w trakcie przetwarzania, po przyjęciu przez stację JTM odpowiedzialności za jego realizację. Postępowanie w tych przypadkach może być różne. W obrębie podstawowej klasy usług JTM przyjmuje się następujące zasady.

Brak wskazanego dokumentu w agencji źródłowej powoduje wygenerowanie odpowiedniego raportu, zaś na miejscu spodziewanego dokumentu w opisie zadania umieszcza się odpowiednią informację diagnostyczną, po czym przetwarzanie zadania kontynuuje się. Agencje, do których zamiast dokumentu trafi informacja diagnostyczna same podejmują decyzje o dalszym postępowaniu.

Błąd podczas przesyłania dokumentów do agencji docelowych lub wykonawczych, bądź błąd transferu opisu zadania powodują zawieszenie wykonywania zadania i wygenerowanie odpowiedniego raportu. Zawieszenie zadania trwa pewien odcinek czasu, w którym użytkownik, po zapoznaniu się z raportami o przebiegu realizacji zadania może, poprzez wygenerowanie zadania manipulującego zawieszonym zadaniem dokonać usunięcia przyczyny błędu. W takim przypadku zadanie się odwołuje i kontynuuje swą realizację. Jeżeli natomiast w przeciągu tego odcinka czasu nie zostanie usunięta przyczyna błędu, zadanie zostaje zakończone, czemu towarzyszy generacja raportu o awaryjnym jego zakończeniu.

Stacje JTM muszą dysponować pewną informacją o stanie innych stacji JTM, z którymi mają współpracować. Stacja podejmując decyzję transferu opisu zadania do innej stacji potrzebuje informacji czy stacja ta jest aktualnie aktywna, czy nieaktywna; w przypadku aktywności potrzebna jest znajomość liczby i rodzajów opisów zadań, które stacja może przyjąć; w przypadku nieaktywności potrzebna jest informacja czy jest to nieaktywność chwilowa czy trwała itp. Tego rodzaju informacje są zgrupowane w tzw. rekordach sterowania transferami (TCR) i mogą one, w całości lub fragmentami być udostępniane innym stacjom poprzez wykonanie odpowiednich zadań typu manipulacyjnego.

Reasumując stacje JTM mogą wymieniać pomiędzy sobą następujące typy opisów zadania:

- przesyłające raporty,
- przesyłające dokumenty,
- manipulujące na opisach,
- manipulujące na rekordach TCR,
- manipulujące na raportach.

4. SPECYFIKACJA USŁUG JTM

Usługi świadczone przez JTM są definiowane na bazie uprzednio określonych elementów modelu usług JTM. Prymitywy tych usług reprezentują wzajemne interakcje pomiędzy stacją JTM (JTM service provider) a różnymi agencjami modelu reprezentującymi użytkowników usługi. Prymitywy te umożliwiają:

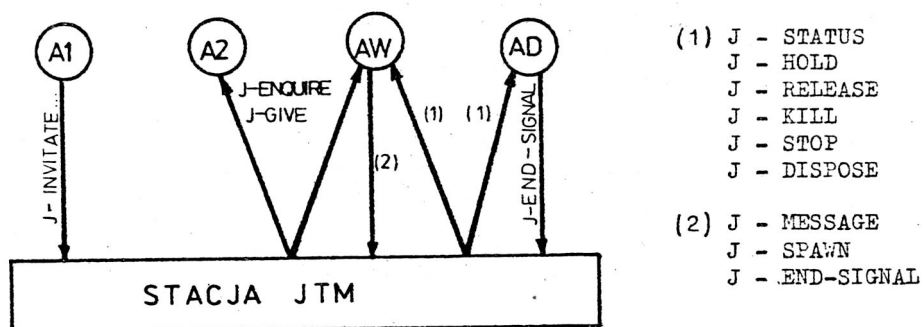
- dostarczanie zadań do stacji JTM przez agencję inicjującą,

- dostarczanie dokumentów do agencji docelowych,
- odbieranie dokumentów z agencji źródłowych,
- dostarczanie dokumentów do agencji wykonawczych w celu ich przetwarzania.

Nawiązując do specyficznych mechanizmów JTM, takich jak kreowanie nowych opisów zadań ze szkieletu, żądanie usługi przez jednego z użytkowników może spowodować cały ciąg interakcji pomiędzy stacją JTM a różnymi agencjami tworząc ciąg akcji pomiędzy użytkownikami usługi JTM w celu realizacji zleconego zadania. Każda z takich interakcji pomiędzy stacją JTM a agencją może być określona poprzez sekwencję przekazywanych prymitywów. Prymitywy wykorzystywane przez usługę JTM mogą mieć bezpośrednie odwzorowanie w prymitywach usług CCR bądź realizować funkcje własne usługi JTM. Wszystkie grupy prymitywów usług JTM mogą być podzielone na cztery klasy realizujące odpowiednio:

- usługę kreowania opisu zadania,
- usługę wymiany dokumentów,
- usługi wymagane dla realizacji zadania,
- usługi manipulacji na istniejących zadaniach.

Wzajemne powiązania pomiędzy poszczególnymi agencjami modelu a stacją JTM wykorzystujące wspomniane klasy prymitywów ilustruje rys. 4.



- AI - agencja inicjująca (initiation agency)
 AW - agencja wykonawcza (execution agency)
 AD - agencja docelowa (destination agency)
 AZ - agencja źródłowa (source agency)

Rys. 4. Prymitywy stosowane przez agencje oraz stację JTM

Fig. 4. JTM service primitives

4.1. Klasy prymitywów usług

Pierwszą klasę prymitywów stanowią prymitywy wykorzystywane przez agencję inicjującą i służą do tworzenia nowych opisów zadań (work specification). Każdy z nich może wystąpić w dwóch postaciach; żądanie utworzenia oraz potwierdzenie utworzenia opisu zadania, przy czym parametry pierwszego z nich praktycznie pokrywają się z parametrami zawartymi w opisie zadania, natomiast w odpowiedzi generowany jest unikalny identyfikator opisu w ramach systemu otwartego. Na powyższą klasę składają się następujące grupy prymitywów:

- J - INITIATE-WORK - stosowany do utworzenia opisu zadania dla przesyłania dokumentów,
- J - INITIATE-WORK-MAN - stosowany do utworzenia opisu zadania dla manipulacji istniejącym opisem,
- J - INITIATE-REPORT-MAN - stosowany do utworzenia opisu zadania dla manipulacji na raportach otrzymanych przez monitor zadań sieciowych,
- J - INITIATE-TCR-MAN - stosowany do utworzenia opisu zadania dla manipulacji rekordami kontrolnymi transferu (TCR).

Druga z klas prymitywów odpowiedzialna jest za usługi związane z przesyłaniem dokumentów i zawiera następujące grupy prymitywów:

- J - DISPOSE - stosowany do przesyłania dokumentów do agencji wykonawczej lub docelowej; tworzona jest nowa aktywność w ramach agencji,
- J - GIVE - stosowany jako żądanie przesłania dokumentu z agencji źródłowej bądź wykonawczej,
- J - ENQUIRE - stosowany w celu otrzymania z agencji źródłowej lub wykonawczej listy nazw dokumentów spełniających podane warunki.

Prymitywy tej klasy są wykorzystywane przez stację JTM jako bezpośredni lub pośredni rezultat jednego z prymitywów klasy J-INITIATE wydanego przez agencję inicjującą. Podobnie jak prymitywy poprzedniej klasy występują w dwóch postaciach (żądanie, odpowiedź), przy czym w pierwszej z postaci parametry prymitywu określają wskazany dokument (grupa dokumentów dla J-ENQUIRE), natomiast parametrami odpowiedzi są odpowiednio:

- informacje identyfikujące aktywność agencji - dla J-DISPOSE,
- wskazany dokument - dla J-GIVE,
- lista nazw dokumentów spełniających podane warunki - dla J-ENQUIRE.

Dla tej klasy prymitywów stacja JTM i wykorzystywane agencje nie muszą należeć do tego samego systemu otwartego.

Trzecia z klas prymitywów jest związana bezpośrednio z występującymi aktywnościami w ramach agencji wykonawczej oraz docelowej (tylko ostatni z prymitywów) i obejmuje następujące grupy prymitywów:

- J - MESSAGE - stosowana w celu wymuszenia na stacji JTM opisu zadania przenoszącego raporty,
- J - SPAWN - stosowana jako żądanie generacji nowego opisu zadania ze szkieletu zawartego w przetwarzanym zadaniu,
- J - END-SIGNAL - stosowana do wskazania zakończenia aktywności w ramach agencji, powstałej w wyniku wcześniejszego potwierdzenia.

Jako parametry prymitywów stosowane są identyfikatory aktywności stacji JTM, dokumenty oraz identyfikator szkieletu.

Czwartą klasę prymitywów stanowią prymitywy wykorzystywane przez stację JTM kierowane do agencji wykonawczej oraz docelowej w celu zapytania o status bądź wykonania wskazanej przez nazwę prymitywu akcji na jednej z bieżących aktywności danej agencji. Jako parametr każdego z prymitywów występuje tylko identyfikator aktywności w ramach agencji. W odpowiedzi na prymityw J-STATUS przekazywany jest komunikat określający stan wskazywanej aktywności. Do wymienionej klasy należą następujące prymitywy:

- J - STATUS - stosowany w celu otrzymania informacji o stopniu zaawansowania aktywności,
- J - HOLD - stosowany w celu zawieszenia możliwości wykonania na wskazanej aktywności operacji definiowanych przez prymitywy klasy trzeciej,
- J - RELEASE - stosowany w celu skasowania zawieszenia otrzymanego w wyniku zastosowania J-HOLD,
- J - KILL - stosowany w celu wymuszenia zakończenia wskazanej aktywności, poprzez wygenerowanie J-END-SIGNAL,

J - STOP - efekt działania prymitywu jak dla J-KILL, z tym że w wyniku późniejszego zastosowania J-GIVE lub J-ENQUIRE dokumenty wygenerowane dotychczas są dostępne (przy J-KILL dokumenty nie są dostępne).

Oprócz wymienionych powyżej klas prymitywów stosowane są również prymitywy przenoszące tylko parametry CCR (nazwy oraz przeznaczenie tych prymitywów jest identyczne jak w przypadku usług CCR [16,17]).

Implementacja usług JTM nie musi zawierać wszystkich z wymienionych w opracowaniu agencji, wtedy prymitywy określonej grupy nie występują w danym systemie otwartym. Dodatkowym ograniczeniem nakładanym przez standard jest wymagana kolejność występowania poszczególnych prymitywów usług.

5. KLASYFIKACJA USŁUG

Jak wspomniano standard wyróżnia dwie klasy implementacji "pełną" oraz "podstawową". Klasa podstawowa zapewnia jedynie pewien podzbiór oferowanych usług JTM, jednakże konstrukcja protokołów zapewnia, że wszystkie opisy zadań wygenerowane przez implementację podstawową mogą być przesyłane i przetwarzane przez implementację pełną. Relacja przeciwna jest również możliwa, z tym, że występują ograniczenia związane z parametrami przenoszonymi przez opis zadania.

W szczególności w przypadku klasy podstawowej nie wykorzystywane są następujące grupy prymitywów:

- J-INITIATE-TCR-MAN,
- J-INITIATE-REPORT-MAN,
- J-ENQUIRE,
- J-HOLD,
- J-RELEASE,
- J-SPAWN.

Powoduje to naturalne ograniczenia w zakresie oferowanych usług w szczególności związane z kreowaniem nowych opisów zadań ze szkieletu opisu.

Podobnie w przypadku klasy podstawowej generowane są tylko pewne raporty określające zdarzenia zaistniałe w wyniku przetwarzania zadania. Generowane raporty obejmują następujące zdarzenia:

- normalne zakończenie,
- błędne zakończenie,
- zakończenie manipulacji,
- komunikaty generowane w wyniku zastosowania prymitywu J-MESSAGE.

Pozostałe typy raportów nie są akceptowane przez monitor klasy podstawowej i żądanie o nie jest odrzucane. Dodatkowym ograniczeniem związanym z przyjmowaniem raportów jest możliwość zdefiniowania tylko jednego miejsca monitorowania raportów, jak również nie jest możliwe manipulowanie raportami.

Opis zadania jest generowany przez klasę podstawową jako rezultat transferu lub w przypadku gdy istnieje agencja inicjująca w wyniku zastosowania jednego z prymitywów klasy J-INITIATE. Podstawowe ograniczenie polega na tym, że dopuszcza się w opisie zadania tylko jeden dokument oraz jeden szkielet opisu najwyższego poziomu. Wynika stąd, że zadanie sieciowe może być tylko prostą sekwencją kolejnych opisów zadań i nie jest możliwe występowanie transferów stowarzyszonych.

Pozostałe ograniczenia nałożone przez standard na klasę podstawową powodują, że:

- wszystkie błędy poza błędami wykrytymi podczas pobierania dokumentów z agencji wykonawczej powodują "błędne zakończenie",
- nie jest realizowany system rozliczeń,
- nie jest możliwy najwyższy poziom zaufania pomiędzy systemami,
- nie są tworzone rekordy zapisu stanu transferu,
- manipulacje na aktywnościach agencji umożliwiają tylko wykonywanie akcji "stop", "kill" oraz krótkiego komunikatu,
- nie dopuszcza się stosowania usług FTAM (file transfer, access management) w celu otrzymania lub dysponowania dokumentami,
- nie jest możliwe zastosowanie mechanizmu "hold" dla aktywności.

Inną ważną charakterystyką implementacji usług JTM jest typ zaimplementowanych agencji oraz rodzaj rzeczywistych urządzeń lub funkcji przetwarzających, w które mogą być odwzorowywane. W szczególności implementacja JTM może wykorzystywać tylko agencję inicjującą lub wykonawczą bądź tylko agencję docelową stanowiącą jedną drukarkę. Z drugiej strony wykorzystywane mogą być wszystkie rodzaje agencji odwzorowujące wszystkie urządzenia oraz funkcje systemu. Jej podział funkcjonalny może być

podstawą do innej klasyfikacji usług JTM niż na klasę pełną oraz podstawową.

Możliwe są następujące klasy funkcjonalne usług uwzględniające zakres usług świadczonych przez implementację podstawową:

- stacja zlecająca - implementacja zapewnia mechanizmy umożliwiające generację prymitywu klasy J-INITIATE, w przypadku gdy dodatkowo możliwe jest wykorzystanie lokalnej pamięci zbiorów, dokumenty specyfikowane w J-INITIATE mogą być z niej pobierane,
- stacja monitorująca - wykorzystywana przy pomocy J-DISPOSE, raporty mogą być przekazywane do lokalnej pamięci zbiorów, skrzynek pocztowych, wysyłane na terminal użytkownika lub wyprowadzone na drukarce,
- stacja przetwarzająca - wykorzystywana jest agencja wykonawcza, do której mogą być przesyłane dokumenty z wykorzystaniem J-DISPOSE, możliwe jest stosowanie prymitywów J-STATUS, J-STOP, J-KILL oraz dokumenty otrzymane z agencji wykonawczej mogą być łączone w jeden,
- pełna stacja klasy podstawowej.

Przedstawione klasy funkcjonalne usług nie wyczerpują pełnej klasyfikacji usług, obrazują jedynie pewne charakterystyczne usługi, które są świadczone przez JTM.

6. UWAGI KOŃCOWE

W pracy przedstawiono usługi transferu i manipulacji zadaniami w sieci komputerowej bazując na aktualnych dokumentach ISO o statusie DP. Ponieważ prace standaryzujące są kontynuowane, należy oczekiwać pewnych zmian w przedstawionym materiale, jednakże nie powinny one dotyczyć podstawowych założeń usługi JTM. W szczególności dotyczyć one będą klasy pełnej, dla której nie opracowano dotychczas dokumentu o statusie DP.

Należy podkreślić, że pierwsze wersje usług JTM opracowane przez ISO [6,7], pomimo wielu różnic w stosunku do późniejszych opracowań [10,11,12,13], zawierają te same - na przyjętym artykule poziomie szczegółowości - podstawowe idee organizacji przetwarzania rozproszonego. Analizowana w opracowaniu usługa transferu i manipulacji zadaniami została zaimplementowana w sieci MSK na komputerze ODRA 1305 [4]. Mimo istotnych różnic pomiędzy strukturą sieci MSK a systemem otwartym brak warstw prezentacji i sesji podstawowe idee zawarte w omawianych stan-

dardach znalazły pełne odzwierciedlenie w implementowanej usłudze, w tym również przeniesione do protokołów transferu i manipulacji zadaniami oraz transferu zbiorów dla sieci MSK elementy które zgodnie ze standardem systemu otwartego powinny znaleźć się w warstwach niższych. Wspomniana implementacja umożliwia przesyłanie zadań do zdalnych systemów przetwarzających oraz manipulację na tych zadaniach (skasowanie, zapytanie o status) oraz otrzymywanie raportów o stanie transferu zadania oraz jego wykonania.

Mimo powyższych zastrzeżeń autorzy mają nadzieję, że przedstawione opracowanie będzie pomocne w zrozumieniu problemów związanych z przetwarzaniem zadań w środowisku rozproszonym jakie tworzy sieć komputerowa.

7. LITERATURA

- [1] Huzar Z., Kwiatkowski J., Protokół transferu i manipulacji zadaniami dla sieci MSK, CO Pwr., Rap.SPR. 16/84, 1984.
- [2] Huzar Z., Programowa specyfikacja protokołu transferu deskryptorów, CO Pwr., Rap.SPR. 28/82, 1982.
- [3] Huzar Z., Kaliś A., Kwiatkowski J., Problemy zarządzania w sieci komputerowej, Prace Nauk. CO Pwr. 3 1, Konferencja "Sieci komputera - usługi, protokoły, modele" cz.1, 109-123, Wrocław 1985.
- [4] Jacukowicz M. i inni, Opis stacji transferu zbiorów i zadań dla KO ODRA 1305, CO Pwr., Rap.SPR. 30/84, 1984.
- [5] Kwiatkowski J., Zasady transferu i przetwarzania zadań w protokole przetwarzania zadań, CO Pwr., Rap.SPR. 11/82, 1982.
- [6] ISO/TC97/SC16 Job transfer and manipulation concepts and services, August 1982.
- [7] ISO/TC97/SC16/N 1248 Job transfer and manipulation protocol, August 1982.
- [8] Network independent job transfer and manipulation protocol, DCPU/JTMP/80/1, 1980.
- [9] Network independent job transfer and manipulation protocol - revised version, DCPU/JTMP/81, 1981.

- [10] ISO/TC97/SC16 N1751, Job transfer and manipulation concepts and services, Stable working draft, November 1983.
- [11] ISO/TC97/SC16 N1757, Job transfer and manipulation - protocol specification, Working draft proposal, January 1984.
- [12] ISO/TC97/SC21 N521, Specification of the basic class protocol for job transfer and manipulation, Draft proposal ISO/DP8832, September 1985.
- [13] ISO/DP8831, Information processing - open systems interconnection - JTM Concepts and Services, July 1985.
- [14] ISO/DP8822, Information processing - Presentation Service Definition.
- [15] ISO/DIS8824, Information processing - Open Systems Interconnection, Specification of abstract syntax notation One ASN.1.
- [16] ISO/DP8649/2, Information processing - Open systems interconnection - Definition of Common Application Service Element.
- [17] ISO/DP8649/3, Information processing - Commitment, Concurrency and Recovery Common Application Service Element.
- [18] ISO 7498, Information processing systems - Open systems interconnection - Basic reference model.

DISTRIBUTED JOB PROCESSING IN COMPUTER NETWORK

One of the services in computer network standardized by International Organization for Standardization(ISO) is Job Transfer and Manipulation service (JTM). In the paper current state of research work from this area is presented. In the first part the basic structure and principle of JTM processing is described. After short description and classification of JTM service with respect to basic class which is standardized separately is given.

РАСПРЕДЕЛЕННАЯ ОБРАБОТКА ЗАДАНИЙ В ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

Одной из услуг, предоставляемых вычислительной сетью является услуга передачи и манипуляции заданиями, стандартизованная Международной Организацией Стандардов (МОС). Рассматривается текущее состояние работ, касающихся стандартизации этой услуги. В первой части работы представляется общая структура и принципы функционирования системы передачи и манипуляции заданиями. Затем определяется диапазон и проводится классификация вышеупомянутых услуг. Особое внимание уделяется так называемому основному классу, для которого разработан отдельный стандарт.

Zbigniew HUZAR^x, Andrzej KALIŚ^x, Piotr KOCIATKIEWICZ^{xx}.

sieć komputerowa,
usługi, protokoły

PROJEKTY I REALIZACJE SIECI KOMPUTEROWYCH

Przedstawiono ogólny stan i tendencje rozwoju sieci komputerowych, a następnie dokonano przeglądu ważniejszych sieci komputerowych w USA i Europie. Na tym tle omawia się dwa projekty: MSK oraz SKJS2 wersja 1, realizowane w Polsce. Opisano podstawowe założenia projektowe i realizacyjne, przedstawiono i oceniono uzyskane wyniki obu projektów.

1. WSTĘP

Aktualny stopień rozpowszechniania sieci komputerowych na świecie jest tak znaczący, że w krajach wysoko rozwiniętych stały się one nieodłącznym elementem środowiska pracy badawczej i naukowej. Pomimo, że sieci wciąż rozwijają się i rośnie liczba ich użytkowników, to jednak liczba sieci funkcjonujących w pełni użytkowym reżimie jest ciągle dosyć ograniczona [20]. Powstrzymująca jest tu bariera finansowa obejmująca koszty projektowania i budowy, a zwłaszcza eksploatacji sieci komputerowych. Sieci wciąż jeszcze nie stanowią jednorodnego, ostatecznie uformowanego obiektu, a raczej przeciwnie: ich rozwój ujawnia wciąż nowe problemy i różnorodne propozycje ich rozwiązań. Wynika to głównie stąd, że pokonawszy problemy techniczne umożliwiające budowanie sieci o praktycznie nieograniczonym zasięgu, stanęły one w obliczu olbrzymiej różnorodności sposobów ich wykorzystania i współpracy pomiędzy różnymi typami użytkowników. Nakreślona sytuacja znajduje, między innymi, odzwierciedlenia w dziedzinie stosowanych standardów. Przykładowo w stosunkowo najlepiej opanowanym obszarze transmisji danych, używa się na świecie - pomimo istniejących norm CCITT - co najmniej kilkanaście różnych standardów. Warto przy tym zwrócić uwagę na to, że w krajach Europy Zachodniej stosuje się około 10 takich standardów, podczas gdy w USA - 3, a w Japonii tylko 2 [33].

^xCentrum Obliczeniowe Politechniki Wrocławskiej

^{xx}Instytut Komputerowych Systemów Automatyki i Pomiarów

Znamienny jest też fakt, że prace standaryzacyjne prowadzone przez ISO posuwają się znacznie wolniej niż to przewidywano jeszcze 2-3 lata temu [7,31]. Obecnie dominujące w świecie sieci są oparte o standardy opracowane w "prehistorii" jak np. sieć ARPANET, bądź też o standardy firmowe np. BITNET, NETNORTH, EARN oparte o rozwiązania korporacji IBM.

Świadomość, że sieci komputerowe będą w przyszłości stawać się elementem światowego obiegu i przetwarzania informacji, na wzór dotychczas stosowanych sieci połączeń telefonicznych, teleksowych, radiowych itp., zrodziła potrzebę nowego spojrzenia na różne systemy łączności. Stąd wyłoniła się koncepcja ISDN- zintegrowanego systemu usług komunikacyjnych i oparcia o takie systemy wszystkich nowoprojektowanych sieci komputerowych. W krajach wysoko rozwiniętych podjęto intensywne prace zmierzające do szybkiego wdrożenia systemów ISDN. Sytuacja jest przy tym niejednorodna, Przykładowo, towarzystwa telekomunikacyjne w USA już od pewnego czasu poświęcają rocznie wielomiliardowe nakłady na rozbudowę i modernizację ("cyfryzację") swych systemów, podczas gdy poczty w Europie przeznaczają na rozwój tylko około 2% swych budżetów, pobierając przy tym znacznie, niekiedy aż 6 krotnie, wyższe opłaty za swoje usługi.

Zasadnicze usługi, które są wykorzystywane w sieci to transfer zbiorów danych oraz różne formy poczty elektronicznej. Na bazie tych usług znalazły rozpowszechnienie głównie systemy gromadzenia i udostępniania informacji. Przykładowo Europejska Agencja Przestrzeni Kosmicznej udostępnia poprzez sieci komputerowe ponad 80 banków danych, wśród nich znane krajowym użytkownikom bazy takie jak INSPEC, ISMPEC, PASCAL. Statystyki wskazują, że przeciętnie w ciągu godziny, odwołuje się do nich około 7 tys. użytkowników, w tym około 30-40% stanowią zapytania w trybie konwersacyjnym [33].

W trakcie swego rozwoju wykształciły się wyraźnie trzy typy sieci komputerowych: lokalne (instytucjonalne), krajowe i międzynarodowe. Z każdym z tych typów wiążą się różne problemy techniczne, standaryzacyjne, organizacyjne i prawne. Pierwsza bazuje na prostej podsieci transmisji danych, pozostającej całkowicie w gestii użytkownika, natomiast pozostałe na podsieciach transmisji danych, stanowiących wydzielone systemy, zarządzane przez poczty lub towarzystwa telekomunikacyjne.

Artykuł dotyczy wyłącznie sieci krajowych i międzynarodowych, określonych wspólnym mianem sieci rozległych.

2. PRZEGLĄD WYBRANYCH SIECI KOMPUTEROWYCH

Przegląd obejmuje najważniejsze, niekomercyjne sieci komputerowe, które są związane ze środowiskami akademickimi lub badawczymi. Największe takie sieci funkcjonują na obszarze USA. Należą do nich: ARPANET,

CSNET, BITNET, UUCP, CONET [20].

ARPANET chociaż jest siecią o kilkunastoletniej historii, ciągle pozostaje siecią żywą i atrakcyjną dla wielu użytkowników. Obecnie łączy ze sobą ponad 1000 komputerów obliczeniowych oraz jeszcze większą liczbę sieci lokalnych. Główne usługi, których dostarcza to: poczta elektroniczna, transfer zbiorów, zdalne przetwarzanie zadań oraz katalogi informacyjne o swych zasobach. Jest eksploatowana poprzez zamknięte grupy użytkowników skupiających się na wybranej dziedzinie zastosowań. Dołączenie do sieci nowego komputera obliczeniowego wymaga zgody departamentu obrony USA.

CSNET powstała w 1979 roku, w celu połączenia ośrodków badawczych głównie informatycznych z terenu USA, a także innych krajów - obecnie z Wielkiej Brytanii, RFN, Izraela i Korei Południowej. Jest siecią sponzorowaną przez firmę Bolt-Beranek-Newman, częściowo korzysta ze standardów sieci ARPA.

BITNET jest siecią komputerów firmy IBM. Inne komputery, np. firmy DEC, pracują w sieci na zasadzie emulacji komputerów IBM. Łączy ze sobą użytkowników z departamentu obrony z innymi użytkownikami, w tym z ośrodkami akademickimi. Chociaż powstała dopiero w 1981 roku, uzyskała szerokie rozpowszechnienie: obecnie wraz ze współpracującymi sieciami, NETNORTH w Kanadzie i EARN w Europie, liczy ponad 1000 komputerów obliczeniowych. Na potrzeby sieci powstała wyspecjalizowana firma EDUCOM zajmująca się tworzeniem i udostępnianiem w sieci nowych usług.

UUCP jest obecnie największą siecią używaną przez uniwersytety. Łączy obecnie ponad 5000 komputerów pracujących pod nadzorem systemu operacyjnego UNIX. Jest siecią całkowicie otwartą dla nowych użytkowników i nie posiada centralnego administratora. Zasadnicze usługi, których dostarcza to poczta elektroniczna i transfer zbiorów, a ponadto dla swych europejskich partnerów (sieci, USENET, EUNET) dostarcza biuletynów informacyjnych o swych usługach.

CONET jest siecią jednorodną komputerów firmy DEC, pracujących pod nadzorem systemu operacyjnego VMS. Jest jedną z nielicznych sieci jednorodnych akceptowanych przez inne sieci, m.in. przez BITNET.

Poza wymienionymi sieciami istnieje w USA wiele innych, niekomercyjnych sieci eksperymentalnych, np. MPENET w Lawrence Livermore Laboratory, łącząca komputery o bardzo dużej wydajności liniami o prędkości transmisji 56 kb/s, bądź też sieci specjalizowanych, np. SOCNET i POLINET w North Carolina University dla potrzeb nauk społecznych i politycznych, udostępniające zresztą swe usługi w skali międzynarodowej za pośrednictwem komercyjnej sieci transmisji danych TINET.

Spośród niekomercyjnych sieci europejskich największą jest obecnie sieć EARN (European Academic Research Network) [20]. Aktualnie łączy ona 15 krajów zachodnich, przy czym w poszczególnych krajach jej użytkownikami są, w większości przypadków, całe sieci krajowe. EARN jest odpowiednikiem i równocześnie partnerem amerykańskiej sieci BITNET. Sieć jest oparta o sprzęt IBM z komputerami pracującymi pod nadzorem systemu operacyjnego RSCS (Remote Spooling Communication Subsystem) i dostarcza typowych usług sieciowych: poczty elektronicznej, transferu zbiorów, komunikacji pomiędzy terminalami, serwisów informacyjnych o zasobach. Sieć rozpoczęła działalność pod koniec 1984 roku i do końca 1987 roku jej eksploatacja jest wspierana przez firmę IBM. IBM pokrywa całkowicie koszty transmisji danych oraz materiałów eksploatacyjnych. Po 1987 roku jest przewidywane wprowadzenie usług "małego" ISDN-tzn. transmisji danych i fonii.

Obecnie prawie każdy kraj w Europie Zachodniej posiada własną, w pełni funkcjonującą sieć komputerową. Niektóre z tych sieci przygotowują się do wdrożenia usług ISDN. Należą do nich brytyjska sieć JANET (Joint Academic Network) [37], oraz zachodnioniemiecka DFN (Das Deutsche Forschungsnetz), które są również użytkownikami sieci EARN [20].

W krajach socjalistycznych rozwój sieci komputerowych jest znacznie mniej zaawansowany. Projekty budowy sieci eksperymentalnych podjęły akademie nauk pod koniec lat siedemdziesiątych. Najbardziej znaczącym efektem tych przedsięwzięć są: AKADEMSET- w ZSRR [23] oraz DELTA- w NRD [24]

AKADEMSET łączyła w 1985 roku 11 dużych ośrodków badawczych, rozprzestrzenionych na terenie całego ZSRR. Każdy z tych ośrodków tworzy sieć regionalną, spośród których największą stanowiła sieć rozmieszczona w Rydze na terenie Łotewskiej Akademii Nauk. Komputerami obliczeniowymi są w niej EC1033, EC1055 z systemem operacyjnym OS 6.1, terminalami - gronami monitorów EC7920, oraz systemy minikomputerowe SM-4, SM-1420, Elektronika-100/25, Iskra-226. Podsieć z komutacją pakietów jest oparta o minikomputery SM-4.

Usługi, które sieć ta oferuje to:

- praca w systemie z podziałem czasu (odpowiednik TSO w komputerze EC1032) w wybranym komputerze obliczeniowym,
- rozproszone przetwarzanie zadań w trybie wsadowym i konwersacyjnym,
- konwersacyjna obsługa terminali (odpowiednik SKOT-a w komputerze EC1032) w wybranym komputerze obliczeniowym,
- system zarządzania i wyszukiwania lokalną bazą danych wybranego komputera,
- system planowania pracy komputerów w sieci.

Sieć DELTA łączy aktualnie około 10 komputerów obliczeniowych typu Riad oraz Besm6, położonych głównie na terenie Berlina i Drezna. Eksploatowana od 1981 roku dostarcza podobnego zakresu usług co sieć AKADEMSET.

3. SIEĆ KOMPUTEROWA MSK

3.1. Geneza projektu

Problematyką projektowania i budowy, a zwłaszcza zastosowań sieci komputerowych w Polsce jest opóźniona w stosunku do krajów produkujących w tej dziedzinie niemal o tyle lat, ile ich liczy historia rozwoju sieci [2]. Pierwsze przedsięwzięcie w tym zakresie zostało podjęte w roku 1979 w ramach problemu resortowego MNSzWiT (RI-14) pod hasłem "Komputeryzacja szkół wyższych" koordynowanego przez Politechnikę Wrocławską. Celem głównym jednego z podprogramów problemu było zbudowanie eksperymentalnej sieci komputerowej (MSK), która po okresie badań i eksperymentów miała stać się załącznikiem trwale funkcjonującej sieci łączącej akademickie środowiska w kraju.

W roku 1980 Centrum Obliczeniowe Politechniki Wrocławskiej rozpoczęło udział w programie RI-14 podpisując jedną szeroką umowę na zaprojektowanie i budowę sieci komputerowej MSK. Wykonawca miał dostarczyć zaprojektowane przez siebie moduły programowe i sprzętowe, oraz stworzyć rozwiązanie modelowe stanowiące sieć komputerową.

3.2. Główne wymagania i ograniczenia

W roku 1980 opracowano program budowy sieci MSK [3], w którym zawarto założenia konstrukcyjne sieci. Podstawowymi założeniami i konstrukcjami sieci było przyjęcie architektury systemów otwartych ISO OSI wraz z uznanymi w tym czasie międzynarodowymi standardami, zaś w obszarze nieogarniętym standaryzacją, oparcie się na rozwiązaniach krajów produkujących w dziedzinie rozległych sieci komputerowych. W momencie opracowywania programu budowy sieci MSK, nie było uznanej opinii dotyczącej wyboru rodzaju podsieci transmisji danych. Organizacje pocztowe krajów zachodnich w głównym zakresie, a nasza poczta wyłącznie, świadczyły usługi transmisji danych w oparciu o komutację kanałów. Stąd strategiczną była decyzja o wyborze podsieci z komutacją pakietów (bez datagramów), szczególnie adekwatną do potrzeb sieciowych.

Podstawowe założenia realizacyjne sieci MSK były następujące:

1. sprzęt komputerowy oraz wyposażenie pomocnicze powinno być produkcyjne krajowej lub krajów RWPG,
2. oprogramowanie sieciowe powinno być zaprojektowane i uruchomione w kraju,

3. sieć MSK powinna być siecią heterogeniczną i otwartą.

Heterogeniczność sieci oznacza, że powinna ona umożliwić połączenie różnych komputerów, celem nawiązania współpracy (w przypadku MSK są to Odry 1305 i EC1032). Zaś otwartość oznacza, że powinna ona umożliwić łatwą jej rozbudowę o nowe węzły sieci.

Głównymi wymaganiami, które legły u podstaw sformułowania powyższych założeń realizacyjnych, były następujące:

1. uniezależnienie się od rozwiązań programowych i sprzętowych krajów produkujących w dziedzinie rozległych sieci komputerowych,
2. stworzenie samodzielnych grup projektantów i realizatorów poszczególnych fragmentów sieci komputerowej,
3. łatwy rozwój stworzonego oprogramowania i sprzętu, jego konserwacja i pielęgnacja,
4. wykorzystanie zasobów programowych innych środowisk akademickich kraju, stymulując w ten sposób ich specjalizacje w zakresie świadczonych usług sieciowych.

Główne ograniczenia z jakimi przyszło borykać się projektantom i wykonawcom sieci MSK, był zupełny brak korelacji pomiędzy programem badawczym a programem inwestycyjnym rozwoju ośrodków obliczeniowych szkół wyższych. Szczególne braki są odczuwalne w zakresie pamięci dyskowych, a zwłaszcza specjalistycznych urządzeń teletransmisyjnych.

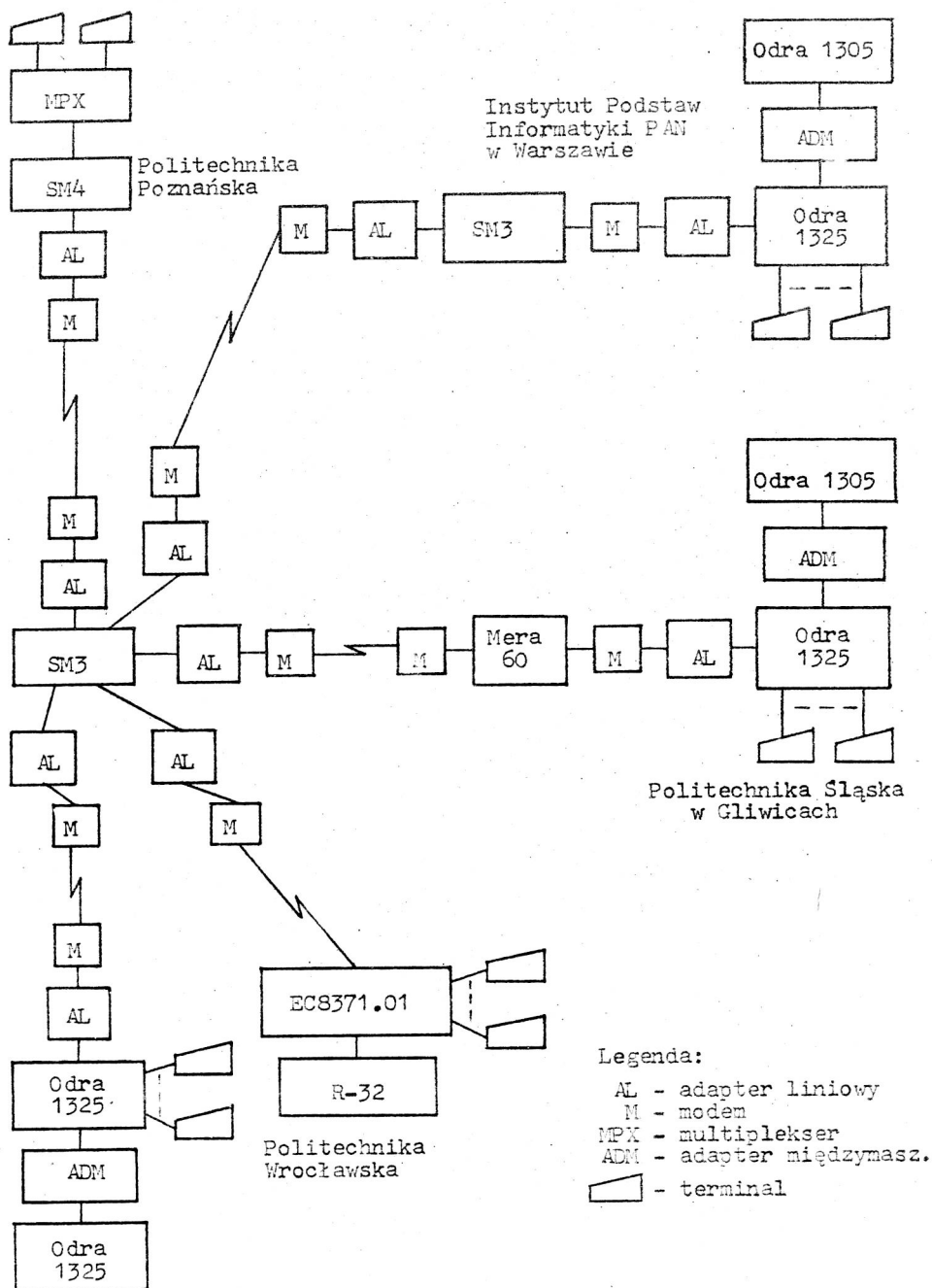
3.3. Struktura fizyczna eksperymentalnej sieci MSK

Sieć komputerowa MSK jest zbiorem systemów komputerowych, połączonych poprzez wspólną podsieć transmisji danych. Strukturę fizyczną sieci przedstawiono na Rys.1.

W zestaw środków technicznych sieci wchodzi:

- a) komputery obliczeniowe - ODRA1305 i EC1032,
- b) komputery czołowe - ODRA1325 i EC8371.01,
- c) komputery węzła - SM-3 i MERA-60,
- d) adaptory międzymaszynowe - ADM305,
- e) adaptory liniowe - ALS11, ADL11, UAL01,
- f) modemy - EC8013, EC8006 (1200 2400 b s),
- g) terminale - rodzina EC7910 i DZM.

Terminale EC7900 włączone są do sieci jako terminale zdalne poprzez procesor teleprzetwarzania danych EC8371.01 a terminale typu DZM, poprzez komputer ODRA1325.



Rys. 1 Struktura sieci komputerowej MSK

Fig. 1 The structure of the MSK computer network

3.4. Oprogramowanie sieci

Oprogramowanie sieci dzieli się na trzy podstawowe grupy:

a) Oprogramowanie komputerów węzła.

W sieci MSK występują dwa rodzaje komputerów węzła, oprogramowane od podstaw na minikomputerach SM-3 [32] oraz MERA-60. Oprogramowanie oparto o następujące zalecenia standaryzacyjne:

- poziom fizyczny: X.21 bis [36],
- poziom liniowy: X.25.2 [35],
- poziom sieciowy: X.25.3 bez usług datagramowych, jako styku do warstwy (protokół zewnętrzny) oraz zmodyfikowaną, semidatagramową wersję X.25 do komunikacji pomiędzy węzłami (protokół wewnętrzny) [15].

b) Oprogramowanie komputerów czołowych

Oprogramowanie komputerów czołowych, tzn. ODRY1325 i EC8371.01 dostarcza usług transportowych oraz wirtualnego terminala. Oprogramowanie oparto o następujące zalecenia standaryzacyjne:

- poziom transportowy: opracowanie własne na podstawie roboczego materiału ISO DP N1169, zbliżonego do późniejszego dokumentu ISO DIS 8073 [1],
- wirtualny terminal: opracowanie własne oparte na wstępnych materiałach ISO oraz ECMA [38].

Do podstawowych usług świadczonych przez oprogramowanie komputera czołowego należą:

- łączność dowolnego terminala z dowolną aplikacją sieciową komputera obliczeniowego,
- łączność dowolnego terminala z dowolnym innym terminalem sieci (namiastka poczty elektronicznej),
- nawiązywanie połączeń na poziomie warstwy transportowej, z których korzystają aplikacje sieciowe komputera obliczeniowego.

Oprogramowanie komputerów czołowych stworzono od podstaw. Nie wykorzystuje się firmowych telekomunikacyjnych metod dostępu [4].

c) Oprogramowanie komputerów obliczeniowych

Moduły programowe stanowiące oprogramowanie sieciowe wykonano przy założeniu nie wprowadzania daleko idących zmian w istniejących systemach operacyjnych. Przyjęto, że we wszystkich komputerach obliczeniowych udostępni się do pracy sieciowej podsystemy wielodostępu (MFC, MSC). Wymagało to wprowadzenia zmian w tych podsystemach. Dodatkowo opracowano specjalny podsystem wyszukiwania i zarządzania bazą danych TIOI umiejscowiony w komputerze ODRA1305. Za jego pośrednictwem użytkownicy ze swoich terminali mogą docierać do informacji zgromadzonych

w bazie INSPEC sekcja C.

We wszystkich komputerach udostępniono usługę transferu zbiorów sekwencyjnych [11].

Oprogramowanie komputerów wykorzystuje następujące zalecenia standardyzacyjne:

- protokół transferu zbiorów: opracowanie własne oparte na protokole brytyjskim NIFTP, równoważnym protokołowi INWG86 opracowaniem przez IFIP [25],
- protokół transferu i manipulacji zadaniami: opracowanie własne, oparte na protokole brytyjskim NIJTMP.

3.5. Charakterystyka użytkowa sieci MSK

Końcowym wynikiem prac nad siecią MSK jest pilotowa wersja heterogenicznej sieci komputerowej.

Zakres usług jakie oferuje ona użytkownikowi jest następujący:

1. umożliwia użytkownikowi terminala sieciowego (podłączonego do dowolnego komputera czołowego) pracę w trybie konwersacyjnym z dowolnie wybranym podsystemem wielodostępnym (MOP, TSO),
2. umożliwia przesyłanie informacji pomiędzy parą dowolnych wybranych terminali sieciowych,
3. pozwala użytkownikowi na transfer zbiorów sekwencyjnych znakowych,
4. udostępnia użytkownikowi w trybie konwersacyjnym informacje bibliograficzne z fragmentu bazy danych INSPEC,
5. udziela użytkownikowi informacji o zasobach sieci.

3.6. Ocena osiągniętych wyników

Do osiągnięć podprogramu "Budowa sieci komputerowej MSK" należy zaliczyć:

- osiągnięcie podstawowego celu technicznego jakim było uruchomienie pilotowej wersji sieci heterogenicznej,
- ustalenie i przyjęcie standardów, zgodnie z aktualnym ich rozwojem w świecie,
- uzyskanie rozwiązań technicznych i programowych opartych całkowicie o sprzęt i oprogramowanie krajowe,
- uformowanie zespołów projektowo-badawczych zdolnych do samodzielnej konwersacji i rozwijania wytworzonych komponentów sprzętowych i programowych,
- utworzenie załączka infrastruktury organizacyjnej ("administrator sieci") zdolnej do przejmowania i wprowadzania do eksploatacji komputerów sieci,

- prowadzenie w skali eksperymentalnej eksploatacji użytkowej sieci. Po stronie braków przewidywanych osiągnięć znajdują się: nie uzyskanie pełnego celu użytkowego sieci (brak zasobów) oraz niedokończenie prac nad kilkoma komponentami sieci (adaptory liniowe z bezpośrednim dostępem do pamięci, koncentrator terminali na SM-4). Nie uwzględnienie polityki inwestycyjnej, a także zagadnienia kosztów eksploatacji sieci komputerowej przy realizacji programu resortowego, stała się powodem opóźnienia oczekiwanego efektu.

Sieć komputerowa MSK przedstawiona jest w ponad 100 raportach oraz opublikowana w 62 pracach (kongresy IFIP, czasopisma zagraniczne i krajowe, konferencje międzynarodowe oraz krajowe). Szczegółowe podsumowanie prac nad siecią MSK zawiera raport [2], oraz seria artykułów [5,7,8].

4. SIEĆ KOMPUTEROWA SKJS 2 WERSJA 1

4.1. Geneza projektu

W 1981 roku przystąpiono w Instytucie Komputerowych Systemów Automatyki i Pomiarów do realizacji programu budowy sieci komputerowej [26, 27]. Główną przesłanką rozpoczęcia prac było szukanie odpowiedzi na pytanie, co w sytuacji opanowania produkcji i dostaw przez ZE ELWRO systemów komputerowych EC1032 i podsystemów TELE JS należy przedsięwziąć by zapewnić ciągłość popytu na te systemy i lepiej zaspakajać zapotrzebowanie użytkowników? Zdecydowano, że jednym ze sposobów spełnienia tego postulatu będzie stworzenie warunków łączenia podsystemów TELE JS w instalacje sieciowe.

Rozpoczęte w IKSaIP prace kontynuowano od roku 1984 we współpracy z Centrum Obliczeniowym Politechniki Wrocławskiej, gdzie opracowano szczegółowy projekt, założenia funkcjonalne i wykonano zasadnicze elementy oprogramowania sieciowego sieci SKJS 2 wersja 1 [9,21].

4.2. Główne wymagania i ograniczenia

Przystępując do projektowania sieci komputerowej SKJS 2 wersja 1 przyjęto wymagania i ograniczenia, które można sklasyfikować w następujący sposób:

a) wymagania ze strony użytkowników:

- zapewnienie komunikacji terminal-zasoby lokalnego i zdalnego komputera obliczeniowego,
- zapewnienie komunikacji terminal-terminal,
- zapewnienie transferu zbiorów pomiędzy komputerami obliczeniowymi,
- realizacja zadań w trybie przetwarzania sieciowego.

- b) wymagania ze strony projektantów systemów informatycznych:
- zapewnienie możliwości tworzenia różnorodnych konfiguracji,
 - zapewnienie możliwości rozbudowy sieci przez dołączenie komputerów obliczeniowych, węzłów podsystemów transmisji danych lub dodatkowych zasobów sieci,
 - zapewnienie swobodnego przepływu informacji w sieci,
 - zapewnienie łatwej organizacji systemów użytkowych,
 - zapewnienie akceptowania różnorodnych urządzeń końcowych.
- c) wymagania ze strony producenta:
- możliwość kompletacji w oparciu o sprzęt produkowany w ZE ELWRO i w kraju,
 - wykorzystanie w maksymalnym stopniu oprogramowania jakim dysponuje producent, szczególnie w zakresie podsystemu TELE JS.
- d) wymagania ze względu na uwarunkowania zewnętrzne:
- wykorzystanie środków łączności jakimi dysponuje resort łączności w Polsce,
 - zgodność projektu z koncepcją sieci komputerowych opracowane we Wspólnej Sekcji Specjalistów d.s. Sieci Komputerowych RGK JS EMC i RGK SM EMC, a zarazem z modelem odniesienia OSI ISO,
 - wykorzystanie doświadczeń uzyskanych w czasie prac projektowych w dziedzinie sieci komputerowych, zwłaszcza sieci MSK.
- e) wymagania eksploatacyjne:
- zapewnienie cechy łagodnego upadku sieci,
 - zapewnienie możliwości diagnozowania sieci,
 - zapewnienie mechanizmów prowadzenia pomiarów i raportowania pracy sieci.

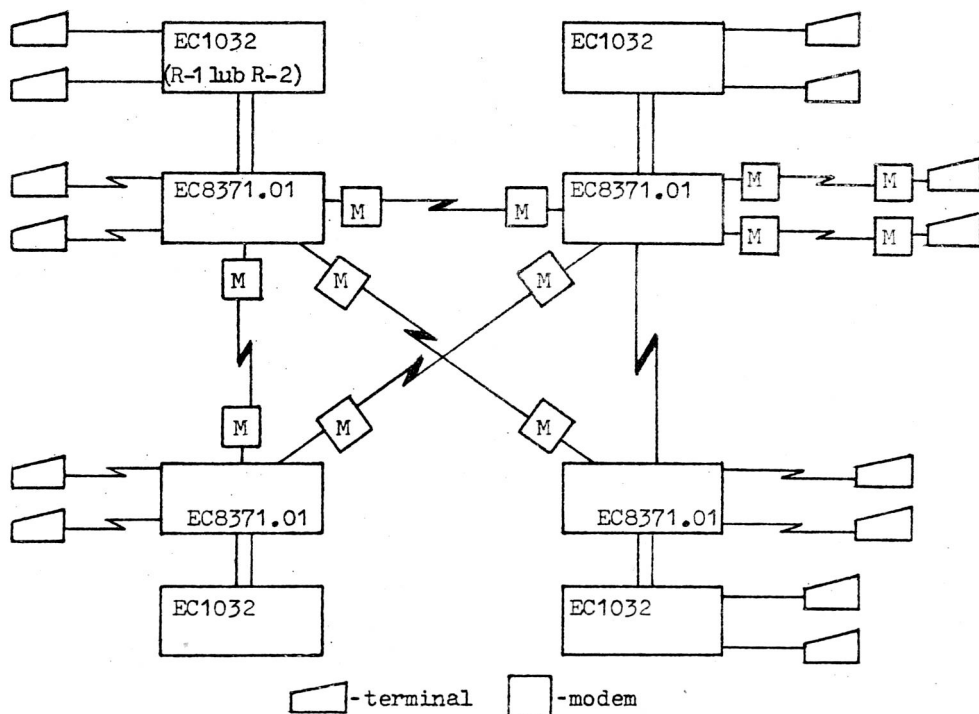
4.3. Struktura fizyczna sieci SKJS 2 wersja 1.

Sieć komputerowa SKJS 2 wersja 1 jest zbiorem systemów komputerowych, połączonych środkami transmisji danych. Struktura komunikacyjna realizowana przez sprzęt i oprogramowanie jest warstwowa, zgodna z modelem OSI/ISO.

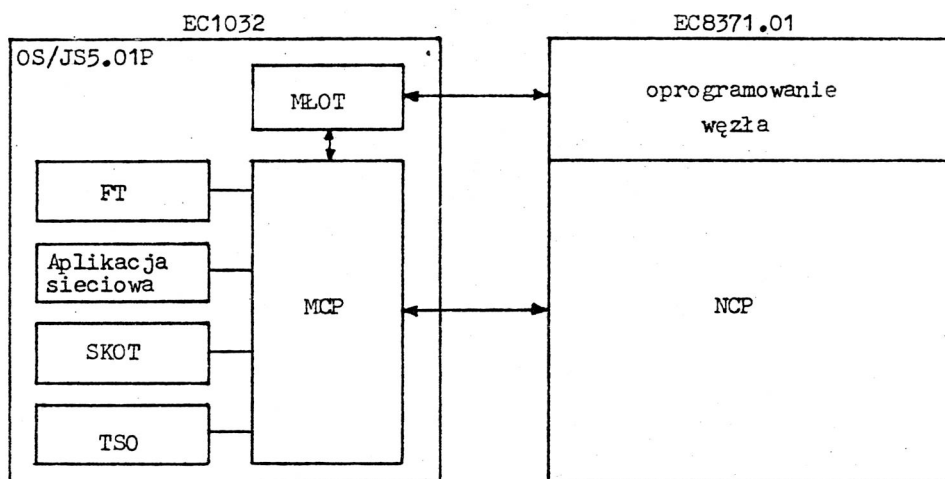
Strukturę fizyczną sieci przedstawiono na Rys.2 Cechą charakterystyczną przyjętego rozwiązania jest połączenie w procesorze teleprzetwarzania FTD funkcji urządzenia przekształcającego sprzęg dla transmisji i koncentratora terminali - dalej nazywanego komputerem czołowym (KC) z funkcjami komputera węzła (KW) podsystemu transmisji danych.

Zestaw środków technicznych sieci obejmuje:

- a) komputer obliczeniowy - w zasadzie funkcje komputera obliczeniowego pełnią systemy komputerowe EQ1032. W sieci SKJS 2 wersja 1 mogą również pracować inne komputery serii Riad-1 oraz komputery serii Riad-2.



Rys. 2 Struktura eksperymentalnej sieci SKJS2 wersja 1
Fig. 2 The structure of the experimental computer network SKJS2 version 1



Rys. 3 Struktura oprogramowania sieci SKJS2 wersja 1
Fig. 3 The structure of the computer network SKJS2 version 1 software

- b) komputer czołowy i komputer węzła-procesorem pełniącym funkcje komputera czołowego i węzła jest procesor EC8371.01,
- c) terminale - podstawowymi terminalami w sieci są: terminale EC8577M, terminale i systemy terminalowe EC7910 (EC7911, EC7914, EC7915, EC7917). Terminale mogą pracować w sieci w trybie lokalnym lub zdalnym poprzez procesor teleprzetwarzania EC8371.01
- d) środki transmisji danych. Terminale są łączone z procesorem EC8371.01 dwuprzewodową linią dzierżawioną poprzez modemy EC8006 (600/1200 b/s) Procesory EC8371.01 są wzajemnie łączone czteroprzewodowymi liniami dzierżwionymi poprzez modemy EC8013 (1200/2400 b/s).

4.4. Oprogramowanie sieci

Spełniając wymagania dotyczące wykorzystania oprogramowania systemu TELE JS wykorzystano następujące elementy oprogramowania firmowego: system operacyjny OS JS 5.01P, metodę dostępu TCAM5, sieciowy program sterujący MCP.

Dla realizacji sieciowego trybu przetwarzania i usług komunikacji w sieci opracowano:

- a) w zakresie oprogramowania komputera obliczeniowego program modułu łącznikowego obsługi terminali-MŁOT. Moduł steruje pracą terminali oraz realizuje połączenia sieciowe. Wygenerowano program MCP (dla TCAM) zapewniający realizowanie funkcji w zakresie obsługi fizycznej terminali oraz pośredniczenia pomiędzy aplikacjami a modułem MŁOT. Dostarczono usługi transferu zbiorów implementując stacje transferu zbiorów.
- b) w zakresie oprogramowania procesora EC8371.01 program sterujący komputerem czołowym PSPG. Spełnia on funkcje:
 - stacji sieciowej metody dostępu,
 - stacji transportowej,
 - stacji operatorskiej,
 - stacji sieciowych.

Ogólną strukturę oprogramowania sieci przedstawiono na Rys.3.

4.5. Charakterystyka użytkowa sieci

Opracowane produkty programowe umożliwiają budowę instalacji sieci komputerowych dla różnorodnych zastosowań. Podstawową bazę sprzętową tworzą urządzenia produkowane przez krajowy przemysł komputerowy. Możliwe jest również stosowanie urządzeń komputerowych wytwarzanych w innych krajach. Podstawowymi zasobami sieci są:

- podsystem SKOT,
- podsystem TSO,
- aplikacje sieciowe opracowane przez użytkowników.

Dostęp do zasobów jest możliwy z:

- terminali lokalnych,
- terminali zdalnych.

Sposób nawiązania połączenia nie zależy od lokalizacji zasobu. Problemy utrzymania połączenia z zasobem i sterowania wymianą informacji nie są widoczne dla użytkowników.

Dostępna jest usługa transferu zbiorów umożliwiająca przesyłanie zbiorów sekwencyjnych pomiędzy komputerami obliczeniowymi w sieci.

Ponadto opracowane oprogramowanie zapewnia:

- możliwość raportowania pracy i podstawowych stanów sieci,
- zapewnia komunikację pomiędzy operatorami systemów komputerowych tworzących sieć,
- zapewnia łagodzenie skutków awarii elementów sieci.

Dokumentacja eksploatacyjna i użytkowa sieci SKJS 2 wersja 1 zawiera 9 pozycji [5, 10, 12, 13, 14, 18, 19, 22, 29].

4.6. Ocena osiągniętych wyników

Opracowane oprogramowanie zostało gruntownie przebadane na eksperymentalnej instalacji składającej się z trzech systemów komputerowych:

1. R-32 + EC8371.01 - laboratorium BSL, IKSAiP
2. R-55 + EC8371.01 - laboratorium BSL, IKSAiP
3. R-32 + EC8371.01 - laboratorium OK, ZE ELWRO

Narzędziami badań były programy i urządzenia opracowane równolegle z oprogramowaniem sieciowym. Należą do nich zadania kontrolne, pomiarowe [28], demonstracyjne [30] oraz aparatura diagnostyczno-pomiarowa [16, 17].

Badania zgodnie z programem obejmowały:

- testowanie pracy i współdziałania zespołów funkcjonalnych sieci,
- sprawdzenie poprawności rozpoczynania, przebiegu i zamykania sesji,
- badanie funkcji użytkowych sieci,
- ocenę charakterystyk ilościowych pracującej sieci,
- ocenę niezawodności sieci,
- badania zmienności struktury sieci.

Wyniki badań wykazały, że sieć SKJS 2 wersja 1 posiada założone własności użytkowe i eksploatacyjne; wartości parametrów dynamicznych nie odbiegają od wartości parametrów eksploatacyjnych istniejących podsystemów TELE JS.

5. UWAGI KOŃCOWE

Sieci komputerowe, jak przedstawiają to rozdziały 1 i 2, stają się powszechnym środowiskiem pracy dla zespołów badawczych w uczelniach i przemyśle. Ich użytkowanie jest wyrazem szerokiego zapotrzebowania

na ich usługi, zwłaszcza na dostęp do różnych baz informacyjnych oraz szybką komunikację. Natomiast fakt ich sprawnej eksploatacji jest wyrazem dojrzałości technicznej i organizacyjnej tworzących ich komponentów, w szczególności odpowiedniego zakresu i poziomu usług telekomunikacyjnych oraz dostatecznych mocy obliczeniowych i poziomu niezawodności ośrodków obliczeniowych włączanych do sieci.

Obok sieci rozumianej jako pewna struktura sprzętowo-programowa musi istnieć "sieć ludzi" - zespołów pielęgnujących i rozwijających oprogramowanie. Rozwój sieci wymaga prowadzenia szerokiej praktyki gospodarczej obejmującej poza planowaniem prac projektowych, prowadzenie inwestycji oraz produkcji.

Przedstawione projekty MSK, SKJS wskazują, że w Polsce sieci komputerowe przeszły dopiero pierwszą fazę eksperymentalną, a zasadniczymi efektami tej fazy to, poza uzyskanymi rozwiązaniami technicznymi, wykształcenie zespołów, które mogą prowadzić dalszy rozwój.

- Rozwój sieci w Polsce wymaga rozwiązania następujących problemów:
- rozwiniecie usług telekomunikacyjnych poczty: początkowo rozbudowę sieci linii telefonicznych a docelowo dostarczenie usług transmisji danych z komutacją pakietów. Konieczne jest także podejmowanie prac nad cyfrowym kodowaniem fonii i wizji w celu przygotowania do pracy w ramach ISDN,
 - prowadzenie właściwej polityki inwestycyjnej w ośrodkach obliczeniowych wyższych uczelni i ośrodków badawczych, które w formie projektantów bądź użytkowników zamierzają uczestniczyć w sieci,
 - powiązanie prac nad rozwojem sieci z planami producentów sprzętu i oprogramowania,
 - skojarzenie prac nad rozwojem sieci z rozwojem krajowych systemów gromadzenia i dystrybucji informacji.

6. LITERATURA

- [1] Barchański J.A., Muehleisen T., Specyfikacja protokołu transportowego dla MSK, Instytut Cybernetyki Technicznej Politechniki Wrocławskiej, Raport SPR 5-82, 1982.
- [2] Battek J. i inni, Podsumowanie prac nad Międzyuczelnianą Siecią Komputerową MSK, Centrum Obliczeniowe Politechniki Wrocławskiej Raport SPR 16-86, 1986.
- [3] Battek J. i inni, Program budowy Międzyuczelnianej Sieci Komputerowej MSK w latach 1980-1985, część I i II, Wrocław 1980.
- [4] Bernardyn J. i inni, Projekt oprogramowania procesora czołowego dla mc R-32 w sieci komputerowej MSK, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 29-82, 1982.

- [5] Bernardyn R. i inni, Procedura generowania sieciowego programu procesora czołowego sieci SKJS 2 wersja 2, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 10-86, 1986.
- [6] Bieleninik L. i inni, Podsieć komunikacyjna Międzyuczelnianej Sieci Komputerowej, INFORMATYKA, nr 11-12, 1985, str.15-18.
- [7] Bilski E. i inni, Rozległe sieci komputerowe, INFORMATYKA, nr 3, 1985, str. 4-11.
- [8] Budzianowski L. i inni, Komputer ODRA 1305 w Międzyuczelnianej Sieci Komputerowej, INFORMATYKA, nr 7-8, 1986, str. 16-19.
- [9] Budzianowski L i inni, Projekt techniczny oprogramowania ITD jako procesora czołowego i komputera węzła w sieci SKJS 2 wersja 1, Instytut Cybernetyki Technicznej Politechniki Wrocławskiej, Raporty SPR 11-84 i 3-85.
- [10] Budzianowski L. i inni, Instrukcja operatorska oprogramowania ITD w sieci SKJS 2, Instytutu Cybernetyki Technicznej Politechniki Wrocławskiej, Raport SPR-3-86, 1986.
- [11] Dubielewicz A. i inni, R-32 jako komputer obliczeniowy sieci MSK- Stacja transferu zbiorów, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 4-83, 1983.
- [12] Dubielewicz A. i inni, Stacja transferu zbiorów w sieci SKJS 2 wersja 1 - Podręcznik programisty systemowego, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 23-86, 1986.
- [13] Dubielewicz A. i inni, Stacja transferu zbiorów w sieci SKJS 2 wersja 1 - Podręcznik użytkownika, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 24-86, 1986.
- [14] Dubielewicz A. i inni, Stacja transferu zbiorów sieci SKJS 2 wersja 1 - Generacja oprogramowania, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 16-86, 1986.
- [15] Dubielewicz I., Protokół poziomu pakietowego w sieci MSK, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SIR 8-84, 1984.
- [16] Dubielewicz I. i inni, Aparatura adiagnostyczno-pomiarowa dla sieci SKJS 2 wersja 1 - Dokumentacja eksploatacyjna, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 12-86, 1986.
- [17] Dubielewicz I. i inni, Aparatura diagnostyczno-pomiarowa dla sieci SKJS 2 - Dokumentacja konstrukcyjna, Raport SPR 13-86, 1986.
- [18] Dyrka K., Rutkowski E., Zasady tworzenia i udostępniania zasobów w sieci SKJS 2 wersja 1, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 14-86, 1986.
- [19] Dyrka K. i inni, Zasady technologii eksploatacji sieci SKJS 2 wersja 1, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 11-86, 1986.

- [20] Harms J., La situation internationale, SWITCH, no.1, April 1986, str.18-22,
- [21] Huzar A., i inni, Projekt techniczny oprogramowania EC1032 jako komputera obliczeniowego w sieci SKJS 2 wersja 1, Centrum Obliczeniowe Politechniki Wrocławskiej, Raporty SPR 12-84, 1984.
- [22] Huzar A., Jacukowicz M., Terminal w sieci SKJS 2 - Podręcznik użytkownika, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SIR 21-86, 1986.
- [23] Jakubajtis E.A., Architecture of computer networks for large offices, COMNET 85, Working Papers P.6-P.30, John von Neumann Society for Computing Science, Budapest 1985.
- [24] Junker E., Meier H.W., Architecture of the computer network DELTA and experience with the system, COMNET'85, Working Papers 1.50-1.54 John von Neumann Society for Computing Science, Budapest 1985.
- [25] Kaszuba R., Protokół transferu zbiorów dla sieci MSK, Wyd.2 rozsz. Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 8-84, 1984.
- [26] Kociatkiewicz P. i inni, Założenia techniczno- ekonomiczne: Sieć komputerowa SKJS 2 wersja 1, IKSAiP, nr 033-I-BPK-82, Wrocław 1982,
- [27] Kociatkiewicz P. i inni, Projekt wstępny: Sieć komputerowa SKJS 2 wersja 1, IKSAiP, nr 093-I-BPK-83, Wrocław 1983.
- [28] Kremeniewski P. i inni, Dokumentacja eksploatacyjna zadań kontrolnych i pomiarowych do badań sieci SKJS 2 wersja 1, IKSAiP, Wrocław 1986.
- [29] Łukaszewicz A., Zasady generowania i eksploatacji oprogramowania sieciowego w sieci SKJS 2 wersja 1, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 20-86, 1986
- [30] Milewska M i inni, Dokumentacja eksploatacyjna zadań demonstracyjnych dla sieci SKJS 2 wersja 1, IKSAiP, Wrocław 1986.
- [31] Prace Naukowe Centrum Obliczeniowego Politechniki Wrocławskiej, 1985, nr 3, Seria Konferencje, nr 1; Sieci Komputerowe-Usługi, protokoły, modele, część 1 i 2.
- [32] Prace Naukowe Centrum Obliczeniowego Politechniki Wrocławskiej, 1983, nr 2, Seria Studia i Materiały, nr 2.
- [33] Sadlak J. Memorandum concerning NIT based networks nad related forms of scientific information exchange in Europe, Standing Conference of Rectors of the European Universities, Policy paper, no. 1, Geneva, March 1986.
- [34] Sadlak J. The future of academic networks in Europe, CRE-INFORMATION, no. 75, 4rd Quarter, 1986.str.163-167.
- [35] Stanisław A., Protokół poziomu liniowego w sieci MSK, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 6-84, 1984.

- [36] Stanisław A., Protokół poziomu fizycznego w sieci MSK, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport-SPR 5-84, 1984.
- [37] Wells M., The Janet project, University Computing, vol.6, 1984. str. 56-62.
- [38] Wietrzych J., Usługi i protokół terminala interakcyjnego w MSK. Instytut Cybernetyki Technicznej Politechniki Wrocławskiej, Raport SPR 3-83, 1983.

COMPUTER NETWORK PROJECTS AND IMPLEMENTATIONS

The current trends of development of a wide computer network and a review of the most important wide area computer network is presented. On this background two projects realized in Poland are discussed - MSK and SKJS2 version 1. General project and implementation assumption and obtained results are presented.

ПРОЕКТЫ И РЕАЛИЗАЦИИ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ

Рассматривается общее состояние и перспективные направления развития вычислительных сетей, затем обсуждаются самые важные вычислительные сети в США и в Европе. На этом фоне описываются два проекта проводимые в Польше: MSK и СКЙС2 версия 1. Приводятся основные проектные и реализационные концепции, оцениваются результаты обоих проектов.

Andrzej KALIŚ^x
Roman KASZUBA^x

sieć komputerowa,
transfer zbiorów

TRANSFER, DOSTĘP I ZARZĄDZANIE ZBIORAMI W MODELU SYSTEMU OTWARTEGO

Jedną z usług świadczonych przez sieć komputerową jest usługa transferu, dostępu i zarządzania zbiorami (FTAM) objęta standaryzacją przez Międzynarodową Organizację Standaryzacyjną (ISO). W pracy przedstawiono koncepcję usług zbiorów w systemie otwartym oraz dane szczegółowe pozwalające na zorientowanie się w zakresie usług objętych standardem.

1. WSTĘP

Środowisko heterogenicznej sieci komputerowej, w której występuje mnogość i różnorodność systemów komputerowych i przynależnych do nich pamięci zbiorów, stwarza potrzebę posiadania mechanizmu pozwalającego na wykorzystanie oferowanie usług przechowywania i wyszukiwania informacji. Powinien on pozwalać na niezawodne przesyłanie informacji pomiędzy systemami. Mechanizm tego rodzaju określany jest mianem usług transferu, dostępu i zarządzania zbiorami, zwanych dalej w skrócie usługami zbiorów. W zasadzie usługi te dotyczą manipulacji na dowolnych identyfikowalnych porcjach informacji, które można traktować jako zbiory. Ponadto usługi te mogą być wykorzystane nie tylko do przenoszenia informacji, pomiędzy pamięciami zbiorów, bądź procesem a pamięcią zbiorów, ale także pomiędzy procesami.

W niniejszym opracowaniu przedstawiono funkcjonowanie takich usług, których funkcje są wzięte z propozycji standaryzacyjnych Międzynarodowej Organizacji Standaryzacyjnej (ISO). W tym zakresie zostały dotychczas opracowane dwa kolejne dokumenty o statusie szkicu standardu (DP). Terminologia opracowania jest również oparta o pojęcia wprowadzone przez ISO.

Dokument standaryzacyjny [1,2,3,4] definiuje model pamięci zbiorów nazywany wirtualną pamięcią oraz usługi i protokół (FTAM) podwar-

^x Centrum Obliczeniowe Politechniki Wrocławskiej, Wrocław

stwy aplikacyjnej należącej do kategorii specyficznych elementów usług aplikacyjnych (SASE). Omawiane usługi są przeznaczone do współpracy użytkownika z odległą pamięcią zbiorów lub do wykorzystania przez inne protokoły aplikacyjne. Są one przedstawiane w sposób abstrakcyjny, zarówno ze względu na model interakcji użytkowników z usługodawcą, który określa jedynie rodzaj i kolejność wymienianych pomiędzy nimi informacji (nie ogranicza to sposobu implementacji usług w konkretnym systemie) jak również ze względu na to, że termin usługi zbiorów oznacza usługi dostarczane przez wykonanie funkcji protokołu FTAM (w oparciu o usługi warstw niższych), a nie usługi systemu przechowywania informacji lub dowolnego rzeczywistego podsystemu posługującego się tym protokołem.

2. KONCEPCJA USŁUG ZBIORÓW

Jak wspomniano we wstępie, usługi zbiorów są abstrakcją podwarstwy warstwy aplikacyjnej, w której elementy warstwy współpracują za pośrednictwem protokołu FTAM. Konwencja ISO zakłada warstwowe przedstawianie usług. Usługi zbiorów są usługami pośrednika pomiędzy jednym z komunikujących się za ich pośrednictwem użytkowników, w praktyce będącym użytkownikiem pamięci zbiorów, zwanych dalej kontrolerem (ang. initiator), a drugim z użytkowników, w praktyce oferującym usługi pamięci zbiorów, zwanym dalej operatorem (ang. responder). Zadaniem operatora jest symulowanie pewnej standardowej pamięci zbiorów, zwanej wirtualną pamięcią zbiorów, z wykorzystaniem rzeczywistej pamięci zbiorów. Oczywiście sposób przechowywania informacji w rzeczywistej pamięci zbiorów jest zazwyczaj inny niż w wirtualnej, wobec czego do operatora będzie należeć funkcja odwzorowania pomiędzy pamięciami. Ta funkcja operatora jest wykonywana lokalnie i nie jest widoczna przy rozważaniu komunikacji systemów. Możliwości i sposób korzystania z wirtualnej pamięci zbiorów opisane są poprzez abstrakcyjny opis danych w tej pamięci, możliwych operacjach na danych oraz sposobu komunikacji z tą pamięcią poprzez sieć.

Opis usług bazuje na standardowym modelu połączenia dwóch użytkowników tej samej warstwy architektury sieci według modelu OSI (Open Systems Interconnection) za pośrednictwem usługodawcy, tj. warstwy bezpośrednio niższej (i wszystkich pod nią). W przypadku usług zbiorów tą warstwą jest warstwa prezentacji. Usługi również wykorzystują wcześniej założone połączenie przez podwarstwę aplikacyjną reprezentującą usługi wspólnych elementów warstwy aplikacji (CASE).

Ze względu na asymetrię ról użytkowników usług zbiorów, również usługi są niesymetryczne. Manifestuje się to w prawach użytkowników do

żądania usług - prawo wywołania większości z nich ma kontroler jako użytkownik pamięci symulowanej przez operatora. Kontroler definiuje czynności do wykonania na wirtualnej pamięci zbiorów. Operator wykonuje czynności zlecane mu przez kontrolera oraz raportuje o swoich czynnościach kontrolerowi. W usługach występuje też inna asymetria związana z jednokierunkowym przepływem danych podczas transferu zawartości zbioru. Kierunek transferu może ulegać zmianom, stąd jedną ze stron nazywa się nadawcą a drugą odbiorcą, niezależnie od ról kontrolera i operatora.

3. WIRTUALNA PAMIĘĆ ZBIORÓW

Sposób w jaki pamięć zbiorów jest implementowana zmienia się w zależności od istniejącego systemu rzeczywistego. Systemy te posiadają szeroki wachlarz stylów opisu pamiętanych informacji (danych) oraz metod umożliwiających dostęp do nich. Dlatego wymagany jest wspólny model opisu zbiorów i ich atrybutów. Z oczywistych względów, model taki powinien być ustanowiony zanim zdefiniuje się usługi, protokoły i procedury dla transferu zbiorów, dostępu i zarządzania w "sposób otwarty". Model abstrakcyjny opisujący pamięć zbiorów systemu otwartego nazwano wirtualną pamięcią zbiorów.

Wirtualna pamięć zbiorów ma nie tylko zapewnić współpracę partnerów w różnych systemach, ale również oddzielić użytkownika od różnorodności szczegółów związanych z rzeczywistymi systemami zbiorów oraz pozwolić na współpracę "prostych" i "rozbudowanych" systemów komputerowych ze sobą. Wiąże się to z pewną konwencją wirtualnej pamięci pozwalającą na określenie opcjonalnych podzbiorów w jej definicji, pozwalającą na komunikację pomiędzy systemami o różnym stopniu złożoności. Dzięki temu możliwe jest również połączenie z programem napisanym przez użytkownika sieci zamiast z wirtualną pamięcią.

3.1. Podstawowe własności modelu wirtualnej pamięci zbiorów

Każda jednostka aplikacyjna mogąca występować w roli operatora w usługach zbiorów jest uważana za jednostkę utrzymującą wirtualną pamięć zbiorów. Unikalny identyfikator tej jednostki może służyć za identyfikator pamięci zbiorów.

Wirtualna pamięć zbiorów zawiera dowolną liczbę zbiorów. W modelu wirtualnej pamięci zbiorów, zbiór jest obiektem posiadającym:

- nazwę, pozwalającą go jednoznacznie identyfikować,
 - atrybuty wyrażające jego własności (wielkość, historie itp.) oraz opisujące zawartość zbioru (struktura, typ danych itp.),
- oraz

- jednostki dostępu do zbioru tworzące jego zawartość.

Wszystkie powyższe własności zbioru mogą być widziane przez dowolnego uprawnionego użytkownika. Atrybuty opisujące zbiór są globalne, Oznacza to, że ta sama wartość atrybutu jest widziana przez wszystkich użytkowników. Kolejni użytkownicy również otrzymają tę samą wartość atrybutów zbioru pod warunkiem, że w międzyczasie nie zostały wprowadzone modyfikacje.

W modelu wyróżnia się atrybuty aktywności, opisujące związek pomiędzy użytkownikiem a zbiorem. Atrybuty te określają własności użytkownika oraz jego działania, tzw. aktywności. Dla każdej aktywności w pamięci zbiorów istnieje oddzielny zestaw wartości tych atrybutów, widziany wyłącznie przez użytkownika tworzącego aktywność. Zestaw atrybutów aktywności jest tworzony w momencie ustanowienia reżymu usług zbiorów (zobacz rozdział 4) i kasowany łącznie z jego uchyleniem.

Atrybuty aktywności są dwójakiego rodzaju: jeden jest związany z atrybutami zbioru i pokazuje ich aktywne wartości dostrzegane przez kontrolera, drugi odzwierciedla stan aktywności oraz opisuje kontrolera.

W wirtualnej pamięci zbiorów może być wykreowana dowolna liczba aktywności w danym czasie. Każda aktywność związana jest z jednym połączeniem oraz w danym czasie może być związana tylko z jednym zbiorem.

Na definicję wirtualnej pamięci zbiorów składają się:

1. opis modelu oraz sposób opisu własności zbioru
za pomocą atrybutów (elementy danych o ustalonym znaczeniu i zestawie wartości, które mogą przyjmować),
2. definicja zestawu działań (akcji) na elementach modelu,
3. definicja atrybutów oraz
4. definicja reprezentacji struktury zbioru.

Należy podkreślić, że definicja dotyczy pojedynczej, wirtualnej pamięci zbiorów i nie zawiera ona własności modelujących systemy baz danych.

3.2. Atrybuty zbioru

Do atrybutów zbioru należą następujące atrybuty: nazwa zbioru, możliwy tryb dostępu, kontrola dostępu (określa uprawnienia potrzebne do uzyskania dostępu), konto rozliczeniowe (za przechowanie), data i czas utworzenia zbioru, data i czas ostatniej modyfikacji zbioru, data i czas ostatniego czytania zbioru, identyfikacja tworzącego zbiór, identyfikacja ostatniego modyfikatora zbioru, identyfikacja ostatniego czytelnika zbioru, dostępność zbioru (natychmiastowy, opóźniony), kon-

tekst prezentacji (standardowe zbiory, typy danych, kontekst definiowany przez użytkownika, zbiór typów danych), identyfikator aktywności tworzącej zbiór, nazwa algorytmu szyfrującego, typ struktury dostępu, nazwa struktury prezentacji, wielkość zbioru, dopuszczalna wielkość zbioru, zastrzeżenia prawne.

Dla każdego atrybutu ustalono zbiór wartości jakie atrybuty mogą przyjmować. Wartości te nie będą tutaj omawiane.

3.3. Atrybuty aktywności

Atrybuty aktywności odzwierciedlają stan bieżącego połączenia. Dla każdego połączenia istnieje zbiór wartości atrybutów aktywności. Wartości atrybutów aktywności odzwierciedlają stan dostępu do pamięci zbiorów.

Atrybutami aktywności są: żądany dostęp (o wartościach read, insert, replace, erase, extend, read attribute, change attribute, delete file), autoryzacja kontrolera (autoryzacja ustanawiania połączenia), hasło (do żądania wyboru atrybutu dostępu), adres jednostki usług zbiorów po stronie kontrolera (adres punktu dostępu do usług prezentacyjnych), bieżący typ struktury dostępu (bezpostaciowa, jednorodna i drzewiasta), bieżący kontekst prezentacji (wynegocjowany kontekst dla transferu zbiorów), sterowanie współużywalnością (dostęp dzielony, dostęp wyłączny) oraz bieżący kontekst dostępu.

3.4. Grupy atrybutów

Nie wszystkie atrybuty podane w poprzednim podrozdziale są potrzebne w każdych okolicznościach. Możliwość uproszczonych implementacji manifestuje się również w podziale atrybutów na grupy, które jeśli są implementowane, to muszą być implementowane w całości. Dlatego zaproponowano podział atrybutów na trzy grupy: grupę podstawową, grupę pamięci oraz grupę ochrony.

Grupa podstawowa jest zawsze definiowana i musi być dostępna dla dowolnej specyfikacji związanej z wirtualną pamięcią. Grupę tą tworzą:

Atrybuty zbioru - nazwa, kontekst prezentacji, typ struktury dostępu, nazwa struktury prezentacji oraz rozmiar zbioru.

Atrybuty aktywności - żądany dostęp, adres jednostki usług po stronie kontrolera, bieżący typ struktury dostępu oraz kontekst prezentacji.

Wyjątkowo w tej grupie wszystkie wartości atrybutów muszą być dostępne.

Grupa pamięci posiada znaczenie, jeśli operator gwarantuje przechowanie informacji i pozwala aktywności odnieść się do informacji ustanowionych przez poprzednie aktywności. Grupę tą tworzą:

- atrybuty zbioru - konto rozliczeniowe, data i czas utworzenia,
 data i czas ostatniej modyfikacji, identy-
 fikator aktywności tworzącej zbiór, dostęp-
 ność zbioru, typ struktury dostępu, dopuszczal-
 na wielkość zbioru,
 atrybuty aktywności - bieżący kontekst dostępu, sterowanie współ-
 używalnością.

Grupę ochrony tworzą pozostałe atrybuty nie wymienione w dwóch poprzednich grupach.

3.5. Odwołanie się do zbioru

Odwołanie się do zbioru odbywa się w dwóch krokach. W kroku pierwszym, ustanawia się połączenie z pamięcią zbiorów zawierającą wymagany zbiór. W kroku drugim, stacji obsługującej daną pamięć zbiorów, muszą być udostępnione informacje pozwalające zidentyfikować wymagany zbiór, spośród innych zawartych w danej pamięci zbiorów. Specyfikacja zbioru odbywa się poprzez podanie nazwy pamięci zbiorów oraz nazwy zbioru.

3.6. Struktura zbiorów

Zbiór składa się z jednej bądź wielu identyfikowalnych jednostek danych (JD). Model wirtualnej pamięci zbiorów dostarcza struktury drzewiastej, odzwierciedlającej związki zachodzące pomiędzy JD ze względu na dostęp do nich oraz ich identyfikację. Stąd, strukturę tą nazwano strukturą dostępu. Poddzewa struktury dostępu tworzą tzw. jednostki dostępu do zbioru (JDZ), które są obiektami danych odpowiedniego typu. JDZ są jednostkami, za pomocą których dokonuje się operacji na zawartości zbioru.

- W standardzie wyróżniono dwa specjalne przypadki struktury dostępu
- bezpostaciową, oraz
 - jednorodną.

Bezpostaciowa struktura dostępu składa się z jednej JDZ oraz jednej JD. Jednorodna struktura dostępu składa się z jednej JDZ oraz wielu JD, zazwyczaj tego samego typu.

Oprócz struktury dostępu wyróżnia się strukturę prezentacji. Każda jednostka danych jest obiektem danych określonego typu. Składa się z niepodzielnych porcji nazywanych elementami danych, dla których zdefiniowana jest indywidualna syntaktyka abstrakcyjna. Informacje o syntaktykach tworzą strukturę prezentacji zbioru.

4. USŁUGI ZBIORÓW

4.1. Usługi a protokoły

Definicja usług zbiorów określa szereg wchodzących w ich skład usług elementarnych, które, w zależności od celu zastosowania, wykorzystywane są w sekwencjach, dających w sumie kompletną usługę o znaczeniu praktycznym. (Termin usługa elementarna wprowadzono tu tylko po to, aby podkreślić różnice pomiędzy pojedynczą usługą zbiorów, nazywaną elementarną, a ich sumą, tj. usługami zbiorów).

Zgodnie z warstwową konwencją modelu OSI usługi zbiorów przedstawia się jako usługi warstwy za pomocą:

- prymitywnych akcji i zdarzeń usług, rozumianych jako niepołączalne i rozłączne w czasie interakcje pomiędzy użytkownikiem usług a usługodawcą, zachodzące z inicjatywy użytkownika lub usługodawcy,
- parametrycznych danych stowarzyszonych z każdą prymitywną akcją lub zdarzeniem,
- związków występujących pomiędzy prymitywnymi akcjami i zdarzeniami oraz ich prawidłowymi sekwencjami.

Te prymitywne akcje i zdarzenia będziemy dalej nazywali prymitywami usług. Prymitywem usług jest więc zarówno akcja użytkownika żądającego usługi elementarnej, jak i zdarzenie usług, tj. powiadomienie użytkownika o wykonaniu usługi lub o akcji jego partnera (drugiego współużytkownika połączenia).

Rozróżnia się dwa typy usługi elementarnej: niepotwierdzaną oraz potwierdzaną. W usłudze niepotwierdzanej jeden z użytkowników żąda usługi od dostawcy, który następnie sygnalizuje drugiemu z użytkowników wywołanie danej usługi przez pierwszego, równocześnie przekazując wartości parametrów. W usłudze potwierdzanej występuje dodatkowo potwierdzenie, umożliwiające przepływ informacji w odwrotnym kierunku. Wykonywane jest ono w następstwie udzielenia odpowiedzi na sygnalizację, jaką dostawca usług przekazuje użytkownikowi żądającemu usługi elementarnej. Otrzymanie odpowiedzi zawsze oznacza zakończenie realizacji usługi elementarnej (z pozytywnym lub negatywnym skutkiem), jednakże może oznaczać zarówno rozpoczęcie jak i zakończenie akcji partnera, żądanej za pomocą tej usługi elementarnej.

W usługach zbiorów żądanie usługi związane jest z przesłaniem odpowiedniego, pojedynczego komunikatu protokołu FTAM przenoszącego parametry prymitywu żądającego usługi. To samo dotyczy potwierdzenia usługi.

4.2. Rodzaje usług

W skład usług zbiorów wchodzi następujące usługi elementarne, pogrupowane w zależności od zastosowania:

Kontrolowanie reżymu usług zbiorów

- ustanowienie reżymu usług zbiorów,
- zakończenie reżymu usług zbiorów,
- zlikwidowanie reżymu usług zbiorów.

Kontrolowanie reżymu przydziału zbioru

- wybór zbioru,
- zwolnienie zbioru,
- utworzenie zbioru,
- usunięcie zbioru.

Zarządzanie zbiorami

- czytanie atrybutów zbioru,
- zmiana atrybutów zbioru.

Kontrolowanie reżymu otwarcia zbioru

- otwarcie zbioru,
- zamknięcie zbioru.

Kontrolowanie łączenia

- początek grupowania,
- koniec grupowania.

Odtwarzanie

- odtwarzanie reżymu.

Dostęp do zawartości zbioru

- lokalizacja jednostki danych dostępu do zbioru,
- wymazanie jednostki danych dostępu do zbioru.

Transfer porcji danych

- czytanie porcji danych,
- pisanie porcji danych,
- transfer jednostki danych,
- kończenie transferu danych,
- kończenie transferu,
- anulowanie transferu danych.

Rozdzielenie i ponowny start

- rozdzielenie,
- ponowny start transferu danych.

Znaczenia poszczególnych usług są podane w następnym podrozdziale.

4.3. Ogólne zasady korzystania z usług zbiorów

Wyróżnia się dwa poziomy usługi zbiorów: rzetelny oraz korygowany przez użytkownika. Poziom korygowany przez użytkownika jest tu opisywany, natomiast poziom rzetelny osiągany jest przez wprowadzenie pomiędzy użytkownika a dostarczyciela usług poziomu korektora błędów i sytuacji awaryjnych. Korektor ten wykorzystuje usługi odtwarzania, rozdzielania i ponownego startu w celu zamaskowania przed użytkownikiem naprawialnych błędów i awarii.

Klasą usług zbiorów jest określona grupa usług elementarnych wybrana do celu: transferu zbioru, dostępu do zbioru lub zarządzania zbiorami. Każda z klas zawiera obowiązkowe oraz opcjonalne zespoły funkcjonalne usług. Przykładowo, klasa transferu zbioru zawiera obowiązkowo zespoły funkcjonalne jądra, czytania albo pisania i grupowania. Do opcjonalnych należą wszystkie pozostałe z wyjątkiem zespołu dostępu do zbioru.

Korzystanie z usług zbiorów może odbywać się w jednym z czterech reżymów pracy, wyznaczających podzbiór możliwych do wykorzystania usług elementarnych.

Podstawowym reżymem jest reżym usług zbiorów, związany z przypisaniem do połączenia pomiędzy partnerami cech wymaganych dla stosowania protokołu FTAM oraz wzajemną identyfikacją partnerów. Pozostałe reżymy są kolejno zagnieżdżane w poprzednich. Są to reżymy: przydzielonego zbioru, otwartego zbioru oraz transferu danych. Każdy z nich może być ustanowiony i uchylony wielokrotnie, bez uchYLENIA reżymu, w którym jest zagnieżdżony. Przykładowe następstwo zdarzeń jest zilustrowane poniżej za pomocą nawiasów oznaczających ustanowienie i uchylenie reżymów:

(r.usług zb.-(r.przydzielonego zb. - operacje na atrybutach zb. -
- (r. otwarcia zb. - (r.transferu danych - r. transferu danych) -
- r. otwarcia zb.) - r. przydzielonego zb.) - (r. przydzielonego
zb. - r. przydzielonego zb.) - r.usług zb.)

W każdym z reżymów można stosować wyłącznie właściwe mu funkcje usług zbiorów. Ustanowienie i uchylenie reżymów odbywa się przy okazji pomyślnej realizacji niektórych usług elementarnych (nie ma oddzielnych funkcji ustanawiania i uchylania reżymu). W tym celu wykorzystuje się następujące usługi:

Ustanowienie reżymu

usług zbiorów - usługa ustanowienia reżymu,
przydzielonego zbioru - usługa wyboru zbioru, utworzenia zbioru,
otwartego zbioru - usługa otwarcia zbioru,

transferu danych - usługa czytania (lub pisanie) jednostki danych
dostępu do zbioru.

Uchylenie reżymu

usług zbiorów - usługa zlikwidowania lub uchylenia reżymu usług
zbiorów,

przydzielonego zbioru - usługa zwolnienia lub usunięcia zbioru,

otwartego zbioru - usługa zamknięcia zbioru,

transferu danych - usługa końca transferu.

W trakcie ustanawiania reżymu usług zbiorów odbywa się negocjacja parametrów połączenia, kontekstów prezentacyjnych (dozwolonych syntaktyk przesyłanych danych), poziomu, klasy i opcjonalnych funkcji usług zbiorów.

Próba ustanowienia reżymu usług może się nie powieść (zostanie odrzucona) zarówno z powodu niezgodności partnerów w poziomie lub klasie usług.

Negocjacja opcjonalnych zespołów funkcjonalnych wewnątrz danej klasy polega na przedstawieniu przez kontrolera listy wymaganych zespołów. Operator może skreślić z listy te zespoły funkcjonalne, których nie jest on w stanie zaakceptować.

Wymiana danych w trakcie negocjacji odbywa się poprzez dołączenie odpowiednich parametrów do usługi ustanowienia reżymu usług zbiorów, która jest usługą potwierdzaną, a więc następuje przesłanie informacji tam i z powrotem. Dodatkowe dane, precyzujące znaczenie poszczególnych usług są przekazywane również jako ich parametry. Wartości większości z tych parametrów nie są modyfikowane przez dostarczyciela usług.

4.4. Definicje usług

4.4.1. Zespół funkcjonalny jądra

Usługa ustanowienia reżymu usług zbiorów

Służy do wprowadzenia reżymu usług zbiorów i powiązania go z połączeniem pomiędzy dwoma użytkownikami usług zbiorów. Jest to usługa potwierdzana, żądana przez kontrolera. Parametry usługi służą identyfikacji obu parametrów oraz inicjującego użytkownika, ustaleniu parametrów usług (poziom i klasa usług, rodzaj danych itp.).

Usługa zakończenia reżymu usług zbiorów

Służy do uchylenia reżymu usług zbiorów i jego powiązania z połączeniem pomiędzy dwoma użytkownikami usług. Jest to usługa potwierdzana, żądana przez kontrolera wyłącznie wtedy, gdy nie wykonuje on żadnej akcji za pośrednictwem operatora (nie jest ustanowiony reżym przydzielonego zbioru). Parametr usługi przekazuje koszt zrealizowanych działań.

Usługa zlikwidowania reżymu usług zbiorów

Służy do bezwarunkowego uchylecia reżymu usług zbiorów i jego powiązania z połączeniem pomiędzy dwoma partnerami. Jest to usługa niepotwierdzana, żądana przez dowolnego z partnerów lub usługodawcę w dowolnym momencie. Jej wykonanie powoduje zaniechanie wszystkich akcji w pamięci zbiorów operatora, pozostawiając przydzielony zbiór w nieokreślonym stanie. Parametr usługi przekazuje przyczynę zlikwidowania.

Usługa wyboru zbioru

Służy do wybrania zbioru spośród istniejących zbiorów oraz złączenia go z reżymem usług zbiorów. Ustanawia ona reżym przydzielonego zbioru. Jest to usługa potwierdzana, żądana przez kontrolera wyłącznie wtedy, gdy inny zbiór nie jest przydzielony. Parametry usługi służą do wyboru zbioru, ustalenia jego atrybutów, określenia sposobu korzystania ze zbioru (sposób dostępu, kontrola współużytkowania i zaawansowania).

Usługa zwolnienia zbioru

Służy do odłączenia zbioru od reżymu usług zbiorów. Uchyla ona reżym przydziału zbioru. Jest to usługa potwierdzana, żądana przez kontrolera wyłącznie wtedy, gdy zbiór jest przydzielony. Po zwolnieniu zbiór istnieje i jest dostępny dla następnej akcji wyboru zbioru. Parametry przekazują rezultat oraz ewentualne przyczyny niepowodzenia.

4.4.2. Zespół funkcjonalny czytania

Usługa czytania porcji danych

Służy do zainicjowania transferu danych od operatora do kontrolera. Jest to usługa niepotwierdzana, żądana przez kontrolera wyłącznie w reżymie otwarcia zbioru i stanie "jałowym" transferu danych. Ustalony przez nią kierunek przepływu danych jest obowiązujący aż do wywołania usługi końca transferu danych. Usługa czytania porcji danych może być odrzucone za pomocą usługi końca danych z parametrem przyczyny istotniejszym niż ostrzeżenie. Parametry usługi przekazują specyfikację transferu porcji danych (identyfikacja jednostki danych dostępu do zbioru, żądany dostęp, kontrola współużytkowania).

Usługa transferu jednostki danych

Służy do przesyłania danych. Jest to usługa niepotwierdzana, żądana przez nadawcę (partner wskazany jako nadawca w danym reżymie transferu danych). Parametry usługi przekazują typ danych oraz ich wartość.

Usługa końca transferu danych

Służy do wskazania zakończenia transferu danych. Jest to usługa niepotwierdzana, żądana przez nadawcę po nadaniu wszystkich danych.

Parametry przynoszą informację diagnostyczną.

Usługa końca transferu

Służy do zatwierdzenia zakończenia transferu danych. Jest to usługa potwierdzana, żądana przez kontrolera po zakończeniu transferu danych (wykonaniu usługi końca transferu danych). Uchyla ona reżym transferu danych. Parametry przynoszą informację diagnostyczną oraz informację o stanie zaawansowania.

Usługa anulacji transferu danych

Służy do unieważnienia transferu danych, tj. skasowania aktywności transferu danych. Jest to usługa potwierdzana przerywająca bieżącą aktywność, żądana przez dowolnego z partnerów w reżymie transferu danych (nadawca nie może jej stosować po zażądaniu usługi końca transferu danych). Parametry przynoszą informację diagnostyczną.

Usługa otwarcia zbioru

Służy do ustanowienia kontekstu prezentacji oraz kontroli współużytkownika i zaawansowania dla aktywności transferu danych. Jest to usługa potwierdzana żądana przez kontrolera w reżymie przydziału zbioru. Usługa ustanawia reżym otwarcia zbioru. Parametry usługi służą do ustalenia trybu przetwarzania, nazwy typu zawartości, kontroli współużytkownika i zaawansowania, trybu odtwarzania, identyfikatora aktywności. Zwrotnie przekazują rezultat oraz informację diagnostyczną.

Usługa zamknięcia zbioru

Służy do uchylenia reżymu otwarcia zbioru. Jest to usługa potwierdzana, żądana przez kontrolera. Parametry służą do ustalenia kontroli zaawansowania, oraz zwrotnie przekazują rezultat oraz informację diagnostyczną.

4.4.3. Zespół funkcjonalny pisanie

Usługa pisanie porcji danych

Służy do zainicjowania transferu danych od kontrolera do operatora. W pozostałych aspektach usługa jest analogiczna do usługi czytania porcji danych.

Pozostałe usługi zespołu pisanie

W skład zespołu wchodzi wszystkie usługi zespołu funkcjonalnego czytania z wyjątkiem usługi czytania porcji danych.

4.4.4. Zespół funkcjonalny dostępu do zbioru

Usługa lokalizacji jednostki danych dostępu do zbioru

Służy do wskazania konkretnej jednostki danych dostępu do zbioru, która ma zostać zlokalizowana w celu udostępnienia jej dla późniejszych operacji. Jest to usługa potwierdzana, żądana przez kontrolera.

Parametry służą do ustalenia konkretnej jednostki oraz kontroli współużytkowania oraz zwrotnie przenoszą rezultat wraz z informacją diagnostyczną.

Usługa wymazania jednostki danych dostępu do zbioru

Służy do usunięcia jednostki danych dostępu do zbioru ze zbioru. Jest to usługa potwierdzana, żądana przez kontrolera. Parametry usługi służą do wskazania konkretnej jednostki, kontroli współużytkowania oraz zwrotnie przenoszą rezultat wraz z informacją diagnostyczną.

4.4.5. Zespół funkcjonalny ograniczonego zarządzania zbiorami

Usługa utworzenia zbioru

Służy do wykreowania zbioru i wybrania tego nowo utworzonego zbioru jako zbioru przydzielonego (usługa ustanawia reżym przydzielonego zbioru). Jeśli zbiór o podanej identyfikacji istnieje, to w zależności od wartości parametrów może być przydzielony lub nie - w tym przypadku usługa zakończy się nieprawidłowo. Jest to usługa potwierdzana, żądana przez kontrolera. Pozostałe parametry przekazują atrybuty zbioru, hasła dostępu, kontrolę współużytkowania i zaawansowania. Zwrotnie przekazywany jest rezultat wykonania usługi.

Usługa usunięcia zbioru

Służy do uchylenia reżymu otwarcia z jednoczesnym skasowaniem przydzielonego zbioru (zbiór przestaje istnieć). Jest to usługa potwierdzana, żądana przez kontrolera wyłącznie w reżymie przydzielonego zbioru. Parametry przekazują hasło oraz zwrotnie rezultat, koszt i informację diagnostyczną.

Usługa czytania atrybutów zbioru

Służy do odczytania wartości atrybutów zbioru. Jest to usługa potwierdzana, żądana przez kontrolera w reżymie przydzielonego zbioru. Parametry przekazują listę atrybutów oraz zwrotnie ich wartości, rezultat oraz informację diagnostyczną.

4.4.6. Zespół funkcjonalny rozszerzonego zarządzania zbiorami

Usługa zmiany atrybutów zbioru

Służy do zmodyfikowania wartości atrybutów zbioru. Parametry przekazują listę atrybutów i ich nowych wartości. Zwrotnie rezultat i informację diagnostyczną. W skład zespołu wchodzi także wszystkie usługi zespołu ograniczonego zarządzania zbiorami.

4.4.7. Zespół funkcjonalny grupowania

Usługa początku grupowania

Służy do zaznaczenia początku sekwencji połączonych usług, które mają być wykonane jako jedna całość. Jest to usługa potwierdzana żada-

na przez kontrolera.

Usługa końca grupowania

Służby do zaznaczenia końca sekwencji połączonych usług, które mają być wykonane jako całość. Jest to usługa potwierdzana, żądana przez kontrolera.

4.4.8. Zespół funkcjonalny odtwarzania

Usługa odtwarzania reżymu

Służby do ponownego założenia reżymu otwartego zbioru po awarii. Odtworzenie poprzedniego reżymu następuje poprzez powołanie się na identyfikator byłej aktywności. Jest to usługa potwierdzana, żądana przez kontrolera. Parametry przekazują identyfikację byłej aktywności, numer porcji danych, żądany dostęp, hasło i punkt odtworzenia, zwrotne rezultaty i informację diagnostyczną.

Usługa rozdzielania

Służby do wprowadzania znaczników punktów przerywania strumienia danych do celów późniejszego odtworzenia lub ponownego startu. Jest to usługa potwierdzana, żądana przez nadawcę.

Pozostałe usługi zespołu

W skład zespołu wchodzi również usługa anulacji transferu danych.

4.4.9. Zespół ponownego startu transferu danych

Usługa ponownego startu transferu danych

Służby do przerywania bieżącego transferu i wynegocjowania miejsca (znacznika), od którego powinien być ponownie wystartowany. Jest to usługa potwierdzana, żądana przez dowolnego z partnerów w reżymie transferu danych. Parametry przenoszą identyfikator znacznika oraz informację diagnostyczną.

W skład zespołu wchodzi również usługi rozdzielania i anulacji transferu danych.

5. PODSUMOWANIE

W pracy przedstawiono koncepcję usług zbiorów, opisano model wirtualnej pamięci zbiorów oraz dosyć szczegółowo przedstawiono usługi transferu, dostępu do i zarządzania zbiorami w systemie otwartym.

Materiał zaprezentowany w pracy zaczerpnięto z dokumentów ISO [1-4], które aktualnie posiadają status szkicu standardu.

Wydaje się, że propozycja dokumentu w zasadniczej, koncepcyjnej części, nie ulegnie zmianie, a zatem może w pewnym zakresie stanowić bazę implementacyjną usług FTAM.

W sieci MSK zaimplementowano stacje transferu zbiorów, których oprogramowanie oparto o zalecenia podane w [6]. propozycje ewentual-

nego wykorzystania stacji do świadczenia usług tutaj prezentowanych podano w [5].

LITERATURA

- [1] Information Processing Systems. Open Systems Interconnection, File Transfer, Access and Management, Part 1: General Description ISO DP8571 1.
- [2] Information Processing Systems, Open Systems Interconnection, File Transfer, Access and Management, Part 2: The virtual filestor ISO DP8571 2.
- [3] Information Processing Systems, Open Systems Interconnection, File Transfer, Access and Management, Part 3: File service definition ISO DP8571 3.
- [4] Information Processing Systems, Open Systems Interconnection, File Transfer, Access and Management, Part 4: File protocol specification, ISO DP8571 4.
- [5] Petersen N.K., Skovgaard T., Anticipating the ISO File Transfer Standards in an Open Systems Implementation, Computer Networks and ISDN Systems, 9 1985 267-280.
- [6] A Network Independent File Transfer Protocol, File Transfer Protocol Implementers Group, NIFTP-B(80), 5th Feb. 1981 ("Blue Book").

FILE TRANSFER, ACCESS and MANAGEMENT FOR OPEN SYSTEMS INTERCONNECTION

One of the services in computer network standardized by International Organization for Standardization (ISO) is File Transfer, Access and Management. The paper presents the FTAM services in Open Systems Architecture and details allowing general orientation in FTAM services standardized by ISO.

ПЕРЕДАЧА ДОСТУП И УПРАВЛЕНИЕ НАБОРАМИ ДАННЫХ В МОДЕЛИ ВЗАИМОДЕЙСТВИЯ ОТКРЫТЫХ СИСТЕМ

Одной из услуг предоставляемых вычислительной сетью является услуга передачи, доступа и управления данными стандартизованная Международной Организацией Стандардов (МОС). В работе предлагается концепция услуг для наборов данных в открытых системах, а также подробные сведения, касающиеся диапазона услуг подлежащих стандартизации.

Wojciech MOLISZ^x

systemy przekazywania
wiadomości (Message
Handling Systems)
Zalecenia CCITT serii
X.400 poczta elektro-
niczna

TELEINFORMATYCZNE SYSTEMY PRZEKAZYWANIA WIADOMOŚCI (MHS)

Przedstawiono model funkcjonalny, architekturę logiczną i usługi teleinformatycznych systemów przekazywania wiadomości (SPW) (Message Handling Systems), umożliwiających wymianę i przetwarzanie wiadomości o różnych postaciach (usługi telematyczne) poprzez sieci teleinformatyczne. Model jest zgodny z zaleceniami serii X.400 CCITT, a także z modelem ISO OSI. Usługi i protokoły SPW mieszczą się w warstwie zastosowań OSI, ale normalizacja obejmuje również pewne elementy warstw niższych. Przedstawiono zagadnienia implementacji SPW w rozległych sieciach komputerowych (RSK) oraz zagadnienia dołączania komputerów osobistych do SPW.

1. WSTĘP

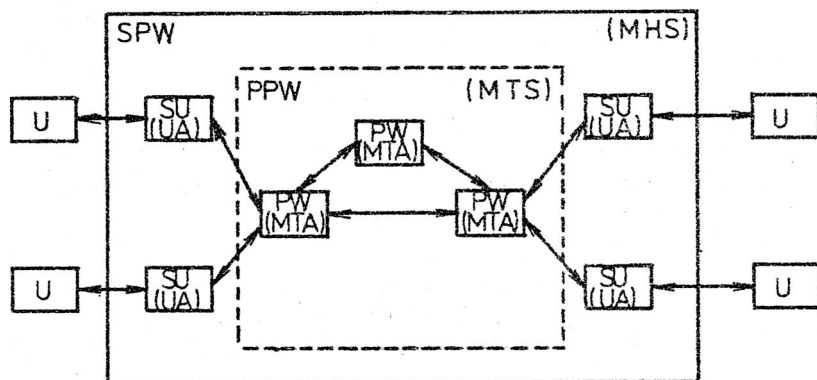
Nowe odkrycia i wprowadzane szeroko nowe technologie w elektronice, informatyce i telekomunikacji spowodowały w ostatnich latach nie notowane wcześniej przyspieszenie rozwoju usług i sieci telekomunikacyjnych. Silne stało się też dążenie do integracji usług w całkowicie cyfrowej sieci telekomunikacyjnej (ISDN) ¹. Trwają intensywne prace nad wprowadzeniem "inteligentnych" usług i systemów (ang.: Value-added s.), tzn. systemów oferujących dodatkowe usługi w stosunku do "czystych", klasycznych usług telekomunikacyjnych. Jednym z typów takich systemów są teleinformatyczne systemy przekazywania wiadomości (dalej w skrócie SPW) ang.: Message Handling Systems-MHS), wdrażane obecnie w ramach działających sieci teleinformatycznych. Początkowo SPW przyjęły postać tzw. poczty elektronicznej (Electronic Mail), instalowanej w rozległych sieciach komputerowych (RSK) w USA i Kanadzie. Szybko okazało się, że objęcie usługą poczty elektronicznej użytkowników więcej niż jednej sieci wymaga wprowadzenia nowych standardów. CCITT zareagowała bardzo szybko i kompleksowo, opracowując nową serię zaleceń: X.400, aktualnie rozwijanych (3 do 10) i uwzględniających szeroko tzw. usługi telematyczne. W międzyczasie

^x Instytut Telekomunikacji, Zakład Teorii Systemów Informatycznych, Politechnika Gdańska

sie SPW w ramach RSK bardzo zwiększyły swoje możliwości i zaczęto tworzyć SPW w ramach lokalnych sieci komputerowych (LSK). Świat stoi też w przededniu zapewnienia dostępu użytkownikom komputerów osobistych do SPW, co może mieć daleko idące następstwa. Ze względu na duże znaczenie SPW, wynikające z szerokiego zakresu zastosowań i intensywne rozwijanie ich za granicą należy się spodziewać wprowadzenia SPW również w naszym kraju. W pracy podjęto próbę omówienia problematyki SPW na tle modelu CCITT. Zalecenia CCITT wprowadziły nową terminologię dla której autor zaproponował odpowiedniki polskie. Ze względu na nowatorski charakter tej propozycji przy każdym nowym terminie polskim podano w nawiasie odpowiednik angielski. Dla ilustracji modelu SPW przedstawiono usługi istniejącego SPW, a następnie omówiono zagadnienia rozwoju SPW w ramach RSK oraz zagadnienia dołączania komputerów osobistych do SPW.

2. MODEL SYSTEMU PRZEKAZYWANIA WIADOMOŚCI.

Całokształt zagadnień SPW został opracowany przez CCITT i wydany w postaci Zaleceń X.400 - X.430 3 do 10. Model SPW, który teraz krótko opiszemy jest zgodny z tymi zaleceniami. Struktura funkcjonalna modelu SPW jest przedstawiona na rys.1.

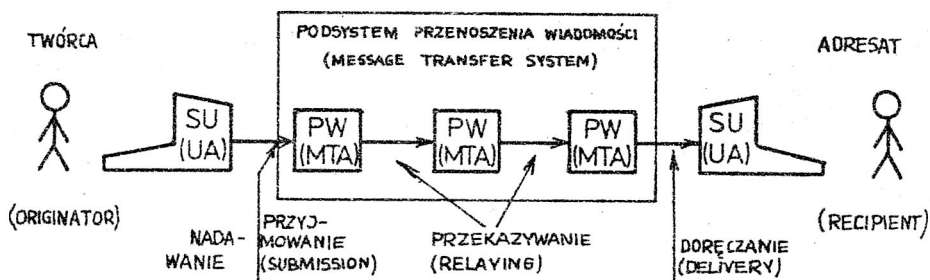


Rys.1. Model funkcjonalny systemu przekazywania wiadomości.

Fig.1. Functional View of the MHS Model.

Użytkownikiem SPW jest człowiek, bądź komputerowy proces zastosowań. Użytkownik jest twórcą wiadomości (originator), bądź jej adresatem (recipient). Wiadomość w kontekście SPW jest jednostką informacji, przenoszenia wiadomości (PPW) Message Transfer System-MTS). Wiadomość składa się z koperty i treści. Koperta zawiera informacje, wykorzystywane przy przekazywaniu wiadomości od twórcy do adresata. Koperty mogą się

zmieniać, o czym powiemy niżej. Stacja użytkownika (SU) User Agent-UA) pozwala mu na utworzenie wiadomości i współpracuje z FPW. Na FPW składa się sieć współpracujących ze sobą przekaźników wiadomości (PW)(Message Transfer Agents - MTAs), które przenoszą wiadomość, a następnie dostarczają ją stacji (bądź stacjom) adresata. W łańcuchu przekazywania wiadomości od twórcy do adresata, przykładowo przedstawionym na rys.2 wyróżnia się 3 typy oddziaływań: 1) twórca-przekaźnik (źródłowy), 2) przekaźnik-przekaźnik, 3) przekaźnik (docelowy) - adresat.



Rys.2. Przekazywanie wiadomości w SPW

Fig.2. Message Transfer within MHS

W zależności od punktu widzenia pierwsze oddziaływanie (twórca-przekaźnik) jest dla użytkownika nadawaniem, a dla FPW przyjmowaniem (submission). Dla tego oddziaływania stosuje się kopertę przyjmowania (submission envelope), która zawiera badania elementów obsługi FPW. Dalej, w obrębie FPW używana jest koperta przekazywania (relaying envelope), zawierająca informacje niezbędne dla FPW i opis elementów obsługi dla SU. Wreszcie trzecie oddziaływanie (przekaźnik-adresat) wymaga koperty doręczania (delivery envelope), która zawiera informacje niezbędne dla doręczenia wiadomości adresatowi.

Wyróżniono podklasę SPW, zwaną systemem osobistego porozumienia się (SOP) Interpersonal Messaging System-IPMS). Użytkownikami SOP są ludzie. SOP świadczy swoim użytkownikom usługi osobistego porozumiewania się i zapewnia dostęp do niektórych usług telematycznych. Do usług tych zalicza się telex, tekst (wymiana standaryzowanych dokumentów), teletext, faksymile, videotext, a także przekazywanie mowy, dźwięków i obrazów (video), w szczególności usługi video-konferencji. Ostatnie trzy nie występują w SOP. Obecnie SOP są dominującą podklasą SPW. SU obsługująca użytkownika SOP powinna umożliwiać:

1. wysyłanie (odbieranie) wiadomości do (od) PW
2. Przedstawianie wiadomości swojemu użytkownikowi
3. Współpracę z innymi SU
4. dodatkowe przetwarzanie wiadomości.

Opcjonalnie SE należąca do SOP może realizować lokalne funkcje nie objęte Zaleceniami CCITT, jak przetwarzanie tekstów, przechowywanie i wyszukiwanie wiadomości w bazie danych itp.

W treści wiadomości SOP wyróżnia się nagłówek (Header) i treść właściwą (Body). Przykładową zawartość nagłówka omówimy w p.6. Treść właściwa może zawierać zakodowane informacje różnych typów, jak tekst, faksymile, grafika, głos i inne. Protokół, określający typ treści jest wyspecyfikowany w zaleceniu X.420. W zakresie usług telematycznych dotychczas unormowano jedynie współpracę terminali teletekstu w ramach SOP (Zalecenie X.430). Videotekst jest uważany za interaktywny (tj. nie służy przekazywaniu wiadomości). Jednakże dopuszczono możliwość włączenia grafiki videotekstu do treści wiadomości. Konwersacja kodów wiadomości różnych typów jest objęta Zaleceniem X.408.

3. ZASADY TWORZENIA NAZW I ADRESÓW W SPW

Dla zapewnienia poprawnej pracy SPW każda SU może mieć unikalną nazwę i adres (użytkownicy SPW nie są elementami systemu i dlatego rozważania dotyczą ich stacji). Zalecenie X.400 szeroko omawia zagadnienie tworzenia nazw i adresów. Wzięto pod uwagę różne aspekty praktyczne, a w szczególności postulaty: aby zasady tworzenia nazw były przyjazne dla użytkownika, oraz aby SPW mogły wykorzystywać istniejące sposoby adresowania systemów na bazie których będą wdrażane. Dlatego dopuszczono kilka wariantów i pewną swobodę tworzenia nazw i adresów. Zarówno nazwy jak i adresy są typu opisowego i składają się z list atrybutów. Nazwy odnoszące się do stacji twórcy adresata są nazwami T A (O R Names). Wyróżniono 4 kategorie atrybutów:

1. Osobowe (Personel A.) (np.: nazwisko, imiona, inicjały itd)
2. Geograficzne (Geographical A.) (np.: nazwa ulicy, nr domu, miasto, kraj)
3. Organizacyjne (Organizational A.) (np.: nazwa instytucji, nazwa jednostki organizacyjnej, stanowisko)
4. Systemowe (Architectural A.) (np.: adres zgodny z Zaleceniem X.121 unikalny identyfikator SU)

Każdy SPW działa w pewnym obszarze geograficznym, zwanym obszarem administracyjnym (OA) Management Domain - MD). W zależności od tego, czy właścicielem SPW jest poczta, czy też inna instytucja mówimy o pocztowym (telekomunikacyjnym) obszarze administracyjnym (POA) Admi-

nistration Management Domain-ADMD), bądź o wydzielonym obszarze administracyjnym (WOA) (Private Management Domain-PRMD). Z uwagi na międzynarodowy zasięg działania SPW wiadomość może w drodze od twórcy do adresata przejść przez kilka OA. Aby poprawnie skierować wiadomość do OA, do którego należy adresat muszą być podane wszystkie atrybuty z tzw. bazowego zbioru atrybutów (Base Attribute Set). Zalecenie X.400 wyróżnia cztery kategorie atrybutów bazowych. Są to atrybuty:

1. Instytucji (Commercial) (nazwa instytucji, kraj)
2. Siedziby (Residential) (region, województwo itp., kraj)
3. Obszaru administracyjnego (Architectural) (nazwa OA, kraj)
4. Terminala (Terminal-oriented) (adres X.121, nr teleksowy, identyfikator terminala telematycznego).

Adresy T A muszą umożliwiać jednoznaczne określenie miejsca dołączenia SU adresata do PFW. Każdy adres T A jest nazwą T A, ale nie zawsze odwrotnie. Zalecenie podaje dwie postacie nazw T A, będących jednocześnie adresami T A. Postać 1 ma 3 warianty w nawiasach kwadratowych są opcjonalne.

Nazwa T A, Postać 1, Wariant 1:

- (a) nazwa kraju
- (b) nazwa POA
 - nazwa WOA
 - nazwisko
 - nazwa instytucji
 - nazwa jednostki organizacyjnej
 - atrybuty organizacyjno-administracyjne

W postaci 1, wariant 2 po atrybutach (a) i (b) występuje obligatoryjnie atrybut wyróżniający: identyfikator SU, oraz opcjonalne atrybuty organizacyjno-administracyjne.

Atrybutem wyróżniającym postać 1, wariant 3, T A jest adres w g Zalecenia X.121 (po atrybutach (a) i (b)).

Nazwa T A postaci 2 składa się z: adresu telematycznego, oraz identyfikatora terminala telematycznego.

Postać 1 jest przeznaczona dla użytkowników SPW, natomiast postać 2 do identyfikacji użytkowników telematycznych. Pod hasłem "Atrybuty organizacyjno-administracyjne" zostały ujęte wszystkie informacje niezbędne do identyfikacji SU, a więc atrybuty nazw T A omawiane poprzednio, nie objęte normalizacją i wybierane w g potrzeb każdego SPW. Oczywiście w konkretnej wymianie wiadomości twórca posługuje się tylko jednym wariantem.

Z zagadnieniem nazw i adresów ściśle wiąże się zagadnienie katalogów adresów, a szerzej usług katalogowych (directory services). Usługi te mają między innymi obejmować:

1. weryfikację podanej nazwy T A
2. określenie adresu T A na podstawie nazwy T A
3. identyfikację usługi wieloadresowej i określenie pełnych adresów na podstawie podanej listy (rozdzielnika)
4. kontrolę uprawnień użytkownika stacji o nazwie T A i inne.

Zbiory katalogowe są częścią bazy danych SPW i oczywiście muszą być dostępne wszystkie operacje typowe dla eksploatacji i utrzymania baz danych.

Systemy i usługi katalogowe są również w centrum zainteresowania CCITT podobnie jak SPW. Przygotowywana jest seria zaleceń o tymczasowym oznaczeniu X. ds1 -X.ds7) 11 - 16). Zalecenia tej serii są opracowywane zgodnie z metodyką modelu ISO OSI 18 a także SPW i obejmują między innymi: model, elementy obsługi, protokoły, zagadnienia nazw i adresów, oraz ważne problemy zabezpieczeń, kontroli uprawnień i dostępu. Ze względu na szeroki zakres zastosowań usług katalogowych, obejmujący systemy informacyjne i sieci zintegrowane, obszerna problematyka katalogów wymaga odrębnego omówienia i wykracza poza ramy niniejszego referatu.

4. ELEMENTY OBSŁUGI

W Zaleceniach serii X.400 wyróżniono podstawowe i dodatkowe elementy obsługi. Podstawowe elementy obsługi (Basic Message Transfer Service Elements) powinny występować w każdym SPW, natomiast dodatkowe elementy obsługi (Optional Service Elements) są opcjonalne. Podstawowe elementy obsługi przedstawione są w tabelicy 1.

Tablica 1. Podstawowe elementy obsługi SPW
Table 1. Basic Message Transfer Service Elements

Id.	Nazwa elementu obsługi	Miejsce powstania	Miejsce oddział.
1	Nadanie identyfikatora wiadomości	PW	PW, SU
2	Zawiadomienie o niemożliwości doręczenia	PW	SUT
3	Poinformowanie o typach używanych kodów	SUT	PW
4	Określenie pierwotnego typu kodu wiadomości	SUT	PW, SUA
5	Zaznaczenie dokonanej konwerwji kodów	PW	SUA
6	Ostemplowanie nadawcze	PW	SU
7	Ostemplowanie odbiorcze	PW	SUA
8	Kontrola dostępu	SU, PW	PW, SU
9	Określenie typu treści wiadomości	SUT	PW

SUT, SUA oznaczają odpowiednio: stację twórcy i stację adresata.

Omówimy krótko niektóre elementy obsługi z tablicy 1. PPW przydziela SU twórcy unikalny identyfikator każdej wiadomości, wykorzystywany później przy powoływaniu się na tę wiadomość, (np. przy zawiadomieniu o niemożliwości doręczenia lub w odpowiedzi na tę wiadomość, w kolejnej wiadomości powiązanej logicznie z poprzednią itp.). Wiadomość można ostemplować (tj. podać datę i czas) zarówno przy nadawaniu, jak i doręczaniu. Użytkownik może zrezygnować z podania czasu, ale system musi oferować taką możliwość.

Kontrola dostępu dotyczy styku SU-PW i może być realizowana po obu stronach styku. Kontrola uprawnień może wykorzystywać często zmieniane hasła. Nie ma to nic wspólnego z hasłami kontrolującymi dostęp użytkownika do SPW.

Dodatkowe elementy obsługi SPW zostały podzielone na cztery kategorie:

1. nadawanie (doręczanie) (Submission and Delivery)
2. konwersja kodów (Conversion)
3. zapytania (Query)
4. doręczenia warunkowe (Status and Inform).

W nadawaniu doręczaniu wyróżniono dziewięć elementów obsługi. Źródłem wszystkich jest SU twórcy. Do ważniejszych elementów należą: określenie stopnia pilności (pilna, normalna i nie - pilna); określenie najwcześniejszego doręczenia (doręczenie okolicznościowe - przed podaną datą i godziną nie powinno się doręczać); żądanie potwierdzenia odbioru; żądanie zwrotu niedoręczonej wiadomości (wykonalne, jeżeli nie było konwersji kodów).

W chwili obecnej zalecenia dotyczące konwersji kodów dotyczą tylko SOP. SUT ma do dyspozycji trzy opcje: wydanie zakazu konwersji kodu danej wiadomości; zezwolenie na konwersję kodów przez PW bez informowania o dokonaniu konwersji, oraz żądanie wykonania określonej konwersji (w połączeniach międzysieciowych przy niekompatybilnych terminalach).

W kategorii zapytań jest obecnie jeden element. SUT może zapytać PPW, czy wiadomość, którą zamierza wysłać może zostać doręczona. Podaje przy tym rozmiar, typ treści i kod wiadomości. PPW na podstawie tego zapytania tworzy wiadomość służbową z zapytaniem i wysyła ją do potencjalnego adresata. Na podstawie odpowiedzi SUA PPW informuje SUT o możliwości przyjęcia (lub nie przyjęcia) wiadomości.

Z kategorii doręczeń warunkowych występują dwa elementy: doręczenie alternatywnemu adresatowi i doręczenie, kiedy będzie to możliwe. W pierwszym wypadku twórca określa podzbiór atrybutów adresata, który musi być spełniony, aby doręczyć wiadomość (np. tylko nazwę instytucji

i atrybuty geograficzne) i zezwala, by inne atrybuty (np. nazwisko) mogły nie pokryć się z atrybutami nazw SUA. Drugi z elementów zezwala PPW w przypadku niemożliwości doręczenia wiadomości przechowanie jej ponownie prób doręczenia w z góry określonym przedziale czasu.

Większość dodatkowych elementów obsługi jest związana z konkretnymi wiadomościami, a zestaw wybranych elementów może się zmienić od wiadomości do wiadomości. Część elementów dodatkowych jest wybierana na określony przedział czasu.

W systemach osobistego porozumiewania się rozszerzono zbiory elementów obsługi w stosunku do zbiorów omówionych. W zbiorze elementów podstawowych dodatkowy element przenosi uprawnienia identyfikatora wiadomości z PPW na stację twórcy. W zbiorze elementów dodatkowych jest o 18 elementów więcej w dwóch dodatkowych kategoriach, ale ramy referatu nie pozwalają na ich omówienie.

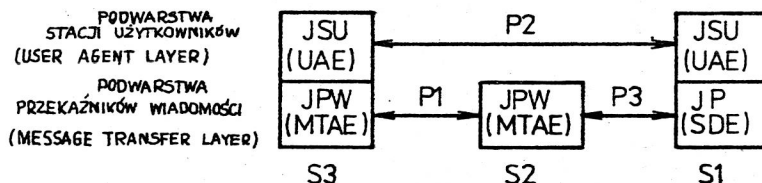
5. ARCHITEKTURA LOGICZNA SPW (SOP)

Architektura logiczna SPW jest w pełni zgodna z architekturą modelu ISO OSI 18 i została opracowana zgodnie z metodyką tego modelu. Jednostki i protokoły SPW zostały umieszczone w warstwie zastosowań modelu OSI, ale SPW korzystają z usług i protokołów warstw niższych. Omówimy architekturę SPW dla klasy SOP, ponieważ tylko dla SOP wystąpiła standaryzacja wszystkich protokołów. Funkcje SPW(SOP) w warstwie zastosowań są realizowane przez dwie podwarstwy:

- podwarstwę stacji użytkowników (PSU) (User Agent Layer-UAL),
- podwarstwę przekaźników wiadomości (PPW) (Message Transfer Layer-MTL).

PSU reprezentuje działania SU związane z zawartością wiadomości, natomiast PPW reprezentuje działania PW i świadczy usługi przenoszenia wiadomości. Wyróżniono 3 typy systemów:

- S1- systemy realizujące tylko funkcje SU
- S2- systemy realizujące tylko funkcje PW
- S3- systemy realizujące zarówno funkcje SU jak i PW.



Rys.3. Architektura logiczna SOP (warstwa zastosowań)

Fig.3. Layer Description of the IPMS

Tablica 2. Zbiór prymitywów usługowych PPW i ich parametrów
 Table 2. Set of MTL service primitives and parameters.

Usługa	Prymityw	Parametry
Nawiązanie dialogu JSU - PPW	LOGON. Request LOGON. Confirm. LOGON. Indication LOGON. Response	Nazwa T/A, Hasło Potwierdzenie /pozytyw. negatyw/, {Przyczyna nie- powodzenia}, ... Nazwa JPW, Hasło, { Licz- ba wiad. czekaj.} Potwierdzenie, { Przyczy- na niepowodzenia}
Modyfikacja paramet- rów JSU zarejestro- wanych w PPW	REGISTER. Request REGISTER. Confirm.	Zestaw nowych wartości parametrów JSU /np. typy akceptowanych kodów, max. dopuszcz. długość treści/ Potwierdzenie, { przyczyna niepowodzenia}
Nałożenie ograni- czeń na param.wiad. akceptowanych przez JSU	CONTROL. Request CONTROL. Confirm.	Zestaw parametrów /podo- bny jak dla REGISTER.Reg. Potwierdzenie, { Przyczy- na niepowodzenia}
Nałożenie ograni- czeń na param.wiad. akceptowanych przez PPW	CONTROL. Indication CONTROL. Response	Zestaw wartości paramet- rów /analogiczny, jak dla CONTROL. Request/ {Wskaźnik czekających wiadomości}
Zapoczątkowanie na- dawania wiadomości przez JSU	SUBMIT. Request SUBMIT. Confirm.	Wiele parametrów wraz z samą wiadomością Wiele parametrów
Zapytanie o możli- wość doręczenia wiadomości	PROBE. Request PROBE. Confirm.	Takie same parametry, jak w SUBMIT. Request bez sa- mej wiadomości Jak w SUBMIT. Confirm.
Doręczenie wiado- mości adresatowi	DELIVER. Indication	Wiele parametrów wraz z wiadomością
Informowanie JSU o losach nadanej w. lub odp.na PROBE.R.	NOTIFY. Indication	Wiele parametrów
Wstrzymanie doręcze- nia	CANCEL. Request CANCEL. Confirm.	Identyfikator SUBMIT. /parametr/ Potwierdzenie, { Przyczy- na niepowodzenia}
Zmiana hasła JSU	CHANGE-PASSWORD. Request CHAN.-PASS. Confirm. CHAN.-PASS. Indic.	Stare hasło, Nowe hasło Potwierdzenie, { Przyczy- na niepowodzenia} Stare hasło, Nowe hasło
Zakończenie dialogu JSU - PPW	LOGOFF. Request LOGOFF. Confirm.	- -

Na rys.3 umieszczono 3 jednostki funkcjonalne:

- jednostkę stacji użytkownika (JSU) (User Agent Entity - UAE);
- jednostkę przekaźnika wiadomości (JPW) (Message Transfer Agent Entity-MTAE),- oraz
- jednostkę pośredniczącą (JP) (Submission and Delivery Entity-SDE) i 3 protokoły:
- protokół przenoszenia wiadomości (Message Transfer Protocol)-P1,
- protokół przyjmowania i doręczania (Submission and Delivery Protocol)-P3, oraz
- protokół osobistego porozumiewania się (Interpersonal Messaging Protocol)-P2.

Protokół P1 definiuje przekazywanie wiadomości pomiędzy JPW i inne funkcje PPW. Protokół P3 zapewnia dostęp JP systemu S1 do usług PPW. Obydwa protokoły są zdefiniowane w zaleceniu X.411 8 . Protokół P2, zdefiniowany w zaleceniu X.420 9 zawiera definicje składni i semantyki swoich elementów używanych do tworzenia wiadomości przez JSU, definicje operacji związanych z wymianą w FSU, oraz opis reguł korzystania przez JSU z usług PPW.

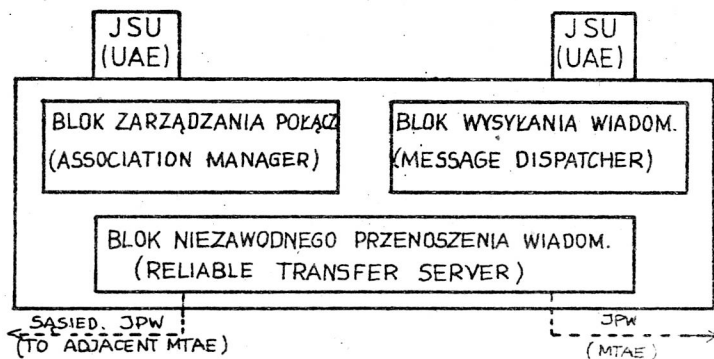
Omówimy teraz krótko usługi PPW i protokół P1, mające kluczowe znaczenie dla architektury logicznej SPW. Podwarstwa przekaźników wiadomości dostarcza środki przenoszenia wiadomości podwarstwie stacji użytkowników, doręcza wiadomości w określonym przedziale czasu i dokonuje konwersji kodów treści wiadomości na żądanie.

Usługi PPW są opisane przy pomocy zbioru prymitywów usługowych (Service Primitives). Z każdym prymitywem usługowym związany jest na ogół zbiór parametrów. Elementy obsługi omówione w p.4 występują w samych prymitywach, bądź w ich parametrach. Wyróżniono dwa typy parametrów : obligatoryjne (O) (Mandatory-M) i warunkowe (W) (Conditional-C). Parametry obligatoryjne muszą wystąpić, chociaż dopuszcza się wartości zerowe, lub standardowe (defaulted). Parametry warunkowe występują tylko po spełnieniu określonych warunków. Usługi, prymitywy usługowe i ich niektóre parametry są przedstawione w tablicy 2. Parametry warunkowe zostały ujęte w nawiasy klamrowe. Zgodnie z ogólnie przyjętym zwyczajem nazwy prymitywów usługowych podajemy tylko w brzmieniu oryginalnym. Bogate listy parametrów prymitywów SUBMIT. Request, PROBE. Request, DELIVER. Indication zawierają między innymi: nazwę T A twórcy, nazwę T A adresata, typ treści, typ kodu (kodów), priorytet, parametry konwersji i różnych usług dodatkowych.

Pewne usługi podwarstwy przekaźników wiadomości wymagają współpracy dwóch lub więcej JPW. Taką współpracę zapewnia protokół przenoszenia

wiadomości P1. Elementy protokołu P1 noszą nazwę jednostek danych protokołu wiadomości (JDPW) (Message Protocol Data Units-MPDUs), które z kolei dzielą się na JDPW użytkownika (JDPWU) (UMPDUs) oraz JDPW obsługi (JDPWO) (SMPDUs). JDPWU przenoszą wiadomości użytkownika, natomiast JDPWO przenoszą informacje o wiadomościach.

Aby JPW była zdolna do współpracy z innymi JPW ma ona strukturę przedstawioną na rys.4.



Rys.4. Model JPW

Fog.4. Model of an MTAE

JPW ma trzy bloki składowe:

- blok wysyłania wiadomości (Message Dispatcher)
- blok zarządzania połączeniami (asocjacjami) (Associaton Manager)
- blok niezawodnego przenoszenia wiadomości (BNP) Reliable Transfer Server-RTS).

Blok wysyłania wiadomości wykonuje działania protokołu P1 wskazane przez JDPW odebrane od innych JPW lub wynikające z wiadomości odebranych od własnych JSU, dołączonych do JPW.

Blok zarządzania połączeniami tworzy, nadzoruje i zwalnia połączenia oferowane przez BNP.

Blok niezawodnego przenoszenia jest odpowiedzialny za utrzymywanie połączeń między JPW i za niezawodne przenoszenie JDPW. Połączenie może być jednokierunkowe (jednokierunkowe przenoszenie JDPW) lub dwukierunkowe naprzemienne.

Formalna definicja protokołu P1 w Zaleceniu X.411 składa się z definicji JDPWU, definicji raportu doręczenia JDPW, oraz definicji JDPW zapytania. W tym samym Zaleceniu X.411 znajduje się specyfikacja protokołu P3, też w PFW.

Opis PSU i specyfikacja protokołu P2 są podane w Zaleceniu X.410. Zalecenie to omawia również tzw. proste dokumenty formatowalne (Simple

Formattable Documents) (SFD) i formalną definicję ich struktury.

Usługi i protokoły opisane w Zaleceniach X.411 oraz X.420 są realizowane w warstwie zastosowań modelu OSI. Sposób wykorzystania warstw modelu OSI poniżej warstwy zastosowań do niezawodnego przenoszenia wiadomości omawia Zalecenie X.470 7, natomiast minimalne konkretne wymagania związane z warstwą prezentacji zalecenia: X.408 5, dotyczące konwersji postaci informacji, oraz X.409 6, związane ze składnią przesyłania w warstwie prezentacji.

6. EAN: PRZYKŁAD USŁUG ISTNIEJĄCEGO SOP.

Dla ilustracji działania SPW omówimy usługi przykładowego kanadyjskiego SOP o nazwie EAN 20.

Użytkownik EAN korzysta z kanadyjskiej sieci komputerowej CDN i może wymieniać wiadomości z innymi użytkownikami SOP w CDN, a także użytkownikami SOL, wdrożonymi w sieciach komputerowych UUCP, CS i ARPA. EAN została zaprojektowana zgodnie z zaleceniami CCITT dla SPW. Umożliwia to połączenie z dalszymi publicznymi i prywatnymi SPW w przyszłości. Użytkownik EAN może stworzyć, zredagować, wysłać, otrzymać i zarchiwizować wiadomość. Wiadomość może być wysłana do pojedynczego użytkownika, bądź do użytkowników z podanej listy adresatów (usługa wieloadresowa). Adresat wiadomości może wiadomość odebrać, zarchiwizować, lub wysłać ją dalej (do innego użytkownika). Może też zredagować odpowiedź na otrzymaną wiadomość. W systemie występują dwa typy wiadomości: wiadomości użytkownika i raporty o stanie. Wiadomość użytkownika zawiera nagłówek i treść, która może być tekstowa, graficzna, binarna, a także mieć postać głosu lub faksymile. Raport o stanie (losach) wcześniej wysłanych wiadomości przyjmuje często postać zawiadomienia o niemożliwości doręczenia wiadomości (na skutek adresu nieznanego dla SOP lub błędu w adresie). Raport o stanie może być potwierdzeniem odbioru wiadomości, lub może zawierać pełną treść wiadomości niedoręczonej (zwrot do nadawcy).

Adres SU składa się z nazwy OA, listy podobszarów i nazwy skrzynki pocztowej. OA oznacza tutaj sieć komputerową (CDN, UUCP, CS, ARPA). Przykład adresu: jsmith abc. CDN.

jsmith jest nazwą skrzynki pocztowej Joe Smith'a, abc nazwą podobszaru, CDN-nazwą OA. Analogicznie adresuje się wiadomości do użytkownika sieci, z których CDN jest połączona, np.: tjones mit. ARPA.

Omówimy teraz usługi i funkcje systemu.

Wprowadzenie nowego użytkownika.

Użytkownik korzystający po raz pierwszy z systemu podaje szereg danych do bazy danych systemu, w szczególności hasło. Hasło jest jedynym ograniczeniem dostępu i może być zmienione w dowolnej chwili przez zainteresowanego zmianą użytkownika. Użytkownik zarejestrowany w bazie danych może wysyłać i przyjmować wiadomości.

Redagowanie i nadawanie wiadomości.

Pierwszą fazą tworzenia nowej wiadomości jest jej redagowanie. Redagowanie rozpoczyna się od instrukcji COMPOSE. Po wprowadzeniu z klawiatury SU adresu, zgodnie z podanymi wcześniej zasadami, twórca wiadomości może korzystać z gotowego schematu szkieletowego wiadomości. EAN stosuje 15 pól nagłówka, przy czym niektóre są wypełniane automatycznie, a kilka można pominąć. Z ważniejszych pól wymienimy pola list adresatów (pierwszej kolejności doręczania, drugiej kolejności i adresatów tajnych (poufnych)), pole "od kogo" identyfikator wiadomości "czego dotyczy". Na ogół wiadomość ma jednego adresata, ale EAN może obsługiwać połączenia wieloadresowe z priorytetami. Adresat z pól "tajnych" nie występuje na żadnej liście i sam nie ma dostępu do listy adresatów "tajnych" (ale może mieć dostęp do pozostałych list adresatów). Istnieje też pole klasyfikacji wiadomości, gdzie przewidziano 3 wartości: OSOBISTA, PRYWATNA, POUFNA. Pole "od kogo" jest wypełnione automatycznie przez SU, chociaż twórca może sam je wypełnić. Unikalny identyfikator wiadomości jest automatycznie przydzielany przez EAN. Pole określające "czego dotyczy" wiadomość wykorzystywane jest przy odbiorze. Inne pola określają na przykład datę ważności, stopień ważności, powiązania z innymi wiadomościami, termin i miejsce wysłania odpowiedzi.

Po przygotowaniu nagłówka twórca korzysta z edytora do napisania treści. Zredagowaną wiadomość może wyświetlić na monitorze, nadać i zarchiwizować we własnej bazie danych w odpowiednim zbiorze.

Odbieranie i przeglądanie wiadomości.

Każdorazowo po wywołaniu EAN automatycznie wyświetlany jest spis nowych wiadomości, które znalazły się w skrzynce pocztowej. Każda pozycja w tym spisie zawiera: numer kolejny w skrzynce, znaczek "NP"-nowej, jeszcze nie przeczytanej wiadomości (w oryginale .U), nazwisko twórcy, datę nadania i czego dotyczy, np.: 7 NU Harry Rudin, Sept.1086 IPIP Symp.

Użytkownik może wyświetlić te nowe wiadomości, jak również przejrzeć zawartość skrzynki, lub też usunąć dowolną wiadomość ze skrzynki (już niepotrzebną). W przypadku użytkowników prowadzących ożywioną wy-

mianę wiadomości celowe jest posiadanie uporządkowanych archiwów. Aby uwolnić takich użytkowników od częstego przeglądania, porządkowania i "opróżniania" skrzynki można zlecić systemowi automatycznie dopisywanie odebranych wiadomości do odpowiednich zbiorów (z pominięciem zbioru "skrzynki pocztowej"). Wcześniej użytkownik musi określić wartości pól, na podstawie których odebrana wiadomość ma być kierowana do określonego zbioru.

Inne usługi

Istnieje możliwość szybkiego zredagowania odpowiedzi na ostatnio otrzymaną wiadomość (za pomocą instrukcji REPLY). Redagowanie odpowiedzi odbywa się analogicznie, jak przy COMPOSE, z tym, że nowe pole "w odpowiedzi na ..." jest wypełniane automatycznie.

EAN ma szeroko rozbudowane usługi katalogowe, umożliwiające znajdowanie adresów T A na podstawie nazw T A.

Wiadomości mogą być powiązane ze sobą logicznie. System umożliwia podanie tych powiązań. EAN również pozwala na opóźnienie doręczenia (doręczenie od wskazanej daty), jak również podanie listy użytkowników i daty, przed upływem której należy wysłać odpowiedź do wszystkich użytkowników z listy.

7. WYBRANE ZAGADNIENIA IMPLEMENTACJI SPW.

Zagadnienia implementacji i ewolucji SPW omówimy na przykładzie sieci CS ze względu na różnorodność jej sprzętu i oprogramowania oraz krąg użytkowników. CSNET jest logiczną siecią teleinformatyczną, której użytkownikami są naukowcy w ośrodkach uniwersyteckich Stanów Zjednoczonych. W skład CSNET wchodzi 5 podsieci fizyczne: sieć ARPA, sieć Telenet, oraz Phone Net. Sieć ARPA jest chyba najstarszą siecią rozległą dla ośrodków uniwersyteckich; wiele sieci amerykańskich, japońskich i europejskich ma do niej dostęp, jest stale rozwijana i dość dobrze znana. Sieć Telenet jest też już stosunkowo długo eksploatowana na zasadach komercyjnych. Nowością w stosunku do dwóch pierwszych sieci jest Phone Net. W pierwszej fazie rozwoju CSNET (do lipca 1982 r. 17) wdrożono usługi poczty elektronicznej na bazie ARPANET i Phone Net. Użytkownicy systemów liczących dużej mocy (dużych Hostów) sieci ARPA komunikują się, używając protokołu MULSTP TCP IP 21. Wiadomości, które są zgodne ze Standardem Formatu Sieciowych Wiadomości Tekstowych ARPA są wymieniane przy użyciu protokołu o nazwie Simple Mail Transfer (SMTP) 21. Odpowiada on protokołowi P1.

Użytkownicy sieci ARPA, mający dostęp do komputerów małej i średniej mocy, które nie pozwalają na wdrożenie JPW (zbyt mała moc, szybkość przetwarzania, pamięć masowa, bądź inne ograniczenia) są w sytuacji systemu S3. Dla nich opracowano Post Office Protocol 21, wdrożony w PPW w dużych komputerach, obsługujących dołączone do nich zdalne SU. Jest to odpowiednik P5.

Dla użytkowników nie należących do sieci ARPA wdrożono usługę poczty elektronicznej w oparciu o SPW o nazwie Phone Net. SPW miał w pierwszej fazie dwa przekaźniki obsługujące 24 miasta na całym terytorium USA (ARPANET ma obecnie przeszło 100 węzłów w USA, na Hawajach, w Europie i Japonii). Podsieć Phone Net jest oparta na sieci telefonicznej komutowanej, ale dostęp użytkowników jest ograniczony (z góry wyznaczone godziny seansów w ciągu dnia). W drugiej fazie rozbudowy SCENT zintegrowano z dotychczasowymi podsieciami sieć Telnet. Kłopot polegał na tym, że sieć Telenet, stosuje protokół X.25, a więc kanały logiczne, a ARPA datagramy. Został pomysłnie opracowany i wdrożony styk X.25 TCP-IP. Dalsza ewolucja CNEST związana była z przetwarzaniem wiadomości. W centralnej bazie danych CNEST na Uniwersytecie Wisconsin wprowadzono usługę katalogów o nazwie CSNET Name Server 22. Początkowo Name Server miał za zadanie ułatwić wysyłanie poczty przez odciążenie użytkowników od szczegółów adresowania i danie różnorodnych możliwości określania adresata (różne nazwy, skróty, itp.). W dalszych fazach projektu Name Server miał umożliwić przesyłanie zbiorów i zdalny dostęp do zasobów komputerowych.

Już w 1983r podjęto próby z dołączeniem komputerów osobistych Apple II, pracujących pod systemem operacyjnym CP M2.2 do CSNET. Protokoły były podobne do IMDF, używanych w Phone Net. Wielu producentów pracuje obecnie nad umożliwieniem dostępu komputerom osobistym do SPW. Przedstawimy krótko koncepcję opracowaną przez Nippon Telegraph and Telephone Corp.¹⁹ Koncepcja zakłada wyposażenie użytkownika komputera osobistego w adapter komunikacyjny, pośredniczący między komputerem i siecią telefoniczną. Adapter ma na poziomie fizycznym wbudowany modem 4.8 kB/s, zgodnie z zaleceniem V.27 ter CCITT. Został on wybrany, ponieważ jest już używany w terminalach faksymile G3, teletekstu i videotekstu (system CAPTAIN) w Japonii. Adapter realizuje też całą architekturę logiczną OSI do poziomu sesji. W warstwie kanału danych: X.25 lub B (protokół T.71), w warstwie sieciowej: protokół T.70, w warstwie transportowej: protokół klasy O(X.224), a w warstwie sesji: X.225. Stykiem fizycznym adapter-komputer jest RS 232 C, natomiast instrukcje logiczne są oparte na zaleceniu X.215. Część warstwy sesji (sterowanie adapterem) i wszystkie mechanizmy specyficzne dla SPW, leżące wyżej

(Tj. X.409, X.410 i X.420) powinny być zaprogramowane w komputerze. Aby napływające wiadomości nie zakłócały funkcji lokalnych i były odbierane automatycznie, komputer osobisty musi mieć mechanizm skrzynki pocztowej (Mailbox Facility). Jest on niestandardowy. Drugi mechanizm związany z przechowywaniem, wyszukiwaniem, odczytywaniem i usuwaniem odebranych wiadomości został nazwany mechanizm tablicy ogłoszeń (Bulletin Board) i jest również niestandardowy. Obydwa mechanizmy rezydują w warstwie zastosowań, nad połączeniami niezawodnego przenoszenia (Zalecenie X.410).

8. PODSUMOWANIE

Omówiono model systemu przekazywania wiadomości, jego elementy i architekturę logiczną, zgodne z zaleceniami CCITT serii X.400. Przedstawiono ważne zagadnienie zasad tworzenia nazw i adresów w SPW. Dużo miejsca poświęcono usługom systemu, oraz architekturze logicznej SPW. Rozważania natury ogólnej zilustrowano opisem usług w istniejącym SPW o nazwie EAM. Następnie przedstawiono przykład implementacji SPW w RSK i zagadnienia dołączania komputerów osobistych do SPW. Referat sygnalizuje nową grupę zastosowań sieci teleinformatycznych, dla której przewiduje się dynamiczny rozwój w najbliższych latach.

LITERATURA

- [1] Bocker P., "Universal Communication Services in the Communication Network of the Future", Proc. ICCO '86, Munich, FRG, 5-10.
- [2] Burgassi M., Agostini V.D.-Montenurro R.: "Issues on an Experimental Implementation and Perspectives for Service Provisioning", Proc. ICCO '86, Munich, FRG, 56-60.
- [3] CCITT Draft Recommendation X.400: "Message Handling Systems: System Model-Service Elements", 1985.
- [4] CCITT D.Rec.X.401: "MHS: Basic Service Elements and Optional User Facilities", 1985.
- [5] CCITT D.Rec. X.408: "MHS: Encoded Information Type Conversion Rules", 1985.
- [6] CCITT D.Rec.X.409: "MHS: Presentation Transfer Syntax and Notation", 1985.
- [7] CCITT D.Rec.X.410: "MHS: Remote Operations and Reliable Transfer Server", 1985.
- [8] CCITT D.Rec.X.411: "MHS: Message Transfer Layer", 1985.
- [9] CCITT D.Rec. X.420: "MHS: Interpersonal Messaging User Agent Layer" 1985.

- [10] CCITT D.Rec. X.430: "MHS: Access Protocol for Teletex Terminals", 1985.
- [11] CCITT D.Rec.X.ds1: "Directory Systems: Model and Service Elements" Reston, 1986.
- [12] CCITT D.Rec. X.ds2: "Directory Systems: Information Framework", Reston, 1986.
- [13] CCITT D.Rec. X.ds3: "Directory Systems: Protocols", Reston, 1986.
- [14] CCITT D.Rec.X.ds4: "Directory Systems: Standard Attribute Types", Reston, 1986.
- [15] CCITT D.Rec. X.ds6: "Directory Systems: Suggested Naming Practices", Reston, 1986.
- [16] CCITT D.Rec. X.ds7: "Directory Systems: Authentication Framework", Reston, 1986.
- [17] Comer D.: "The Computer Science Research Network CSNet: History and Status Report" Commun. of the ACM, 26, 747-53.
- [18] ISO TC 97 SC 16: "Information Processing Systems - Open Systems Interconnection - Basic Reference Model", IS 7498, 1985.
- [19] Nakayama T., Nomura M.: "Developments of MHS in NTT-Connection of Personal Computers to MHS", Proc. ICCO '86, Munich, FRG, 61-66.
- [20] Neufeld G., Sample R., Hilpert E., Deering S., Demco J.: "The EAH Distributed Message System", Dept. of Comp. Sci. Univ. of British Columbia, Vancouver, Canada, 1983.
- [21] Rose M.T., Stefferud E.A., Sweet J.N.: "MH: A Multifarious User Agent", Computer Networks and ISDN Systems, 10, 1985.
- [22] Solomon M., Landweber L.H., Neuhengen D.: "The CSNET Name Server" Computer Networks, 6, 1982, 161-172.

MESSAGE HANDLING SYSTEMS

Message Handling Systems, based on CCITT X.400 series Recommendations are considered. First the system model is presented, followed by naming and addressing concepts. Then Message Transfer service elements are discussed and layered representation of the MHS model is described. The Message Handling entities and protocols are located in the Application Layer of the OSI Reference Model. Canadian EAH distributed message system is presented as an example of existing MHS. Then development of MT service in US CSNET is described. Finally the concept of connection of personal computers to MHS is sketched. An extraordinary growth in use of messaging systems over the next decade, together with other value-added systems is expected.

КОМПЬЮТЕРНЫЕ СИСТЕМЫ ПЕРЕДАЧИ СООБЩЕНИЙ

В выступлении описано системы перелачи сообщений (СПС), внедряемые в виду новых услуг в сетях телеобработки и соответствующие рекоменда-
циям МККТТ серии X.400. Представлено модель СПС ,принципы построения
названий и адресов, элементы обслуживания и логическую архитектуру со-
гласную с моделью МОС (OSI). Выводы проиллюстрировано описанием сущес-
твующей СПС. Представлено тоже пример имплементации СПС в обширной се-
ти ЭВМ и описано вопросы присоединения персональных компьютеров к СПС.

sieci komputerowe
protokoły
metody specyfikacji
inżynieria protokołów

Maciej STROIŃSKI*

METODY FORMALNEJ SPECYFIKACJI PROTOKOŁÓW ORAZ ICH ZASTOSOWANIA

W artykule omówiono stan badań związanych z rozwojem i zastosowaniem metod formalnej specyfikacji protokołów. Przedstawiono najpopularniejsze metody formalnej specyfikacji oraz scharakteryzowano prace zajmujące się oceną przydatności tych metod. Następnie omówiono elementy zintegrowanego systemu wytwórczego protokołów i podano przykłady systemów wspomagających. Na zakończenie wskazano kierunki przyszłych prac badawczych w inżynierii protokołów.

1. WSTĘP

Duże zainteresowanie metodami formalnej specyfikacji protokołów (mfsp) i wynikający stąd rozwój tych metod są motywowane możliwościami praktycznego zastosowania metod formalnych do wspomagania prac na etapie projektowania, implementacji i eksploatacji protokołów. W ostatnich latach stosowano wiele metod formalnej specyfikacji protokołów. Stan prac nad rozwojem i zastosowaniem mfsp jest doskonale dokumentowany materiałami sympozjów grupy roboczej WG 6.1 IFIP odbywanych pod wspólnym tytułem "protocol specification, testing, and verification" [14,23,51,54,68,82].

Analiza tych prac ukazuje, jak zainteresowanie badaczy początkowo dotyczące projektowania nowych metod i przystosowywania metod specyfikacji stosowanych w inżynierii oprogramowania aktualnie koncentrują się na pracach związanych z wybranymi metodami będącymi przedmiotem standaryzacji. Do rozważanych zastosowań mfsp, takich jak weryfikacja i testowanie protokołów, doszły: automatyczna implementacja protokołów oraz ocena ich efektywności. Przystosowywanie mfsp do tych nowych zastosowań oraz tworzenie zintegrowanych narzędzi wspomagających wielospektową ich użycie wyznaczają aktualne kierunki prac badawczych. Uzys-

*Instytut Automatyki, Politechnika Poznańska, Poznań.

kane w tym zakresie wyniki (patrz np. [14]) upoważniają nas już obecnie do stwierdzenia, że rozpoczyna się intensywny rozwój nowej dziedziny nazywanej inżynierią protokołów.

Pojęcie inżynierii protokołów wprowadzono w 1980 r. [46] jako określenie kompleksu działań związanych z projektowaniem, specyfikacją, oceną, implementacją, zarządzaniem i zastosowaniem protokołów [46, 48, 49, 56]. Zadaniem inżynierii protokołów jest dostarczenie bazy teoretycznej i narzędzi wspomagających te działania. W kontekście inżynierii protokołów prace nad rozwojem i zastosowaniem mfsp odgrywają kluczową rolę.

W artykule przedstawimy najpopularniejsze metody formalnej specyfikacji protokołów, a także scharakteryzujemy prace zajmujące się oceną przydatności tych metod. Następnie omówimy elementy zintegrowanego, hipotetycznego systemu wytwórczego protokołów i wskażemy przykłady systemów wspomagających.

2. METODY FORMALNEJ SPECYFIKACJI PROTOKOŁÓW

Na przestrzeni ostatnich 10 lat stosowano wiele metod formalnych do specyfikacji protokołów, usług i architektury logicznej sieci komputerowych. Wśród tych metod wyróżnić można [59]: automaty skończone; maszyny abstrakcyjne obejmujące: automaty kontekstowe; interlokutory; diagramy przejść stanów przetwarzania; rozszerzone sieci Petriego, czasowe sieci Petriego, modele hybrydowe, gramatyki formalne, wyrażenia sekwencyjne, logikę czasową, abstrakcyjne typy danych, formalne języki specyfikacji oraz języki programowania. Przegląd, porównanie i bogatą literaturę dotyczącą tych metod znaleźć można w pracach [12, 59, 63].

Aktualnie najszerzej stosowane są języki formalnej specyfikacji, które stały się przedmiotem intensywnych prac normalizacyjnych w CCITT i ISO. Wśród innych metod formalnej specyfikacji dużą popularność zyskały sobie rozszerzone sieci Petriego, a szczególnie numeryczne sieci Petriego i ich pochodne. Rosnące znaczenie jako narzędzia specyfikacji protokołów mają także języki programowania, a szczególnie ADA i PROLOG oraz gramatyki atrybutowe. Omówimy pokrótce podstawowe cechy tych metod oraz podamy przykłady ich zastosowania.

2.1. Języki formalnej specyfikacji

W efekcie prac rozpoczętych w 1980 r. w ramach grup roboczych ISO (ISO TC97/SC16/WG1 - ad hoc group on Formal Description Techniques) oraz podkomitetów grup studialnych CCITT (CCITT Study Group: VII/Q27, VII/Q29, XI/3-1) powstały następujące propozycje języków formalnej

specyfikacji [24,76]: SDL i X.250 (CCITT) oraz ESTELLE i LOTOS (ISO). Języki te oparte są na: modelach automatów kontekstowych (SDL) i języku programowania PASCAL (X.250 i ESTELLE) oraz na teorii Milnera "rachunku komunikujących się procesów" (LOTOS). Automaty kontekstowe są rozszerzonymi automatami skończonymi. Rozszerzenia polegają na wprowadzeniu zmiennych kontekstowych dla opisu informacji związanych z dynamiką protokołu, których uwzględnienie w modelu prowadziłoby do tzw. eksplozji stanów. W tym miejscu warto również wspomnieć o stosowanym przez IBM języku specyfikacji FAPL [50], który bazuje na rozszerzonych automatach skończonych i języku programowania PL/1.

SDL (Specification and Description Language) umożliwia specyfikowanie protokołów za pomocą komunikujących się procesów, z których każdy jest modelowany maszyną abstrakcyjną [24]. Maszyny abstrakcyjne w SDL reprezentowane są przez rozszerzone automaty skończone. Zdefiniowano dwie operacje związane ze zmiennymi kontekstowymi: jedną dotyczącą badania warunków w odniesieniu do parametrów wejść i zmiennych kontekstowych oraz drugą związaną z wykonywaniem funkcji przetwarzania zmiennych kontekstowych i manipulowania parametrami sygnałów wejściowych i wyjściowych. SDL umożliwia modelowanie przejść o niezerowym czasie trwania przez wprowadzenie pojęcia ogólnodostępnego czasu globalnego. Możliwe jest także kolejkovanie sygnałów (zdarzeń) wejściowych nadchodzących podczas realizacji danego przejścia w modelu. Kompletna specyfikacja protokołu wymaga zdefiniowania struktury systemu SDL oraz jego dynamicznego zachowania się. Statyczna struktura systemu SDL składa się z systemu, bloków i kanałów. System jest kompozycją bloków połączonych między sobą i do granic systemu przez kanały. Kanały działają jako media transportowe dla sygnałów między blokami. Bloki zawierają procesy. Dynamiczne zachowanie się systemu SDL jest obrazowane przez działające równolegle procesy. Proces działa w odpowiedzi na zewnętrzne, dyskretne pobudzenie i może generować do swego otoczenia dyskretną odpowiedź. Inne procesy mogą ją akceptować jako swoje pobudzenie. Poszczególne komponenty systemu SDL mogą być strukturalizowane, np. bloki w podbloki i podkanały, kanały w bloki i kanały, procesy w podprocesy. Definicja procesu może być uszczegółowiona w wielu krokach. Możliwe jest uzyskanie hierarchicznego, wielopoziomowego opisu systemu. Specyfikacja protokołu w SDL może być podana w formie graficznej (SDL/GR) lub w formie wyrażen tekstowych (SDL/LA). Obie formy są równoważne semantycznie. SDL zastosowano do specyfikacji protokołów: X.25 LAPB, X.75, protokołu transportowego ISO klasy 0,1,2 i 3. Omawiany język specyfikacji stanowi także podstawę do budowy efektywnych symulatorów protokołów (np. [28]). SDL jest omówiony w zaleceniach CCITT: Z100-Z104.

ESTELLE (Extended State Transition Language) jest językiem zorientowanym automatowo [38,76]. Protokół jest specyfikowany jako kompozycja komunikujących się komponentów zwanych modułami. Połączenia modułów realizowane są przez podsystemy zwane kanałami. Moduł jest maszyną abstrakcyjną, w której użytkownik specyfikuje przejścia i funkcje wyjść stosując język PASCAL. Ogólna przestrzeń stanów modułu zawiera stany główne reprezentowane przez zmienne stanów oraz stany pomocnicze reprezentowane przez zmienne kontekstowe związane ze zmiennymi programowymi. Kanał jest pełnodupleksową parą kolejek. Moduły i kanały mogą działać współbieżnie. Sygnały kanałowe mają charakter dyskretny. Istnieje możliwość związania z przejściem opóźnienia jako warunku przejścia. Moduły mogą tworzyć struktury hierarchiczne. Specyfikacja w ESTELLE jest modułem najwyższego poziomu (tak jak program w PASCAL-u). Może ona zawierać kompletny moduł lub może być widziana jako zbiór zagnieżdżonych modułów, procedur i funkcji. Mogą być dwie klasy modułów: procesy i czynności. Procesy mogą działać współbieżnie z innymi procesami na tym samym poziomie hierarchii. Czynności muszą znajdować się na najniższym poziomie hierarchii specyfikacji i nie mogą realizować się równolegle. Kanały i moduły są deklarowane jako specjalne typy danych. Syntaktyka ESTELLE jest zorientowana znakowo typu PASCAL-owskiego. ESTELLE został użyty do modelowania protokołu transferu zbiorów NBS, protokołu transportowego ISO i sieciowego protokołu bezpołączeniowego [38]. Dla języka ESTELLE powstało szereg narzędzi wspomagających przygotowanie specyfikacji i jej sprawdzenie przez symboliczne wykonanie, np. [5]. Proponowane jest także narzędzie dla graficznej specyfikacji protokołu równoważne specyfikacji w ESTELLE [79]. Język ESTELLE jest zdefiniowany w dokumencie ISO - DP 9074.

LOTOS (Language for Temporal Ordering Specification) jest językiem specyfikacji wyprowadzonym z niewielkiego rozszerzenia "rachunku komunikujących się procesów" Milnera [16]. Systemy rozproszone specyfikowane są w LOTOS-ie jako procesy. System jako całość jest pojedynczym procesem, który może zawierać zbiór komunikujących się podprocesów, które z kolei mogą zawierać podprocesy itd. Jest więc opisywana hierarchia procesów. Proces jest specyfikowany za pomocą opisu jego zdolności do komunikowania się z otoczeniem za pomocą interakcji zwanych w ich formie elementarnej zdarzeniami. Zdarzenie jest jednostką synchronizowanej komunikacji między procesami. Obserwowane zewnętrznie zachowanie systemów opisywane jest w LOTOS-ie za pomocą wyrażeń (ang. behaviour expression). Wyrażenia te formalizują porządek, w którym mogą wystąpić zdarzenia. Reguły formułowania tych wyrażeń są główną częścią LOTOS-u. Umożliwiają one prefiksowanie akcji, wybór, specyfikację procesów w formie drzewa, rekursję, współbieżność, niedeterminizm,

sekwencyjną kompozycję procesów, przerywanie procesów, zdarzenia strukturalizowane, konstrukcje warunkowe (predykaty wyboru, wyrażenia dozorowane). Język LOTOS zastosowano do modelowania usług i protokołów transportowego i sesji [16]. Równocześnie trwają prace nad narzędziami wspomagającymi użycie języka [15,16]. LOTOS zdefiniowany jest w dokumentacji ISO - DP 8807.

2.2. Języki programowania

Modele językowe są modelami opartymi na koncepcji stanów [75]. Protokoły modelowane są jako struktura połączonych komponentów, z których każdy jest specyfikowany z zastosowaniem ogólnie znanego języka programowania. Komponenty mogą być wykonywane współbieżnie, jeśli dany język programowania umożliwia współbieżność. Dobrze opisywane są takie komponenty, jak automaty, funkcje kombinacyjne, kolejki. Możliwe jest także zastosowanie zegarów (liczniki znaczników czasu), ale modelowanie opóźnień jest trudne. W specyfikacji stosowana jest notacja wykorzystująca syntaktykę zorientowaną znakowo. Przykładem zastosowania języków programowania do specyfikacji protokołów może być użycie PASCAL-a w normach IEEE 802. Wśród innych języków programowania stosowanych do specyfikacji protokołów wskazać możemy rosnące zainteresowanie językiem ADA (np. [18,49]) oraz językiem PROLOG (np. [13,39]). Język ADA wskazywany jest jako konkurent ESTELLE ze względu na zapowiadane efektywne systemy wspomagające jego użycie [49]. PROLOG natomiast ma większe znaczenie dla zastosowań związanych z analizą protokołów. Szczególnie interesująco zapowiadają się próby wykorzystania PROLOG-u i języka ADA do wspomagania modelowania za pomocą sieci Petriego (np. [6,17]) i gramatyk atrybutowych (np. [74]).

2.3. Sieci Petriego

Sieci Petriego, a szczególnie ich rozszerzone odmiany są szeroko stosowane do modelowania protokołów [21,22]. Umożliwiają one bowiem modelowanie współbieżności, hierarchii warstwowej, stanów głównych i kontekstowych, generacji sygnałów wewnętrznych, upływu czasu i niedeterminizmu [59,60]. Wśród wielu odmian tych sieci na szczególną uwagę zasługują numeryczne sieci Petriego (NPN) [70], dla których istnieją systemy wspomagające ich zastosowania: system PROTEAN dla weryfikacji protokołów [11] i system PROPER dla predykcji działania protokołów [33, 34]. Numeryczne sieci Petriego stały się także podstawą dla kilku odmian sieci, np. [45,17]. Przykładem użycia NPN dla specyfikacji protokołów mogą być modele protokołu transportowego klasy 0 [10] i 1 [19]. Wśród wspomnianych wyżej wymagań spełnianych przez sieci Petriego kluczowe znaczenie ma modelowanie czasu. Modelowanie czasu umożliwia łatwe odwzorowanie czasów przeterminowania (ang. timeout), szacowanie e-

fektywności protokołu w fazie projektowania oraz zmniejsza przestrzeń stanów modelu [57]. Czas w sieciach Petriego może być wiązany z przejściami, np. [42,52,71] lub z miejscami, np. [80]. Przy pewnych założeniach oba te czasy są równoważne [64]. Czas związany z przejściem dotyczyć może opóźnienia przejścia i/lub czasu jego trwania. Może on być deterministyczny lub określony zmienną losową. W tym drugim przypadku mamy do czynienia ze stochastycznymi sieciami Petriego, których wiele mutacji opisanych jest w pracach roboczego seminarium [72]. Ich równoważność z modelami Markowa oraz istniejące symulatory tych sieci sprawiają, że są one wygodnym narzędziem do oceny efektywności systemów, w tym także protokołów. Przykładami modelowania protokołów za pomocą stochastycznych sieci Petriego mogą być prace [24,40]. Sieci Petriego są metodą specyfikacji, która daje możliwość integrowania modeli funkcjonalnych i operacyjnych. Przez przekształcenia sieci, np. redukcję, możliwe jest uzyskiwanie modeli na różnym poziomie abstrakcji. Wspomniane cechy, jak również próby interpretacji sieci Petriego w środowisku programowym języków ADA i PROLOG wyznaczają trwałą, znaczącą pozycję sieci Petriego wśród mfsp.

2.4. Gramatyki atrybutowe

Gramatyki atrybutowe należą do sekwencyjnych technik specyfikacji protokołów. Techniki te umożliwiają opisanie sekwencji zdarzeń i akcji danego protokołu jako zdań generowanych przez daną gramatykę. Nieformalnie można powiedzieć, że gramatyka atrybutowa jest rozszerzeniem gramatyki bezkontekstowej o zbiór atrybutów. Każdy symbol należący do zbioru symboli nieterminalnych może być związany ze zbiorem atrybutów. Każdy z tych atrybutów opisuje specyficzną, kontekstowo zależną cechę tego symbolu. Dodatkowo każda produkcja jest związana z regułami oceny atrybutów symboli należących do tej produkcji. Gramatyki atrybutowe umożliwiają opis syntaktyki i semantyki obiektu (np. języka programowania) i stąd są one stosowane do budowy kompilatorów. Do modelowania protokołów stosuje się rozszerzone gramatyki atrybutowe, np. gramatykę atrybutową czasu rzeczywistego [2], gramatykę protokołów [30]. Rozszerzenia te umożliwiają specyfikowanie współbieżności i ograniczeń czasowych [2], rozdzielne definiowanie syntaktyki i semantyki protokołu, podział symboli terminalnych na oddzielne klasy związane z komunikującym się otoczeniem protokołu, wprowadzenie funkcji generowania symboli terminalnych [30]. Głównym zastosowaniem specyfikacji protokołów za pomocą gramatyk atrybutowych jest automatyczna implementacja protokołów (np. [2,3]) i wspomaganie testowania protokołów (np. [2,44,74]). Wskazany zakres zastosowań, jak również możliwość implementowania modeli gramatyk atrybutowych w środowisku programowym PROLOG-u wyznacza

przyszły wzrost zainteresowania tym językiem jako metody specyfikacji protokołów.

3. OCENA METOD FORMALNEJ SPECYFIKACJI PROTOKOŁÓW

Duża różnorodność metod formalnej specyfikacji protokołów w naturalny sposób implikuje poszukiwania metody najlepszej dla konkretnego celu i obiektu modelowania. Brak formalnych, wymiernych kryteriów porównawczych dla oceny mfsp był czynnikiem utrudniającym takie poszukiwania. W ostatnich latach opublikowano kilka prac dotyczących tej problematyki [1,26,49,59,60,75]. Omówimy uzyskane w nich wyniki.

W pracach [59,60] dokonano analizy szeregu mfsp pod kątem ich mocy opisowej dla zamodelowania pełnego zbioru funkcji komunikacyjnych modelu OSI/RM. Wykazano, że kryterium to nie różnicuje w zasadniczy sposób mfsp. Wyboru mfsp dla konkretnego protokołu dokonuje się więc w sposób nieformalny, zależny od subiektywnych ocen projektanta protokołu. Najbardziej znaczące jest w takim przypadku kryterium dostępności środowiska programowego wspomagającego przygotowanie i użycie formalnej specyfikacji protokołu oraz zakres planowanych zastosowań modelu. Z punktu widzenia zastosowania do oceny efektywności protokołów za pomocą symulacji wskazano jako najlepszą metodę numeryczne sieci Petriego.

W pracy [75] do porównania mfsp użyto metaspecyfikacji. Wybrano do tego celu metodę o nazwie SAN (State Architecture Notation [49]). Oceniane cechy mfsp obejmowały: orientację modelu (stanowa lub sekwencyjna), strukturę (jednoelementowa lub wieloelementowa), współbieżność lub sekwencyjność komponentów, rodzaj sygnałów (ciągłe, statyczne, impulsowe), typy komponentów (np. kolejki, zegary, maszyny abstrakcyjne), zapis specyfikacji (graficzny lub znakowy), filozofię specyfikacji (typu metaimplementacja lub monitor). Wskazano największą użyteczność metod: SAN i ESTELLE. Sieci Petriego zostały uznane za bardziej odpowiednie do analizy protokołów niż do rozpowszechnianej specyfikacji.

W pracy [1] oceny mfsp dokonano poprzez klasyfikację metod względem kompetencji opisu (opis architektury, zachowania, walidacyjny) oraz wymaganego dla określonego zastosowania poziomu abstrakcji opisu (specyfikacja abstrakcyjna, konkretna i implementacyjna) i odniesienie względem uzyskanych klas mfsp kryteriów ogólnych o różnej wadze, np. jednoznaczność syntaktyczna i semantyczna, nieimplikowanie nadmiaru opisu oraz 17 kryteriów specyficznych dla danych klas. Przykładowej ocenie poddano pięć metod (języków formalnej specyfikacji protokołów), z których najwyższej oceniono język PDIL [4] i LOTOS, a najniższej SDL.

W omawianych pracach zauważyć można pewną prawidłowość - wysoko oceniane są mfsp najlepiej znane zespołowi oceniającemu ze względu na

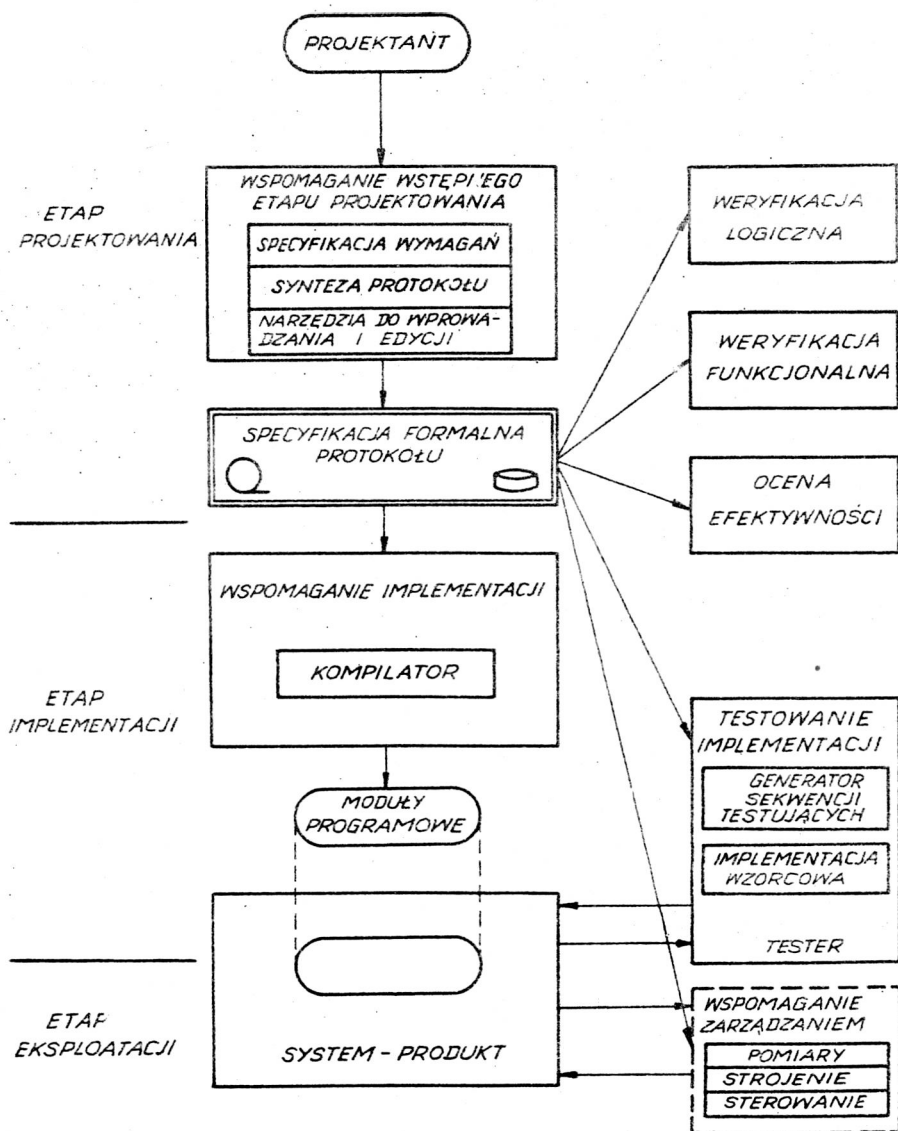
praktyczne doświadczenia w użyciu danej metody bądź też autorstwo danej metody. Podobne wrażenie odnosi się studiując materiały robocze grupy zajmującej się w ISO mfsp.

4. METODY I NARZĘDZIA INŻYNIERII PROTOKOŁÓW

We wstępie do tego artykułu powiedzieliśmy, że znajdujemy się u progu intensywnego rozwoju tzw. inżynierii protokołów. Korzystając z formalnej specyfikacji protokołów dostarcza ona metod i środków wspomagających proces projektowania, implementowania i eksploataowania protokołów sieci komputerowych. Środki, tj. narzędzia programowe, wspomagają czynności na wstępnym etapie projektowania, weryfikację logiczną i funkcjonalną, ocenę efektywności, implementację i jej testowanie, a także zarządzanie stacjami protokołów. Narzędzia te uważać możemy za elementy zintegrowanego systemu wytwórczego protokołów. System taki, inspirowany pracą [56], przedstawiono na rysunku 1. "Sercem" systemu jest model formalny protokołu, co do którego przyjmujemy założenie, że jest on w postaci zrozumiałej przez komputer. Omówimy pokrótce elementy tego systemu.

Wspomaganie wstępnego etapu projektowania obejmuje między innymi specyfikację wymagań użytkowych, syntezę protokołu i narzędzia do wprowadzania oraz edycji modeli formalnych. Specyfikacja wymagań użytkowych oczekuje na zastosowanie metod formalnych. Istniejące przykłady stosowania formalnej specyfikacji wymagań dla oprogramowania czasu rzeczywistego (np. [73]) wskazują na możliwość jej użycia w inżynierii protokołów. Dotychczas jednak brak jest takich przykładów. Synteza protokołów umożliwiła konstruowanie protokołów o z góry przyjętych cechach. Technika ta bazuje na dodawaniu krok po kroku przejść i stanów w modelu automatowym protokołu. Systemy wspomagające mają charakter interakcyjny (np. [83]). Narzędzia do wprowadzania i edycji modeli formalnych umożliwiają najczęściej interakcyjne wprowadzenie opisu z możliwością edycji i wstępnego sprawdzenia poprawności syntaktycznej opisu z równoczesną transformacją modelu do postaci przyjętej w danym systemie (np. system PROPER [34], system ANALYST [8], CARDE/ESTELLE [79]).

Weryfikacja logiczna polega na automatycznym sprawdzeniu logicznej poprawności specyfikacji. Przykładami wykrywanych błędów mogą być blokady, niekompletność specyfikacji. Bogaty przegląd weryfikowanych cech protokołów oraz metod weryfikacji znaleźć można w [61]. Ogólnie możemy wskazać, że najlepiej do weryfikacji nadają się modele o charakterze automatowym, sieci Petriego itp. Przykładami systemu weryfikacji logicznej protokołów mogą być: PROTEAN [11], VALIRA [77], ANA-



Rys. 1. Elementy zintegrowanego systemu wytwórczego protokołów bazującego na formalnej specyfikacji

Fig. 1. Components of an integrated development system based on formal specification

LYST [8], PAV [58], VEDA [32].

Weryfikacja funkcjonalna polega na sprawdzeniu, czy protokół realizuje wszystkie wymagania funkcjonalne, tzn. czy rezultaty jego działania są zgodne ze specyfikacją usług protokołu. Weryfikacja ta oparta jest na technice dowodzenia twierdzeń. Istnieje szereg półautomatycznych pakietów weryfikujących ogólnego zastosowania (np. AFFIRM, GIPSY, SPECIAL). Doświadczenia w weryfikacji protokołów za pomocą tych pakietów omówiono w [69]. Innym przykładem może być zastosowanie częściowo zautomatyzowanego interakcyjnego systemu dowodzenia twierdzeń, opartego na języku CIL [36] oraz systemu ANALYST i PROTEAN.

Ocena efektywności obejmuje predykcję efektywności protokołu uzyskiwaną poprzez analizę i symulację, a także konfigurowanie sieci z dostępnych komponentów. Analiza efektywności szeroko stosuje narzędzia z zakresu teorii kolejek i badań operacyjnych. Przegląd tych prac można znaleźć w [53]. Ostatnio pojawiły się prace stosujące do analizy efektywności protokołów modele formalne uzupełnione o parametry charakteryzujące efektywność protokołu w środowisku abstrakcyjnym i rzeczywistym (np. czasy wykonania operacji, opóźnienia, rozmiary i wielkość puli buforów) [55,8,29,52,78]. Interesujące rezultaty w zakresie oceny efektywności modelu OSI/RM uzyskano także przez zastosowanie kombinacji modeli kolejkowych i formalnych [35]. Modele formalne protokołów znajdują również zastosowanie w symulacji działania protokołów [9, 25]. Druga z tych prac jest równocześnie przykładem konfigurowania sieci. W pierwszej natomiast uzyskano model symulacyjny automatycznie z opisu formalnego. Zastosowanie modeli formalnych do oceny efektywności protokołów znajduje się we wczesnej fazie rozwoju.

Wspomaganie implementacji dotyczy automatycznej i półautomatycznej generacji kodu modułu programowego protokołu bezpośrednio z jego modelu formalnego. Przykładem może być generowanie modułów protokołu transportowego ISO [31], protokołu transportowego klasy 4 [3], wzorcowej implementacji protokołu transferu zbiorów [37]. We wszystkich jednak przypadkach pewne części programu musiały być kodowane ręcznie, np. we wzorcowej implementacji protokołu transferu zbiorów około 60% [56]. Warto odnotować, że do bezpośredniej implementacji, szczególnie w języku programowania wysokiego poziomu, najbardziej nadają się protokoły wyższych warstw architektury sieci.

Testowanie implementacji umożliwia testowanie zdolności implementacji do dostarczenia wymaganych usług oraz testowanie zachowania się implementacji w odniesieniu do wymagań narzuconych na komunikację przez odpowiednie standardy. Testowanie implementacji (ang. conformance testing) jest przedmiotem intensywnych prac normalizacyjnych ISO. Testowanie implementacji dotyczy w zasadzie obserwowalnego działania

zewnętrznego protokołu, tj. wejść i wyjść stacji protokołu, nie dotyczy natomiast sposobu wewnętrznej implementacji stacji. Od sposobu implementacji stacji zależy jej efektywność. Ocena efektywności implementacji planowaną początkowo jako element systemów testujących znajduje się aktualnie poza normalizacją. Formalna specyfikacja stosowana jest w testowaniu implementacji do generowania sekwencji sterujących (np. [27, 41, 62]) oraz wzorcowej implementacji stacji protokołu w testach [37]. Podkreślenia wymaga fakt, że metody formalnej specyfikacji protokołów odgrywają bardzo ważną rolę w procesie implementacji i testowania implementacji, nawet wtedy gdy nie stosuje się systemów automatyzujących te procesy.

Wspomaganie zarządzania obejmuje czynności związane z diagnostyką, pomiarami, strojeniem i sterowaniem stacjami protokołów. Modele formalne mogą stanowić w tym przypadku podstawę konstrukcji narzędzi wspomagających. Przykładem może być dynamiczna diagnostyka sieci komputerowych [7, 43, 66]. W zakresie zastosowania metod formalnych do wspomaganie pomiarów, strojenia i sterowania przykładem mogą być prace [20, 65, 67].

Jak można to zauważyć z powyższej dyskusji, zintegrowany system wytwórczy protokołów ma charakter hipotetyczny i mimo istnienia wielu elementów tego systemu zintegrowanie ich w jeden system jest sprawą przyszłości. Z powołanych wyżej przykładów najbliższy spekulacji tych postulatów jest system ANALYST, powstający w ramach projektu CUPID [81].

5. KIERUNKI PRZYSZŁYCH PRAC BADAWCZYCH

Przeprowadzona w artykule dyskusja metod formalnej specyfikacji protokołów i ich zastosowania w procesie wytwórczym protokołów upoważnia do wskazania kierunków przyszłych prac badawczych. Naszym zdaniem dotyczyć one będą:

- 1) zdefiniowania semantyki języków formalnej specyfikacji protokołów, co ułatwi i przyspieszy powstawanie systemów wspomagających ich użycie,
- 2) formalnego zdefiniowania i rozbudowy teorii dla zastosowań rozszerzonych, czasowych i stochastycznych sieci Petriego,
- 3) rozszerzeń języków programowania ADA i PROLOG dla zastosowania ich jako mfsp, a także w kierunku łączenia ich z innymi mfsp dla uzyskania efektywnych systemów wspomagających,
- 4) zastosowań mfsp do automatycznej implementacji i oceny efektywności protokołu,
- 5) uwzględnienia w mfsp modelowania wpływu czasu.

Dalszy rozwój inżynierii protokołów warunkowany jest postępiem w rozwiązywaniu dwóch zasadniczych w naszej opinii problemów: równowagi między mfsd oraz przekształcania modeli. Rozwiązanie pierwszego problemu wymagać będzie rozbudowy teorii i stworzenia automatycznych narzędzi dla translacji modelu formalnego z jednej metody na inną. Aby przybliżyć realizację tego problemu, zagadnienie to powinno być uwzględnione na etapie projektowania metod formalnej specyfikacji. Rozważyć także należy wprowadzenie metaspecyfikacji jako modelu podstawowego dla systemów wspomagających. Problem przekształcania modelu z zachowaniem jego cech logicznych i funkcjonalnych, ale na różnym poziomie abstrakcji opisu, wymaga rozbudowy teorii i stworzenia narzędzi dla formalnej transformacji modeli pomiędzy różnymi poziomami abstrakcji opisu, właściwymi dla określonych zastosowań modelu.

Szeroka tematyka wymaganych badań uświadamia fakt, że rozpowszechnianie zasad i systemów inżynierii protokołów będzie procesem długotrwałym.

LITERATURA

- [1] A h t i a i n e n A., et al., An Approach for Evaluating Formal Description Techniques, w: [23].
- [2] A n d e r s o n D.P., L a n d w e b e r L.H., Protocol Specification by Real-Time Attribute Grammars, w: [82], pp. 457-466.
- [3] A n d e r s o n D.P., L a n d w e b e r L.H., A Grammar-Based Methodology for Protocol Specification and Implementation, Proc. 9th Data Comm. Symposium, British Columbia, sept. 1985.
- [4] A n s a r t J.P., et al., Description, Simulation and Implementation of Communication Protocols Using PDIL, Proc. ACM SIGCOMM'83, Austin, Texas, USA, 1983.
- [5] A n s a r t J.P., et al., Software Tools for ESTELLE, w: [14], pp. 2-52 - 2-60.
- [6] A z e m a P., et al., Specification and Verification of Distributed Systems Using PROLOG Interpreted Petri Nets, Proc. 7th Int. Conf. on Software Engineering, 1984, pp. 510-518.
- [7] A y a c h e J.M., A z e m a P., D i a z M., Observer: a concept for on-line detection of control errors in concurrent systems, Proc. 9th Int. Symp. FTCS, Madison 1982.
- [8] B a r g h o u t i N., N o u n o u N., Y e m i n i Y., An Interactive Protocol Development Environment, w: [14], pp. 2-61 - 2-76.
- [9] B a u e r f e l d W.L., Performance Prediction of Computer Network Protocols, Proc. ICC'83, Boston, Mass., June 1983, pp. 1311

-1315.

- [10] Bearman M.Y., Wilbur-Ham M.C., Billington J., Specification and Analysis of the OSI Class 0 Transport Protocol, Proc. 7th ICCG'84, pp. 597-602.
- [11] Billington J., Wilbur-Ham M.C., Bearman M.Y., Automated Protocol Verification, w: [23].
- [12] Bochmann G.V., Sunshine C.A., Formal Methods in Communication Protocol Design, IEEE Trans. on Comm., vol. COM-28, no. 4, 1980, pp. 624-631.
- [13] Bochmann G.V., et al., Use of Prolog for Building Protocol Design Tools, w: [23].
- [14] Bochmann G.V., Sarikaya B. (eds), Sixth Int. Workshop on Protocol Specif. Testing and Verification, Montreal, June 10-13, 1986.
- [15] Briand J.P., et al., Executing LOTOS Specification, w: [14].
- [16] Brinksmä E., A Tutorial on LOTOS, w: [23].
- [17] Bruno G., Marchetto G., Proces-Translatable Petri Nets for the Rapid Prototyping of Process Control Systems, IEEE Trans. on Soft. Eng., vol. SE-12, no. 2, 1986, pp. 346-357.
- [18] Castanet E., Dupeux A., Guittou P., Ada, a well adapted Language for specification and implementation of protocols, w: [23].
- [19] Cellary W., Sajkowski M., Stroiński M., Defining a Transport Layer Using Numerical Petri Nets, Proc. 1st IEEE Int. Conf. on Computers and Applications, Peking 1984, pp. 353-360.
- [20] Cellary W., Stroiński M., Performance Parameters of Computer Network Protocols Referred to Their Formal Model, w: [82].
- [21] Courtiat J.P., Ayache J.M., Algayres B., Petri Nets are Good for Protocols, Computer Comm. Review, vol. 14, no. 2, 1984, pp. 66-74.
- [22] Diaz M., Modeling and Analysis of Communication and Cooperation Protocols Using Petri-Net Based Models, w: [68].
- [23] Diaz M. (ed), Protocol Specification, Testing and Verification, V, North-Holland, 1985.
- [24] Dickson G.J., Chazal P.E., Status CCITT Description Techniques and Application to Protocol Specification, Proc. of IEEE, vol. 71, no. 12, 1983, pp. 1346-1355.
- [25] Didic M., Wolfinger B., Simulation of a Local Computer Network Architecture Applying a Unified Modeling System. Computer Networks 6, 1982, pp. 75-91.

- [26] Dickson G.J., Wheeler G.R., A Comparison of Formal Description Techniques Proposed For International Standardization, Proc. 7th ICCG'84, 1985.
- [27] Favreau J.P., Linn R.J., Automatic generation of test scenarios from protocol specification written in Estelle, w: [14].
- [28] Fisher W., Sauer K.P., Denzel W., A Simulation Technique for Communications Protocols Based on a Formal Specification by SDL, w: [23].
- [29] Gressier E., A Stochastic Petri Net Model for Ethernet, w: [72].
- [30] Haas O., Formal Protocol Specification based on Attribute Grammars, w: [23].
- [31] Hansson H.A., Automatic Implementation of Formal Description of Communication Protocols, w: [23].
- [32] Jard C., Monin J.F., Groz R., Experience in Implementing X.250 (a CCITT subset of Estelle) in Veda, w: [23].
- [33] Kosarzycki K., Rajski J., Stroiński M., PROPER - Symulator numerycznych sieci Petriego i jego zastosowanie do badania protokołów, Konferencja "Sieci Komputerowe - Usługi, Protokoły, Modele", Wrocław 1985.
- [34] Kosarzycki K., Rajski J., Stroiński M., A Numerical Petri Net Based Simulator for Communication Protocols, Foundation of Control Engineering, vol. 10, no. 4, 1985, pp. 191-200.
- [35] Kritzing P.S., A Performance Model of the OSI Communication Architecture, IEEE Trans. on Comm., vol. COM-34, no. 6, 1986.
- [36] Krumm H., Drobniak O., Interactive Verification of Communication Software of the Basis of CIL, Computer Communication Review, vol. 14, no. 2, 1984.
- [37] Linn R.J., An Evaluation of the ICST Test Architecture after Testing Class 4 Transport, w: [82], pp. 611-622.
- [38] Linn R.J., The Features and Facilities of Estelle, w: [23].
- [39] Logrippo L., Simon D., Ural H., Executable Description of the OSI Transport Service in Prolog, w: [82], pp. 279-294.
- [40] Marsan M.A., Chiola G., Fumagalli A., An Accurate Performance Model of CSMA/CD Bus LAN, Proc. 7th European Workshop on Petri Nets, Oxford, UK, 1986, pp. 167-182.
- [41] Meer J., Derivation and validation of test scenarios based on the formal specification language Lotos, w: [14].
- [42] Merlin P., Farber D.J., Recoverability of Communica-

- tion Protocols: Implication of a theoretical study, IEEE Trans. Comm., vol. COMM-24, Sept. 1974, pp. 1036-1043.
- [43] M o l v a R., D i a z M., Observer: a run-time checking tool for local area networks, w: [23].
 - [44] O r t h W., Formalized Definition and Analysing of Protocols, w: T.Kalin (ed), EUTECO - European Teleinformatics Conference, North-Holland, 1983, pp. 619-627.
 - [45] O z s u M.T., Modeling and Analysis of Distributed Database Concurrency Control Algorithms Using an Extended Petri Net Formalism, Trans. on Soft. Eng., vol. SE-11, no. 10, 1985, pp. 1225-1240.
 - [46] P i a t k o w s k i T.F., An Engineering Discipline for Distributed Protocol Systems, Proc. of the Nato Advanced Study Institute: New Concepts in Multi-User Communications, 1980.
 - [47] P i a t k o w s k i T.F., I p L.K., H e D., State Architecture Notation an Simulation: A Formal Technique for Specification and Testing of Protocol Systems, w: [68].
 - [48] P i a t k o w s k i T.F., Protocol Engineering, Proc. ICC'83, Boston 1983.
 - [49] P i a t k o w s k i T.F., The State of the Art in Protocol Engineering, SIGCOM'86, Symp. Com. Architectures and Protocols, 1986.
 - [50] P o z e f s k y D.P., S m i t h F.D., A metaimplementation for Systems Network Architecture, IEEE Trans. on Comm., vol. COM-30, no. 6, 1982, pp. 1348-1355.
 - [51] R a y n e r D., H a l e R.W.S. (eds), Protocol Testing - Towards Proof?, Proc. INWG/NPL Workshop, vol. 1 Specification and Validation, vol. 2 Testing and Certification, NPL, 1981.
 - [52] R a z o u k R.R., P h e l p s C.V., Performance Analysis Using Timed Petri Nets, w: [82], pp. 561-576.
 - [53] R e i s e r M., Performance Evaluation of Data Communication Systems, Proc. of the IEEE, vol. 70, no. 2, 1982.
 - [54] R u d i n H., W e s t C.H. (eds), Protocol Specification, Testing and Verification, III, North-Holland, 1983.
 - [55] R u d i n H., An Improved Algorithm for Estimating Protocol Performance, w: [82].
 - [56] R u d i n H., An Informal Overview of Formal Protocol Specification, IEEE Communications Magazine, vol. 23, no. 3, March, 1985.
 - [57] R u d i n H., Time in Formal Protocol Specification, Proc. GI/NTG, Karlsruhe, 1985.
 - [58] S a b n a n i K.K., L a p o n e A.M., PAV-Protocol Analyzer and Verifier, w: [14].
 - [59] S a j k o w s k i M., S t r o i n s k i M., Evaluation of Formal Methods for Communication Protocol Specification from Proto-

- col Designer's Viewpoint, Proc. 5th Int. Conf. on Software Eng. for Telecomm. Switching Syst., Lund 1983, pp. 37-42.
- [60] S a j k o w s k i M., S t r o i ń s k i M., Projektowanie protokołów sieci komputerowych z zastosowaniem metod formalnej specyfikacji, Konferencja nt. "Podstawy metodologiczne syntezy i oceny systemów rozproszonego przetwarzania danych", Wrocław 1983.
 - [61] S a j k o w s k i M., Protocol Verification Techniques: Status Quo and Perspectives, w: [82].
 - [62] S a r i k a y a B., B o c h m a n n G.V., Synchronisation and Specification Issues in Protocol Testing, IEEE Trans. on Comm., vol. COM-32, no. 4, 1984, pp. 389-395.
 - [63] S c h w a r t z R.L., M e l l i a r - S m i t h P.M., From State Machines to Temporal Logic: Specification Methods for Protocol Standards, IEEE Trans. on Comm., vol. COM-30, no. 12, 1982, pp. 2486-2496.
 - [64] S i f a k i s J., Performance Evaluation Using Nets, Net Theory and Applications Lecture Notes in Computer Science, no. 84, Springer Verlag, 1979, pp. 307-319.
 - [65] S t r o i ń s k i M., External Measurement Method for Computer Network Protocol Performance Evaluation, przyjęte na CMG XIII Int. Conf. on Comp. Perf. Evaluation, San Diego 1982, opublikowane w CMG Transactions, 1983.
 - [66] S t r o i ń s k i M., A Method for the Diagnosis of Computer Network Protocols, Proc. III Inf. Conf. RELCOMEX'84, Książ 1984, ss. 235-241.
 - [67] S t r o i ń s k i M., Koncepcja eksperymentalnej metody oceny działania zdalnych i lokalnych sieci komputerowych, Szkoła Naukowa, Sieci Komputerowe'84, Bierutowice 1984.
 - [68] S u n s h i n e C.A. (ed), Protocol specification and verification II, North-Holland, 1982.
 - [69] S u n s h i n e C.A., et al., Specification and Verification of Communication Protocols in AFFIRM Using State Transition Models, IEEE Trans. on Softw. Eng., vol. SE-8, no. 5, 1982, pp. 460-489.
 - [70] S y m o n s F.J.W., Modelling and Analysis of Communication Protocols Using Numerical Petri Nets, Ph. D. Thesis, Univ. of Essex, 1978.
 - [71] S y m o n s F.J.W., The Description and Definition of Queueing Systems by Numerical Petri Nets, A.T.R., vol. 13, no. 2, 1980.
 - [72] Proc. of the International Workshop on Timed Petri Nets, Torino 1985.
 - [73] T s e n g J.S., S z y m a ń s k i B., S h i Y., P r y w e s N.S., Real-Time Software Life Cycle with the Model System, IEEE

- Trans. on Soft. Eng., vol. SE-12, no. 2, 1986.
- [74] U r a l H., P r o b e r t R.L., Automated Testing of Protocol Specifications and their Implementations, Computer Comm. Review, vol. 14, no. 2, 1984, pp. 149-155.
 - [75] V e n k a t r a m a n R.C., P i a t k o w s k i T.F., A Formal Comparison of Formal Protocol Specification Techniques, w: [23].
 - [76] V i s s e r s C.A., T e n n e y R.L., B o c h m a n n G.V., Formal Description Techniques, Proc. of the IEEE, vol. 71, no. 12, 1983, pp. 1356-1364.
 - [77] V u o n g S.T., H u i D.D., C o w a n D.D., A Tool for Protocol Validation via Reachability Analysis, w: [14].
 - [78] W o o d s i d e C.M., V i l l a c o r t e R.M., On Packet Buffering and Protocol Performance, w: [23].
 - [79] W o o d s i d e C.M., B u h r R.J.A., P e a r c e T., The CAEDE/ESTELLE graphical tool for formal definition of protocols, w: [14].
 - [80] W o u n g C.Y., D i l l o n T.S., F o r w a r d K.E., Analysis of Timing Aspects of Communication Computer Systems Using Timed Places Petri Nets, Proc. Seventh Int. Conf. on Computer Comm., Sydney 1984, pp. 585-590.
 - [81] Y e m i n i Y., N o u n o u N., CUPID: A Protocol Development Environment, w: [54].
 - [82] Y e m i n i Y., S t r o m R., Y e m i n i S., Protocol Specification, Testing, and Verification, IV, North-Holland, 1984.
 - [83] Z a f i r o p u l o P., et al., Towards Analyzing and Synthesizing Protocols, IEEE Trans. on Comm., vol. COM-28, no. 4, 1980, pp. 651-660.

PROTOCOL SPECIFICATION METHODS AND THEIR APPLICATION

In this paper the state of art of formal protocol specification is presented. Most popular methods are discussed and attempts to evaluate these methods are characterized. Next, components of a integrated protocol development system based on formal specification are described. Examples of systems are given. Finally, a future direction of research in the protocol engineering area are indicated.

МЕТОДЫ ФОРМАЛЬНОЙ СПЕЦИФИКАЦИИ ПРОТОКОЛОВ И ИХ ПРИМЕНЕНИЕ

В статье представляется состояние исследовательских работ связанных с развитием и применением методов формальной спецификации протоколов. Дается характеристику часто используемых методов и описывается работы касающиеся оценки этих методов. В дальнейшем представляется элементы интегрированной системы для выработки протоколов и дается примеры вспомогательных систем. В заключении намечается направления будущих исследовательских работ в области инженерии протоколов.

Jerzy WIETRZYCH[✉]sieć komputerowa,
standardy, wirtualne
terminaleUSŁUGI I PROTOKÓŁ WIRTUALNEGO TERMINAŁA
W MODELU POŁĄCZEŃ SYSTEMÓW OTWARTYCH ISO/OSI

W artykule przedstawiono aktualny stan prac nad standardem wirtualnego terminala według propozycji modelu odniesienia ISO/OSI. Omówiono zakres zastosowań standardu, model współpracy użytkowników sieci komputerowych przy jego użyciu i dostarczone w nim mechanizmy sterowania. Dalej przeanalizowano możliwości implementacji w rzeczywistych systemach i relacje do innych propozycji norm w rozważanym zakresie.

1. WSTĘP

Istotną częścią każdego systemu cyfrowego są urządzenia wejścia-wyjścia. Rodzaje i funkcje tych urządzeń są na ogół podobne we wszystkich systemach cyfrowych ogólnego przeznaczenia. Jednakże szczegółowe rozwiązania ich sprzęgów do komputera oraz inne składowe specyfikacji technicznej wykazują istotne rozbieżności w zależności od producenta. Łącząc systemy w sieci komputerowe napotyka się na problemy wynikające z chęci uzyskania współpracy poprzez sprzęgi sieciowe z dowolnymi komputerami. Różne aspekty tej współpracy były omawiane w materiałach I Krajowej Konferencji nt. sieci komputerowych [16, 17].

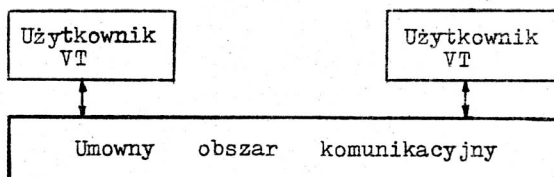
Niniejszy artykuł koncentruje się na problemach zobrazowania informacji w trakcie współpracy użytkownika z systemem cyfrowym, przy użyciu urządzeń wejścia-wyjścia podłączonych przez sieć komputerową. W ujęciu Modelu Odniesienia Połączeń Systemów Otwartych ISO/OSI problemy te są rozstrzygane w warstwie aplikacji przy użyciu usług i protokołu wirtualnego terminala. Przedmiotem tych rozstrzygnięć są zagadnienia takie jak: dostępne struktury danych, repertuary znaków graficznych, rodzaje sygnałów sterujących, rodzaje przemieszczeń informacji w polu

[✉] Instytut Cybernetyki Technicznej Politechniki Wrocławskiej

2. MODEL USŁUG

2.1. "Umowny obszar komunikacyjny"

Jak wskazano we wstępie, współpraca użytkowników, w tym także programów wirtualnych terminali odbywa się przez tzw. "Umowny obszar komunikacyjny" - CCA (Conceptual Data Area), co symbolicznie przedstawia rys. 1.



Rys. 1. Model usług
Fig. 1. Service Model

Przesyłanie danych według tego modelu polega na wpisywaniu danych do CCA przez jednego z użytkowników i odczytywaniu przez drugiego. CCA podzielono na następujące części:

CDS (Conceptual Data Store) - "Umowny obszar danych" - zawierający tzw. "Zespoły przekazywane" (Display Objects) służące do przekazywania danych;

CSS (Control, Signalling and Status Store) - "Pamięć wskaźników sterowania" - zawierająca tzw. "Zespoły sterowania" (Control Objects) reprezentujące statusy i sygnały wymienione pomiędzy użytkownikami;

ACS (Access Control Store) - "Pamięć praw dostępu" - zawierająca bieżące przypisanie praw dostępu do pozostałych części CCA;

DSD (Data Structure Definition) - "Pamięć definicji struktur" - zawierającą definicje struktur "Zespołów przekazywania", "Zespołów sterowania" oraz tzw. "Zespołów urządzeń" i innych parametrów środowiska podlegających negocjacji.

"Zespoły urządzeniowe" (Device Objects) stanowią reprezentację w CCA własności rzeczywistych urządzeń oraz procesów odwzorowujących zespoły wirtualne na urządzenia rzeczywiste.

W specyfikacji rozszerzonej obok wymienionych wyżej części składowych CCA występują także:

RIS (Reference Information Store) - "Pamięć informacji odniesienia" - zawierająca informacje o charakterze makroinstrukcji, możliwe do przywołania i umieszczenia w "Zespołach przekazywania" lub "Zespołach sterujących";

SIS (Structure Information Store) - "Pamięć struktur informacji" - stanowiąca część CDS i przechowująca informacje o dalszym podziale "Zespołów przekazywania" w trakcie trwania połączenia na bloki i pola.

zobrazowania, kroje czcionek, repertuar barw, stopni zaczerznienia, ochrona przed usunięciem i inne atrybuty przekazywanych danych.

Zgodnie z wymogami Modelu Odniesienia, podstawą rozważań na temat funkcjonowania połączeń wirtualnych terminali (VT) jest definicja pewnej abstrakcyjnej syntaktyki transferu specyfikującej protokół VT. Uznano jednak, że takie ogólne rozstrzygnięcie jest niewystarczające i do celów prac standaryzacyjnych skonstruowano dodatkowo abstrakcyjną strukturę danych - "Umowny obszar komunikacyjny" - tworzącą środowisko pracy wirtualnych terminali. W strukturze tej występuje szereg obiektów nawiązujących swymi własnościami do funkcji podzespołów terminali rzeczywistych.

Bieżące prace standaryzacyjne dotyczą bazowej klasy wirtualnych terminali, to jest takich, które służą do przekazywania tekstu, rozważa się przy tym specyfikację tzw. podstawową [12, 14] i rozszerzoną [13, 15]. Różnice między nimi ujawniają się w zawartości owego "Umownego obszaru komunikacyjnego". W ramach powyższych ogólnych podziałów operuje się jeszcze pewną ilością parametrów obiektów w tej abstrakcyjnej strukturze, pozwalających na dokładniejsze opisanie cech odwzorowywanych urządzeń rzeczywistych i programów aplikacyjnych. Należy się spodziewać, że przyszłe rozszerzenia standardu o możliwości przekazywania rysunków o dużej rozdzielczości będą się odbywały drogą wzbogacania zawartości "Umownego obszaru komunikacyjnego".

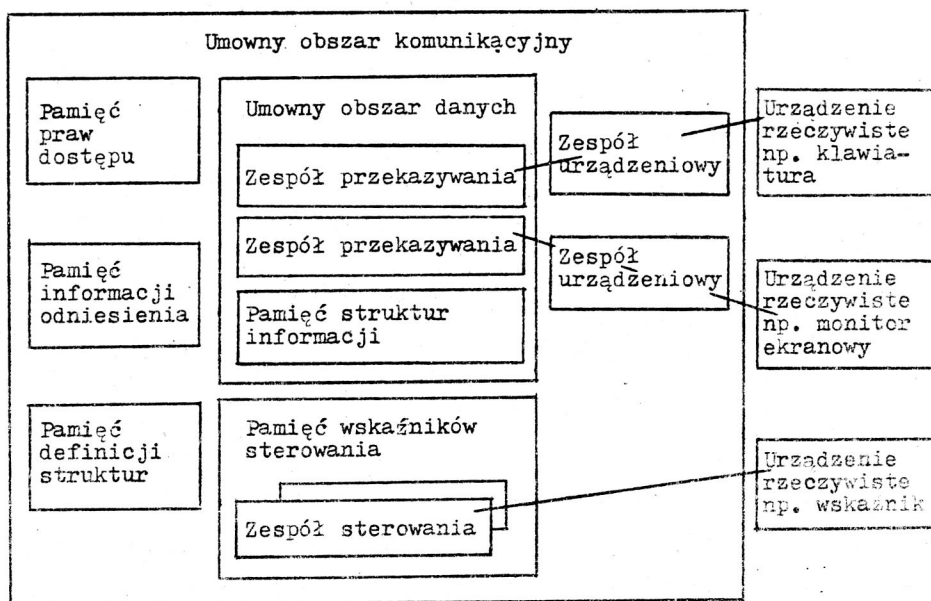
Aczkolwiek element modelu ISO/OSI realizujący usługi VT nie tworzy samodzielnej warstwy (umieszczony jest w warstwie aplikacji), to jednak sposób określenia standardu jest ściśle zgodny z ogólną konwencją opisu własności i funkcji warstw Modelu Odniesienia [6, 7]. Oznacza to, że specyfikuje się usługi świadczone aplikacji oraz protokół realizowany między stacjami VT. Dodatkowym elementem specyfikacji jest określenie budowy "Umownego obszaru komunikacyjnego" jako środowiska pracy wirtualnych terminali.

Prezentowany sposób rozwiązywania problemu ma aspekty nie tylko normalizacyjne. Analiza pracy urządzeń wejścia-wyjścia doprowadziła bowiem do skonstruowania nowych ogólnych mechanizmów sterowania działających na obiektach abstrakcyjnych a dających się interpretować w warunkach rzeczywistych. Taka zmiana podejścia nawiązuje do przełomu, jakim w programowaniu było przejście od języków assemblerowych do algorytmicznych.

Bloki stanowią realizację techniki podziału pola zobrazowania danych na tzw. "okna", szeroko rozpowszechnione w oprogramowaniu komputerów osobistych. Niestety używa się tu innej nazwy, gdyż termin "okno" w Modelu Odniesienia występuje w innym znaczeniu. Pola są strukturami wyznaczonymi przez przypisanie atrybutów przyznających prawa wprowadzania i kasowania danych w "Zespole przekazywania" partnerom dialogu. Wymienione wyżej części składowe CCA są powiązane między sobą w sposób przedstawiony na rys. 2.

Należy pamiętać o całkowitej umowności obiektów występujących w tym modelu. Oznacza to, że CCA i wszystkie części składowe istnieją tylko pozornie dla użytkownika usług VT i nie muszą mieć żadnej realizacji fizycznej. Stacja świadcząca usługi ma prawo konwersji interakcji użytkownika z obiektem w CCA na protokół terminala wirtualnego lub rzeczywistego bez konieczności tworzenia odpowiadających tym obiektom struktur danych w pamięci komputera. Ważne jest tylko zachowanie zgodności syntaktyki abstrakcyjnej realizowanej przez użytkownika i stację [8].

Budowa wewnętrzna wszystkich części CCA składa się na środowisko VT i jest opisana przez zespół parametrów, które pokrótce omówimy w następnych rozdziałach.



Rys. 2. Podział CCA
Fig. 2. CCA Partitioning

2.2. "Zespoły przekazywania"

"Zespół przekazywania" jest to jedno, dwu lub trójwymiarowa tablica, w której mogą być umieszczone symbole graficzne (znaki). Zespół ten jest określony przez parametry specyfikujące: nazwę, liczbę wymiarów, możliwości adresowania elementów, dozwolone repertuary znaków, możliwość ich kasowania, rodzaj praw dostępu, liczbę dostępnych barw, stopni zaczernienia itp. Przeznaczeniem zespołu jest przekazywanie informacji tekstowej pomiędzy użytkownikami usług VT. Najczęściej ma on strukturę dwuwymiarową nawiązującą do możliwości powszechnie znanych terminali takich jak dalekopis lub monitor ekranowy.

Ponieważ rzeczywiste terminale mają różne własności w zakresie dostępu do poszczególnych pozycji w polu zobrazowania, również własności "Zespołów przekazywania" są w tym zakresie dokładniej określane przez parametry adresowania. Podają one: maksymalne wartości współrzędnych, możliwość przeskakiwania o wskazaną liczbę pozycji do przodu lub wstecz, możliwość wyznaczania bezwzględnych wartości współrzędnych oraz cofania licznika pozycji w trakcie wprowadzania danych. W sytuacji, gdy dana operacja jest zabroniona parametry przyjmują wartości jednoznacznie określające to ograniczenie.

Liczba "Zespołów przekazywania" w CCA jest wyznaczona przez wynegocjowaną strukturę dialogu między użytkownikami usług VT. Może to być dialog dwukierunkowy naprzemienny tu noszący nazwę synchronicznego lub dwukierunkowy równoczesny - asynchroniczny. W pierwszym przypadku wystarcza jeden "Zespół przekazywania" w CCA, w drugim potrzebne są dwa. Ze strukturą tego dialogu związane są parametry określające prawa dostępu użytkowników do zespołów. Parametr ten może przyjmować jedną z trzech wartości "WAVAR", "WACI" lub "WACA". Wartość "WAVAR" oznacza, że prawo dostępu podlega przekazywaniu pomiędzy użytkownikami usług - zatem jest to dialog synchroniczny. Wartość "WACI" oznacza, że prawo wprowadzania do danego zespołu jest na stałe przypisane temu użytkownikowi, który zainicjował połączenie. Trzecia wartość "WACA" daje prawo dostępu użytkownikowi, który połączenie zaakceptował. Wartości "WACI" i "WACA" muszą być przypisane dwu różnym zespołom i są używane w dialogu asynchronicznym.

Szczególnego omówienia wymaga repertuar przekazywanych znaków graficznych. Użytkownicy mają do dyspozycji równocześnie jeden podzbiór 94 znaków graficznych identyfikowany jako G0 i opcjonalnie trzy dalsze podzbiory po 94 znaki graficznie identyfikowane jako G1, G2 i G3. Dodatkowo można także używać dwu podzbiorów znaków sterujących C0 i C1.

Identyfikatory G0, G1, G2, G3, C0 i C1 są to nazwy ogólne, które mają znaczenie w technice kodowania w trakcie trwania połączenia. Fak-

tyczna zawartość każdego z tych podzbiorów jest ustalana podczas negocjacji parametrów VT, polegającej na wskazaniu, który z zarejestrowanych przez upoważniony urząd repertuarów znaków ma być użyty jako odpowiednio GØ, G1 itd. [2, 3].

Wśród zarejestrowanych repertuarów znajdują się także dwa pokazane w tabeli 1. Tabela ta pokazuje równocześnie następujące propozycje pochodzące z różnych norm ISO:

- tzw. "łaciński" repertuar znaków graficznych, wg normy ISO 646 wersja IRV;
- uzupełniający repertuar znaków do komunikacji tekstowej, wg normy ISO 6937/2;
- przyporządkowanie powyższych repertuarów do podzbiorów GØ i G1 wirtualnego terminala;
- przyporządkowanie kodu ośmiobitowego do powyższych podzbiorów, co w Modelu Odniesienia jest domeną warstwy prezentacji, tu wyznaczonego przez złożenie numeru kolumny i wiersza w tabeli.

Należy w tym miejscu wyjaśnić, że podzbiory G2 i G3 jeśli zostaną wyneogcowane, będą obsadzały w tabeli kodowej tę samą pozycję co GØ i G1.

Przełączanie na te alternatywne repertuary symboli graficznych odbywa się przez poprzedzenie właściwego ciągu kodowego sekwencją ESC, I, F, gdzie I oraz F są bajty identyfikujące o którą z alternatyw chodzi. Początkowe przyporządkowanie repertuarów jest takie jak przedstawiono w tabeli.

Przedstawione w tabeli repertuary GØ i G1 łącznie pozwalają na tworzenie reprezentacji liter alfabetów wszystkich europejskich języków wywodzących się z łaciny. Specjalne znaki narodowe uzyskuje się przez złożenie obrazów tzw. "znaku diakrytycznego" z kolumny 12 i znaku z podzbioru GØ. Na przykład polską literę "ę" tworzy się przez złożenie znaku o kodzie 12/14 (ogonek) i znaku o kodzie 6/5 (e). Jest to możliwe dzięki specjalnej definicji zawartości kolumny 12, w myśl której wydruk umieszczonego tam znaku nie przesunął pozycji pisania i pozwala na naniesienie na tę samą pozycję kolejnego normalnego znaku. Dokument ISO [5] zawiera pełną analizę jakie złożenia w jakich językach występują i dopuszcza do użytku tylko przeanalizowane.

2.3. Inne części "Umownego obszaru komunikacyjnego"

Spośród innych części "Umownego obszaru komunikacyjnego" omówimy jeszcze "Zespoły sterowania" i "Zespoły urzędzeniowe".

"Zespoły sterowania" służą do przekazywania między użytkownikami usług VT informacji sterującej. Tak jak i inne umowne zespoły w CCA opisane są przez parametry precyzujące: nazwę, prawa dostępu, typ i strukturę przekazywanych danych oraz aktywność mechanizmu tzw. "prze-

wiersz	kolumna															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			0	Q	P	'	p					°		-	Ω	K
1			1	A	Q	a	q				i	±	'	1	Æ	æ
2			"	2	B	R	o	r			¢	2	'	Ⓡ	Ⓡ	Ⓡ
3			#	3	C	S	c	s			£	3	'	Ⓒ	a	ø
4				4	D	T	d	t			\$	X	'	™	H	h
5			5	5	E	U	e	u			¥	μ	-	♩		γ
6			&	6	F	V	f	v			π	'			U	u
7			'	7	G	W	g	w			§	•	•		l	l
8			(	8	H	X	h	x			÷	••			Ł	ł
9			)	9	I	Y	i	y			¢	'			⊙	φ
A			*	:	J	Z	j	z			«	»	°		Ⓔ	Ⓔ
B			+	:	K	[	k	{			«	»	°		Ⓔ	B
C			,	<	L	\	l				←	1/4	=	1/8	p	p
D			-	=	L	]	m	}			↑	1/2	"	3/8	F	£
E			.	>	N	^	n	~			→	3/4	u	5/8	n	n
F			/	?	O	-	o				↓	z	'	7/8	'n	

pozycja podzbioru C0 podzbiór łaciński uzupełniający re-
 pozycja pod- zbioru G0,G2,G3 pertuar znaków
 zbioru C1 pozycja podzbiorów G1, G2, G3

Tabela 1. Zbiory znaków graficznych do przesyłania tekstów
 Table 1. Graphic character sets for text communication

kaźnika". Parametr przekaźnik (trigger), jeśli ma wartość "selected" jest interpretowany jako żądanie natychmiastowego przekazywania wszelkich zmian wartości w "Zespole sterowania" i innych obiektach CCA drugiemu partnerowi dialogu.

"Zespoły urządzeń" stanowią uzupełnienie opisu środowiska wirtualnego terminala o własności urządzeń rzeczywistych i procesu sprzęgającego je z obiektami CCA. Obecność tych zespołów w modelu pozwala na uwzględnienie dodatkowych wymagań narzucanych na proces przekazywania danych.

Parametry "Zespołu urządzeniowego" mogą określać repertuar dostępnych znaków, ich krój, ilość dostępnych barw i stopni zaciemnienia, ograniczenia czasu wprowadzania i długości komunikatów, znaki kończące komunikat itp. Taki układ parametrów może budzić pewne wątpliwości gdyż niektóre z nich powtarzają informacje związane z "Zespołem przekazywania". Istniejący nadmiar może być w pewnych warunkach trudny do zinterpretowania, szczególnie wtedy parametry "Zespołu urządzeniowych" i "Zespołów przekazywania" określają różne wartości tych samych własności, np. różne repertuary znaków. Należy się spodziewać, że dalsze prace i specyfikacja formalna modelu rozstrzygną te problemy w jednoznaczny sposób. Pozostałe parametry unikalne w środowisku stanowią istotne uzupełnienie opisu i dają się dobrze interpretować w warunkach rzeczywistych.

2.4. Profile wirtualnych terminali

Jak widać z dotychczasowych rozważań zawartość CCA tworząca środowisko wirtualnych terminali opisywana jest przez dużą liczbę parametrów. Negocjacja i interpretacja ich wartości jakkolwiek jest możliwa, może narażać wielu problemów implementacyjnych. Dlatego też zdecydowano się na tworzenie specyfikacji łącznych, przypisujących równocześnie wartości wielu parametrów tzw. profili. Profile takie będą rejestrowane przez upoważniony do tego urząd administracyjny. Wartości niektórych parametrów w profilu może być pozostawiona do uzgodnienia między użytkownikami usług, jednakże większość z nich musi mieć wartości stałe.

Przykładem profilu, który będzie zarejestrowany może być przedstawiony niżej profil domniemany w trybie dialogu synchronicznego. Funkcjonalnie odpowiada on własnościom dalekopisu. W specyfikacji użyto następujących oznaczeń:

- { } - ograniczenie bloku syntaktycznego,
- , - oddzielenie parametrów,
- - oddzielenie komentarzy,
- . - zakończenie bloku syntaktycznego.

Display-object =

```
{display-object-name = DISPLAY-OBJECT,
 object-access-right = "WAVAR",
 dimensions = 2,
  x-dimension =
  {x-bound = 80,
   x-forward = "no constraint",
   x-backward = 0,
```

```
--Zespół przekazywania
--Nazwa zespołu
--Prawo dostępu przenośne
--Liczba wymiarów
--Definicja wymiaru x
--Maksymalna wartość x
--Skoki do przodu
--Skoki wstecz zabronione
```

```

    x-absolute = "not permitted",    --Bezwzględne x zabronione
    x-window = 1,                    --Brak możliwości cofania
},
y-dimension =                        --Definicja wymiaru y
{
    y-bound = "unbounded",           --Brak ograniczenia y
    y-forward = "no constraint",      --Skoki do przodu
    y-backward = 0,                  --Skoki wstecz zabronione
    y-absolute = "not permitted",     --Bezwzględne y zabronione
    y-window = 1,                    --Brak możliwości cofania
},
repertoire-capability = 1,           --Jeden repertuar znaków
emphasis-capability = 1,             --Jedna jaskrawość
erasure-capability = "no",           --Brak kasowania
foreground-colour-capability = 1,    --Jeden kolor znaków
background-colour-capability = 1,    --Jeden kolor tła
},
device-object =                      --Zespół urządzeń
{
    device-name = DEVICE,             --Nazwa urządzenia
    device-default-CO-initial-value = "TRUE" --Stan początkowy
    device-repertoire-assignment = ESC 2/8 4/0, --Znaki ISO 646 IRV
    device-font-assignment = "device dependent", --Krój dowolny
    device-emphasis-assignment = "normal", --Jaskrawość normalna
    device-foreground-colour-assignment = "device dependent",
    device-background-colour-assignment = "device dependent",
    device-minimum-line-length = 1,    --Minimalna długość linii
    device-minimum-page-length = 1,    --Minimalna długość strony
    device-display-object = DISPLAY-OBJECT, --Związany zespół przek.
    device-termination-event-list = NUL, --Brak znaków ograniczaj.
    device-termination-timeout = 0,0,  --Brak ograniczenia czasu
    device-termination-length = 0,     --Brak ograniczającej dług.
    device-termination-control-object = NUL --Brak zespołu sterowania
}.

```

3. USŁUGI WIRTUALNEGO TERMINALA

Definicja usług VT określa sposób współpracy aplikacji ze stacją umożliwiającą dostęp do określonego wcześniej "Umownego obszaru komunikacyjnego". Współpraca ta polega na wzajemnym oddziaływaniu użytkownika ze stacją przy pomocy elementarnych operacji (primitive). Tak jak i w innych warstwach Modelu Odniesienia operacje te mogą mieć następujący charakter:

- żądania (request) z aplikacji do stacji VT;

Tabela 2. Zestawienie usług wirtualnego terminala
Table 2. Summary of Virtual Terminal Services

operacja	typ	znaczenie
Udogodnienie ustanawiania połączenia		
VT-ASSOCIATE	potwierdzana	Ustanowienie połączenia
Udogodnienie likwidacji połączenia		
VT-RELEASE	potwierdzana	Likwidacja połączenia
VT-U-ABORT	niepotwierdzana	Zerwanie połączenia przez użytkownika usług
VT-P-ABORT	tylko wskazanie	Zerwanie połączenia przez realizatora usług
Negocjacja typu przełączanie profilu		
VT-SWITCH- -PROFILE	potwierdzana	Przełączanie profilu
Negocjacja konwersacyjna		
VT-START-NEG	potwierdzana	Rozpoczęcie negocjacji
VT-END-NEG	potwierdzana	Zakończenie negocjacji
VT-NEG-INVITE	niepotwierdzana	Pytanie o parametry
VT-NEG-OFFER	niepotwierdzana	Przedstawienie parametrów
VT-NEG-ACCEPT	niepotwierdzana	Przyjęcie parametrów
VT-NEG-REJECT	niepotwierdzana	Odrzucenie parametrów
Przesyłanie danych		
VT-DATA	niepotwierdzana	Dane
Sterowanie dostarczaniem		
VT-DELIVER	niepotwierdzana	Żądanie przekazania danych
VT-ACK-RECEIPT	niepotwierdzana	Potwierdzenie odebrania
Zarządzanie prawami dostępu		
VT-GIVE-TOKENS	niepotwierdzana	Przekazanie prawa dostępu
VT-REQUEST- -TOKENS	niepotwierdzana	Żądanie prawa dostępu

- wskazania (indication) ze stacji VT do aplikacji współpracującej;
- odpowiedzi (response) z aplikacji współpracującej do stacji VT;
- potwierdzenia (confirmation) ze stacji VT do aplikacji.

W zależności od typu usługi wykorzystuje się wszystkie cztery rodzaje operacji (usługa potwierdzana), tylko żądanie i wskazanie (usługa niepotwierdzana) lub tylko wskazanie do obu aplikacji.

Zestaw usług VT pozwala na tworzenie, likwidowanie i awaryjne zrywanie połączeń, negocjowanie środowiska, wymianę danych oraz sterowanie dostępem i dostarczaniem danych. Każda z operacji przekazuje związane z danym typem usługi parametry, które dokładniej określają obiekt, którego dotyczy. Pełny repertuar dostępnych usług przedstawiono w tabeli 2.

Najbardziej interesujące są usługi wymiany danych. Umożliwiają one wprowadzanie informacji do obiektów CCA: "Zespołów przekazywania", "Zespołów sterujących", "Pamięci informacji odniesienia" lub "Pamięci struktur informacji". Analogicznie stacja VT może zgłaszać zmiany wartości tych obiektów do aplikacji.

Warunkiem wprowadzenia danych jest uzyskanie prawa dostępu do danego obiektu. Jak wskazano w rozdziale 2.1 prawa takie są przechowywane w "Pamięci praw dostępu" i mogą być przekazywane pomiędzy współpracującymi użytkownikami przy zastosowaniu usług zarządzania prawami dostępu lub są przypisane na trwałe jednemu z nich.

Z wymianą danych wiążą się usługi kontroli dostarczenia. Istnieje tu kilka możliwości sterowania: współpraca bez sterowania dostarczaniem, współpraca z podstawową metodą sterowania dostarczaniem lub współpraca z usługą "kwarantanny" danych. W pierwszym przypadku przekazywanie danych partnerowi dialogu wynika z wewnętrznych wymogów realizatora usług. Dodatkowo przekazywanie to może być wymuszane przez niektóre operacje użytkownika, takie jak np. przekazanie praw dostępu lub wprowadzenie zmiany do "Zespołu sterowania" z parametrem przekaznik = "selected". Współpraca z podstawową metodą sterowania dostarczaniem przebiega podobnie, ale użytkownik może jawnie korzystać z usługi VT-DELIVER w celu wymuszenia przekazania zmian naniesionych do CCA. W metodzie z usługą "kwarantanny" danych stacja VT zamraża przekazywanie wszystkich zmian w CCA do momentu, gdy użytkownik zażąda tego jednym ze sposobów jak w metodach poprzednich.

4. PROTOKÓŁ WIRTUALNEGO TERMINAŁA

Protokół wirtualnego terminala jest definiowany poprzez abstrakcyjną Maszynę Protokołu Wirtualnego Terminala (VTM), która realizuje procedury używane w tym protokole. VTM komunikuje się z użytkownikiem

Tabela 3. Zestawienie komunikatów protokołu VT
Table 3. Summary of VT Protocol Data Units

kod	nazwa	znaczenie	podzbiór
Ustanawianie połączenia			A, B, C
ASQ	VT-ASSOCIATE-REQ	Zgłoszenie połączenia	
ARQ	VT-ASSOCIATE-RESP	Odpowiedź na zgłoszenie	
Likwidacja połączenia			A, B, C
RLQ	VT-RELEASE-REQ	Żądanie rozłączenia	
RLR	VT-RELEASE-RESP	Odpowiedź na żądanie rozłączenia	
AUQ	VT-U-ABORT	Zerwanie połączenia przez użytkownika usług	
APQ	VT-P-ABORT	Zerwanie połączenia przez realizatora usług	
Przesyłanie danych			A, B, C
NDQ	VT-DATA	Dane normalne	
UDQ	VT-UNCONTROLLED-DATA	Dane niezależne	
Sterowanie dostarczaniem			A, B, C
DLQ	VT-DELIVER	Żądanie przekazania danych	
DAQ	VT-ACK	Potwierdzenie otrzymania danych	
Sterowanie dostępem			A, B, C
GTQ	VT-GIVE-TOKEN	Przekazanie prawa dostępu	
RTQ	VT-REQUEST-TOKEN	Żądanie prawa dostępu	
Negocjacja typu przełączanie profilu			B, C
SPQ	VT-SWITCH-PROFILE-REQ	Żądanie przełączenia profilu	
SPR	VT-SWITCH-PROFILE-RESP	Odpowiedź na żądanie przełączenia profilu	
Negocjacja konwersacyjna			C
SNQ	VT-START-NEG-REQ	Zgłoszenie rozpoczęcia negocjacji	
SNR	VT-START-NEG-RESP	Odpowiedź na rozpoczęcie negocjacji	
ENQ	VT-END-NEG-REQ	Zgłoszenie końca negocjacji	
ENR	VT-END-NEG-RESP	Odpowiedź na zgłoszenie końca negocjacji	
NIQ	VT-NEG-INVITE	Pytanie o wartości parametrów	
NOQ	VT-NEG-OFFER	Przedstawienie wartości parametrów	
NAQ	VT-NEG-ACCEPT	Przyjęcie wartości parametrów	
NJQ	VT-NEG-REJECT	Odrzucenie wartości parametrów	

usług za pośrednictwem operacji omówionych w rozdziale 3. Efektem przetwarzania zleceń od użytkownika oraz przyczyną ich wydawania przez VTPM jest wymiana komunikatów protokołu z drugą współpracującą VTPM w innym systemie. Maszyna protokołu składa się z jednostek funkcjonalnych utworzonych przez logiczne pogrupowanie udogodnień udostępnianych użytkownikom. Pełny zestaw jednostek funkcjonalnych i związanych z nimi komunikatów protokołu przedstawiono w tabeli 3.

W celu ułatwienia implementacji jednostki funkcjonalne łączone są w większe zespoły o narastającym stopniu złożoności. W chwili obecnej rozważa się trzy podzbiory: podstawowy (VT-A), z negocjacją typu przełączanie profilu (VT-B) i z negocjacją typu konwersacyjnego (VT-C). Przynależność poszczególnych jednostek funkcjonalnych do podzbioru można prześledzić w tabeli 3.

Zasadnicze różnice między podzbiórami polegają na możliwościach negocjowania środowiska VT. W podzbiorze podstawowym profil jest narzucony jednym z parametrów komunikatu VT-ASSOCIATE. Wybór wartości parametrów tego narzuconego profilu może być dokonany tylko z listy propozycji przesyłanej w tym samym komunikacie. W podzbiorze VT-B możliwe jest także wybieranie samego profilu. Podzbiór VT-C pozwala na indywidualne negocjowanie wartości wszystkich parametrów CCA.

Struktura każdego z komunikatów jest w dokumentach ISO [13, 15] opisana przy użyciu notacji ASN 1. Z faktu tego wynika także podstawowa technika kodowania - wyznaczona przez reguły kodowania do ASN 1 [11]. Inne alternatywne techniki mogą być ustalone przez wykorzystanie usług warstwy prezentacji [8, 9].

Integralną częścią specyfikacji protokołu wirtualnego terminala są tablice stanów przedstawiające akcje i operacje podejmowane pod wpływem zdarzeń napływających do VTPM.

5. WSPÓŁPRACA Z WARSTWĄ SESJI I PREZENTACJI

Usługi wirtualnego terminala są realizowane w oparciu o usługi warstwy prezentacji, w tym także te, które przejęła ona z warstwy sesji. Jednakże operacje nawiązywania i likwidowania połączeń są wykonywane za pośrednictwem CASE (Common Application Service Elements) - czyli Wspólnego Zestawu Usług Aplikacyjnych, który to element Modelu Odniesienia znajduje się w warstwie aplikacji.

Dokumenty ISO dokładnie określają sposób odwzorowania żądań wymienianych pomiędzy aplikacją a realizatorem usług VT na usługi CASE, warstwy prezentacji i co za tym idzie sesji. Stosuje się tu następujące techniki odwzorowania parametrów operacji VT i komunikatów warstw niższych:

- transparentne przejście do użytku VT komunikatu warstwy prezentacji, np. REQUEST TOKEN, GIVE TOKEN (przejęte tą samą techniką z warstwy sesji);
- przekazanie parametru lub ich zestawu do zakodowania w komunikacie warstwy prezentacji, np. przyczyna rozłączenia w komunikacie P-U-ABORT;
- przekazanie komunikatu VT wraz ze wszystkimi parametrami w polu danych użytkowych komunikatu warstwy prezentacji, np. NDQ w komunikacie DATA, UDQ w komunikacie TYPED DATA;
- podział zestawu parametrów operacji usług VT na dwie grupy przekazywane w jednym komunikacie warstwy prezentacji, ale przy odpowiednim użyciu obu technik jak wyżej.

Zestawienie przyporządkowania komunikatów protokołu VT przekazywanych w polu danych użytkowych operacji CASE i warstwy prezentacji przedstawiono w tabeli 4. Uwzględnienie komunikatów przejmowanych transparentnie oraz analiza zależności pomiędzy CASE a warstwą prezentacji pozwala na sformułowanie kilku wniosków.

Operacje usług CASE i prezentacji	Typy przekazywanych komunikatów protokołu VT
A-ASSOCIATE	ASQ, ASR
A-U-ABORT	Parametr przyczyny zerwania
P-DATA	NIQ, NOQ, NAQ, NJQ, NDQ
P-TYPED-DATA	SPQ, SPR, SNQ, SNR, ENQ, ENR, UDQ, DLQ, DAQ

Tabela 4. Odzworowanie komunikatów protokołu VT na operacje usług CASE i prezentacji

Table 4. VT Protocol PDU-s mapping to CASE and presentation primitive

Najważniejszy z nich dotyczy operacji nawiązywania i likwidacji połączeń VT. Operacje te są odwzorowywane w usługi CASE, a dalej warstwy prezentacji i sesji czwartą z wymienionych wyżej metod. Powoduje to, że w sesyjnym komunikacie CONNECT zanurzony jest w polu danych użytkowych komunikat CONNECT warstwy prezentacji, w nim z kolei ASSOCIATE CASE, a w nim ASQ wirtualnego terminala. Parametry operacji VT rozdzielane są odpowiednio na poszczególne komunikaty warstw. Analogicznie zbudowany jest komunikat RELEASE, co w sumie daje efekt nierozłączności czasów istnienia połączeń aplikacji, VT, CASE, prezentacji i sesji. Jest to rozwiązanie odmienne niż w warstwach komunikacyjnych, gdzie czasy istnienia połączeń w poszczególnych warstwach nie muszą się ze sobą pokrywać.

Inne spostrzeżenie dotyczy rezygnacji z użycia w protokole VT przyspieszonej drogi przesyłania (**EXPEDITED**) w warstwie prezentacji oraz danych uzgadniania (**CAPABILITY**). Do celów sterowania wykorzystuje się na rzecz połączeń VT dane niezależne (**TYPED DATA**).

Dokładniejsza analiza zależności pomiędzy usługami VT, warstwą prezentacji i sesji pozwala na stwierdzenie, że definicja VT określając abstrakcyjną syntaktykę transferu, wymusza użycie pewnego tylko zestawu usług warstw niższych. Autonomia warstwy prezentacji jest ograniczona, gdyż opisanie syntaktyki protokołu VT przy użyciu notacji ASN1 narzuca użycie konkretnej syntaktyki transferu zgodnie z regułami kodowania podanymi w dokumencie [11]. Inne syntaktyki konkretne mogą być użyte tylko opcjonalnie. Również z warstwy sesji wykorzystuje się wybrane tylko funkcje i usługi.

Odnosnie realizowanego protokołu narzucenie zestawu funkcji, nierozłączność czasu istnienia połączeń oraz przenikanie całych komunikatów powoduje, że stacja VT, CASE, prezentacji i sesji wspólnie realizują jeden protokół funkcjonalny - kodowany przy użyciu elementów protokołów tych warstw. W praktyce większość komunikatów tego protokołu ma swoje źródło i dociera w formie operacji usług do aplikacji. Wewnątrz warstwy sesji i prezentacji zamykają się tylko fazy negocjacji, gdy ustalony jest zestaw używanych usług tych warstw.

6. MOŻLIWOŚCI IMPLEMENTACJI STANDARDU

Omówione standardy znajdują się obecnie w stadium propozycji roboczych. Ich implementacja będzie celowa po uzgodnieniu wszystkich szczegółów. Warto jednak przeanalizować możliwości zastosowania usług i protokołu wirtualnego terminala w różnych rodzajach sieci komputerowych. Mogą one być wykorzystane do współpracy między aplikacją a terminalem, między dwoma aplikacjami lub dwoma terminalami. Najbardziej interesujący jest przypadek współpracy terminala i programu aplikacyjnego.

W chwili obecnej stosuje się różne rodzaje sprzężenia terminali z komputerami:

- poprzez wspólny obszar pamięci operacyjnej np. w komputerach osobistych,
- poprzez połączenia trwałe z transmisją szeregową lub równoległą,
- poprzez sieci lokalne,
- poprzez sieci komutacji obwodów,
- poprzez sieci komutacji pakietów.

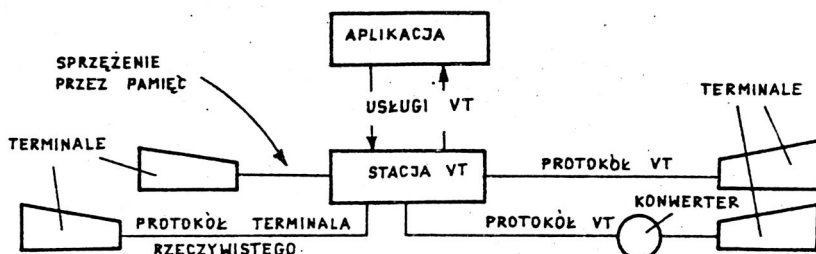
Z siecią komputerową mamy do czynienia w trzech ostatnich przypadkach, jednakże największe korzyści można odnieść traktując wszystkie przy-

padki jednakowo i umieszczając stację VT wewnątrz systemu operacyjnego komputera. Dostęp do niej z poziomu procesu aplikacyjnego powinien być realizowany w konwencji usług VT. Pozwoliłoby to na tworzenie oprogramowania użytkowego w jednolity sposób, niezależny od rzeczywistych reguł współpracy komputera z terminalami. Uzależnienie od własności rzeczywistych powinno być uwzględnione w takim zakresie na jaki pozwala negocjacja środowiska VT.

Nieco bardziej skomplikowana wydaje się być kwestia zastosowania protokołu VT. Pierwszy z wymienionych sposobów sprzężenia nie używa żadnego protokołu w znanym nam sensie. Natomiast w przypadkach pozostałych mamy do wyboru trzy możliwości:

- konwersję usług VT na protokół terminala rzeczywistego bezpośrednio w systemie operacyjnym świadczącym usługi,
- budowę stacji pośredniczących dokonujących konwersji między protokołem VT i protokołem terminala rzeczywistego,
- budowę terminali tzw. inteligentnych realizujących wprost protokół VT.

Zestawienie wszystkich tych rozwiązań przedstawiono na rysunku 3.



Rys. 3. Rodzaje sprzężeń terminala z aplikacją
Fig. 3. Spectrum of terminal to application connections

Pierwsze z wymienionych rozwiązań może być zastosowane tylko wtedy, gdy charakterystyka podłączonego terminala jest dokładnie znana i nie wymaga się negocjowania parametrów. Może to zatem dotyczyć połączeń trwałych lub sieci lokalnych, gdzie typ terminali daje się dokładnie ustalić na podstawie adresu.

W rozległych sieciach publicznych, gdzie mamy do czynienia z terminalami różnych firm, pożądane jest zastosowanie metody drugiej lub trzeciej. Warto jednak zwrócić uwagę na to, że istnieją inne alternatywne standardy, np. rekomendacja X3 CCITT specyfikująca PAD terminali start-stopowych, do której zdalny dostęp jest możliwy zgodnie z rekomendacją X29. Propozycja ISO zawiera profil VT równoważny tej rekomen-

dacji, pozostaje jednak sprawą otwartą ustalenie miejsca, w którym dokonana zostanie konwersja standardów. Można tego dokonać zarówno wewnątrz PAD-u jak i w systemie operacyjnym komputera użytkownika lub też w stacji pośredniczącej.

W trakcie ustanawiania współpracy terminala rzeczywistego z odległą aplikacją pojawia się problem doboru odpowiedniego profilu VT do tego połączenia. Porównanie własności popularnych terminali i profili oferowanych w standardzie pozwala na stwierdzenie, że po tej stronie połączenia nie powinny się pojawiać istotne problemy. Nieco inaczej wygląda sytuacja po stronie aplikacji.

W chwili obecnej w eksploatacji jest bardzo duża liczba programów aplikacyjnych napisanych w sposób ściśle przystosowany do określonego terminala firmowego i wykorzystujący ich specyficzne własności nie dające się odwzorować na funkcje VT. Brak elastyczności tych programów może poważnie ograniczyć możliwości ich przystosowania do pracy w sieciach komputerowych.

Z doświadczeń wyniesionych z budowy sieci komputerowych MSK i SKJS/2 wynika, że najistotniejsze ograniczenia pojawiają się w związku z użyciem w aplikacjach - takich jak na przykład podsystem MOP systemu GEORGE3 - specyficznych mechanizmów synchronizacji oraz brakiem możliwości negocjowania wymiarów pola zobrazowania. Protokoły zrealizowane w tych sieciach sankcjonowały zastaną sytuację i wprowadzały do użycia w sieci mechanizmy wymuszane przez aplikacje. Wprowadzenie do tych sieci protokołu według propozycji ISO będzie wymagało poruszenia powyższych problemów ponownie. Inne elementy protokołu VT nie powinny wnieść większych trudności. Warto zauważyć, że np. repertuar znaków i technika kodowania używana w protokole współpracy terminali w sieci MSK jest podzbiorem propozycji ISO przedstawionej w tabeli 1.

7. UWAGI KOŃCOWE

Omówione standardy dotyczą terminali, których organizacja pozwala na przekazywanie ograniczonej liczby symboli graficznych - zwykle używanych do przekazywania tekstu i rysunków o niewielkiej rozdzielczości. Należy się jednak spodziewać, że będą również podejmowane prace nad standaryzacją przekazywania informacji graficznej o dużej rozdzielczości. Obecna bazowa klasa usługi i protokołu VT stanie się wówczas szczególnym przypadkiem specyfikacji ogólniejszej.

Według obecnej koncepcji usługi i protokołu VT są mocno powiązane z innymi normami dotyczącymi przekazywania informacji tekstowej, co sygnalizowano w rozdziale 2.1. W niektórych przypadkach prowadzi to do powstania nadmiarów i trudności w ustalaniu jak będzie wyglądał tekst

po przekazaniu partnerowi dialogu. Przykładem takich problemów może być ustalanie jak wpłynie na przedstawienie informacji przesłanie znaków sterujących z podzbioru C1 według normy ISO 6429, skoro adresowanie pozycji w "Zespole przekazywania" dokonywane jest w sposób charakterystyczny dla VT, tj. przez aktualizację parametru określającego pozycję przekazywania. Rozstrzygnięcie tych i innych problemów sygnalizowanych w rozdziale 2.3 nastąpi prawdopodobnie w toku dalszych prac standaryzacyjnych i formalizacji specyfikacji.

Pomimo pewnych wątpliwości zgłoszonych w artykule wydaje się, że uzyskane dotychczas rezultaty stanowią istotny krok na drodze do budowy w pełni heterogenicznych sieci komputerowych oraz uniezależniania oprogramowania użytkowego od technicznych własności urządzeń wejścia-wyjścia komputerów.

Literatura

- [1] ISO646 - ISO 7 bit coded character set for information interchange. 1983-07-01.
- [2] ISO2022 - ISO 7 bit and 8 bit coded characters sets - Code extension techniques. 1985-11-01.
- [3] ISO2375 - Procedure for registration of escape sequences. 1985-11-01.
- [4] ISO6429 - ISO 7 bit and 8 bit coded characters sets - Additional control functions for character imaging devices. 1983-05-01.
- [5] ISO6937/2 - Coded character sets for text communication - Part 2: Latin alphabetic and non alphabetic graphic characters. 1983-12-15.
- [6] ISO7498 - Open Systems Interconnection - Basic Reference Model.
- [7] ISO/TR8509 - Open Systems Interconnection - Service Conventions.
- [8] ISO/DP8822 - Connection Oriented Presentation Service Definition. 1984-10-15.
- [9] ISO/DP8823 - Connection Oriented Presentation Protocol Specification. 1984-10-15.
- [10] ISO/DIS8824 - Specification of Abstract Syntax Notation One ASN 1. 1985-06-06.
- [11] ISO/DIS8825 - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN 1). 1985-06-06.
- [12] ISO/TC97/SC21/WG5 N79 - Virtual Terminal Service - Basic Class. Part I. November 1985.
- [13] ISO/TC97/SC21/WG5 N42 - Virtual Terminal Service - Basic Class. Part II. September 1985.
- [14] ISO/TC97/SC21/WG5 N80 - Virtual Terminal Protocol - Basic Class. January 1986.
- [15] ISO/TC97/SC21/WG16 N527 - Virtual Terminal Protocol - Basic Class. Part II. February 1985.
- [16] Sieci komputerowe. Usługi, protokoły, modele. Część 1. Wydawnictwo Politechniki Wrocławskiej, Wrocław 1985.
- [17] Sieci komputerowe. Usługi, protokoły, modele. Część 2. Wydawnictwo Politechniki Wrocławskiej, Wrocław 1986.

THE VIRTUAL TERMINAL SERVICE AND PROTOCOL IN THE ISO/OSI
OPEN SYSTEMS INTERCONNECTION MODEL

In the paper the state of the art in the area of the Virtual Terminal standards development of OSI/RM is presented. The major part of the paper describes service model and control procedures proposed in the standard. Finally possibility of implementation and relations to CCITT standards are indicated.

СЕРВИС И ПРОТОКОЛ ВИРТУАЛЬНОГО ТЕРМИНАЛА
В МОДЕЛИ ВЗАИМОДЕЙСТВИЯ ОТКРЫТЫХ СИСТЕМ

В статье представлено актуальное состояние работ над стандартом виртуального терминала по рекомендованной модели ISO/OSI. Описан предел применений стандарта, модель содействия пользователей вычислительной сети при его применении и его механизмы управления. Далее представлен анализ возможных имплементаций в реальных системах и отношений к другим рекомендованным нормам ISO и CCITT.

Witold ABRAMOWICZ^x

sieć komputerowa,
rozpowszechnianie informacji,
system CODI

O NOWYM SPOSOBIE UDOSTĘPNIANIA INFORMACJI W SIECIACH KOMPUTEROWYCH

CODI - Context Oriented Dissemination of Information stanowi nową koncepcję automatycznego rozpowszechniania informacji w sieciach komputerowych. CODI jest systemem zaprojektowanym dla:

- automatycznego rozpowszechniania heterogenicznych dokumentów w sieciach komputerowych przez ich producenta do konsumentów,
- ułatwiania konsumentowi oceny przesłanych dokumentów,
- umożliwienia konsumentowi organizowania i pomownego wyszukiwania dokumentów.

Każdemu dokumentowi towarzyszą trzy nowo zdefiniowane klasy relewantności wyrażające korelację pomiędzy dokumentem a zainteresowaniami konsumenta.

1. Rozpowszechnianie informacji

Wyróżniam dwa sposoby udostępniania informacji:

1. Wymianę informacji. Nadawca wysyła wieloznaczne informacje do wieloznacznego grona odbiorców. Może wypowiedzieć się o każdej wysyłanej informacji oraz zna zapotrzebowanie informacyjne każdego z odbiorców. Wymiana nie wymaga żadnych specjalnych kwalifikacji od nadawców i odbiorców.

2. Rozpowszechnienie informacji.

a/ Ze względu na ilość informacji nadawca nie może wypowiedzieć się o każdej z osobno wysyłanych informacji.

b/ Liczba potencjalnych odbiorców uniemożliwia nadawcy znajomość zapotrzebowan informacyjnych każdego z nich.

W procesie rozpowszechniania zachodzi jedna lub obie sytuacje, dlatego realizowany jest on tylko w systemie zautomatyzowanym.

Wymiana informacji może być wspomagana przez systemy poczty komputerowej np. /3/, /4/, /5/.

Komputerowo wspomagane rozpowszechnienie informacji, z objęciem skomputeryzowaną obsługą zarówno odbiorców jak i nadawców, nie zostało jeszcze rozwiązane. Dlatego proponuję metody budowy systemu in-

^x Akademia Ekonomiczna, Poznań

formacyjnego umożliwiającego /2/:

- formalizację opisu zainteresowań odbiorców,
- formalne opisywanie rozpowszechnianych informacji /pojedynczo rozważane informacje nazywamy dokumentami/,
- rozdzielanie informacji na podstawie opisów zainteresowań odbiorców i dokumentów,
- pełne uzasadnienie decyzji o rozdziale,
- przesłanie odbiorcom dokumentów wraz z uzasadnieniem,
- prezentację użytkownikom dokumentów oraz towarzyszącego uzasadnienia.

Model o wyliczonych właściwościach określam mianem kontekstowo zorientowanego rozpowszechniania informacji /context oriented dissemination of information - CODI/, tj. rozpowszechniania informacji w kontekście zainteresowań odbiorców.

2. Model

Punktem wyjścia dla sformułowanego modelu jest wyodrębnienie podmiotów i przedmiotów modelu.

Określam dwa rodzaje podmiotów: producenta i konsumenta.

Producent rozpowszechnia informacje na podstawie jej analizy oraz znanych mu zainteresowań odbiorców. Konsument /odbiorca/, na podstawie zainteresowań przedstawionych producentowi, otrzymuje informacje. Zakładam, że w realizacji modelu - w systemie informacyjnym - zarówno producent jak i konsument dysponują komputerami. Przyjmuję dalej, że komputery te połączone są w sieci, w której dostępny jest system poczty komputerowej.

Przedmiotami modelu są: rozpowszechnianie informacji oraz zainteresowanie konsumentów. Zakładam, że informacja jest dostępna w formie czytelnej dla systemu liczącego oraz w trakcie jej rozpowszechniania może pominać problematykę ochrony danych, a także, że pomiędzy poszczególnymi informacjami nie zachodzą żadne relacje, np. żadna z nich nie jest kontynuacją innej. Taką informację nazywam dokumentem. Korzystając z możliwości opisu dokumentów z użyciem znanych metod automatycznego indeksowania tekstów stosowanych w systemach wyszukiwawczych, przedstawiam dokument jako zbiór wybranych termów tezaursu opatrzonych wagami /profil dokumentu/. W tej samej formie opisuję zainteresowania konsumentów. Profil konsumenta jest zbiorem wybranych termów tezaursu przedstawiających zainteresowania konsumenta. Waga każdego z termów precyzuje jego znaczenie dla konsumenta. Profile konsumentów gromadzone są przez producenta.

Producent ma do zrealizowania trzy zadania:

1. Tworzenie profili dokumentów.

2. Organizowanie profilów konsumentów.

3. Znaleźnienie dla każdego z pojedynczo analizowanych dokumentów profilów konsumentów, dla których jest on relatywny.

Dla tworzenia profilów dokumentów zalecam jedną z klasycznych metod indeksowania. Jej wybór wynika z empirycznych badań metod indeksowania oraz zaproponowanie przeze mnie metody określania relewantności dokumentu.

Producent otrzymuje do konsumentów za pośrednictwem poczty komputerowej sformułowane przez nich profile konsumentów. Zadaniem producenta jest, przed przystąpieniem do rozpowszechniania dokumentów, uaktualnienie struktury profilów konsumentów.

W trosce o dobrą obsługę konsumentów dokładnie precyzuje informacje o relewantności. Określam ją ściślej niż w klasycznych systemach wyszukiwawczych, ponieważ każdy z dokumentów rozpowszechniany jest przez producenta tylko jeden raz tak, że późniejsze zmiany profilów konsumentów nie mają żadnego wpływu na określanie relewantności raz rozesłanego dokumentu. Konsument, dla uniknięcia przekazywania mu dokumentów z niepotrzebną informacją i dla spowodowania otrzymywania oczekiwanych dokumentów, powinien być zainteresowany przyczynami przekazania mu każdego z dokumentów. Ich analiza powinna wnieść wkład do precyzyjniejszego sformułowania profilu konsumenta. Dlatego wyróżniam trzy klasy informacji o relewantności. Stanowią one odpowiedź na pytanie:

1. Jak ważny jest dokument dla profilu konsumenta?
2. Jakie są przyczyny relewantności dokumentu dla profilu konsumenta?
3. Jakie są potencjalne przyczyny relewantności dokumentu dla profilu konsumenta?

Pierwsza informacja o relewantności jest miarą podobieństwa pomiędzy profilami dokumentu i konsumenta. W formie wartości jednej zmiennej precyzują znaczenie dokumentu dla profilu konsumenta.

Informacja ta nie uzasadnia jednak przekazania dokumentu konsumentowi. Niezbędne jest wskazanie tych termów, które przyczyniły się do tego, że dokument został uznany jako relewantny dla profilu konsumenta. Są to termy należące równocześnie do profilów: konsumenta i dokumentu. Wkład każdego z tych termów do relewantności dokumentu dla profilu konsumenta - pierwszej informacji o relewantności - jest wartością funkcji jego wag w obu profilach. Zbiór tych wartości z przyporządkowanymi im termami nazywam drugą informacją o relewantności.

Dla konsumenta, chcącego udoskonalić swój profil, ważne są nie tylko rzeczywiste przekazanie dokumentu, ale także te potencjalne.

Ważnym jest, aby dowiedzieć się o tym, co mogłoby się przyczynić do otrzymania dokumentów z podobną informacją jak ta, którą zawiera przekazany właśnie dokument. Wskazań tych dostarczają termy należące do profilu dokumentu i nie należące do profilu konsumenta. Zbiór tych termów z ich wagami w profilu dokumentu nazywam trzecią informacją o relewantności.

Ponieważ producent nie może zakładać, że wybranym przez niego momencie, konsumenci będą gotowi odebrać przeznaczony dla nich dokument, korzysta ze systemu poczty komputerowej. Przesyła dokument do skrzynek pocztowych wybranych konsumentów. Informacje o relewantności dostarczane są konsumentowi razem z dokumentem. Jeżeli konsument uzna dokument za istotny dla siebie, to może go włączyć, korzystając z informacji o relewantności, do lokalnego zbioru dokumentów. Z pomocą systemu wyszukiwawczego pożądane dokumenty mogą być w miarę potrzeby znów znalezione.

Zmieniając pod wpływem informacji o relewantności swój profil, konsument korzysta z edytora profili. Udoskonalony profil konsumenta zostaje przekazany producentowi za pośrednictwem poczty komputerowej. Przed przystąpieniem do rozpowszechniania dokumentów producent odbiera nowe wersje profili konsumentów ze skrzynki pocztowej i włącza je do struktury danych.

3. Realizacja modelu

Ideę CODI sprawdziłem realizując system informacyjny ATHENE /1/. Na przykładzie rozpowszechniania informacji bibliograficznej zwerifikowałem przydatność zaproponowanego modelu. Możliwości wynikające z rozpatrywanej koncepcji są niewątpliwie szersze i mogą obejmować np. rozpowszechnianie informacji patentowej, oprogramowania, serwisu prasowego o treści ściśle odpowiadającej potrzebom każdego z odbiorców.

Implementacji dokonałem na komputerze Liliith. Językiem programowania była Modula2. Komputery połączone są w lokalną sieć komputerową Magnet wywodzącą się z koncepcji Ethernet.

Literatura:

- [1] W. Abramowicz, Documentation of ATHENE, Institut fuer Informatik, ETH Zurych, 1985.
- [2] W. Abramowicz, Rozproszony kontekstowo zorientowany system informacyjny, praca doktorska, Politechnika Wrocławska 1985.
- [3] D.K. Brotz, Laurel Manuel, XEROX Palo Alto Research Center, 1981.

- [4] B.I. Edison, H. Raag, R. Smith, INTERPOST - An Experimental International Electronic Message System, w: J. Salz /ed./, Computer Communication: Incoreasing Benefits for Society. Proceedings of the Fifth International Conference. Atlanta 1980.
- [5] E. Motolese, et al., MAIL2K: A Computer Mail Package for Local Applications, w: R.P. Uhlig /ed./, Computer Message Systems, Proceedings of the IFIP TC-6 International Symposium on Computer Message Systems, Ottawa, Canada, 1981.

CODI - A NEW CONCEPT TO THE DISSEMINATION OF INFORMATION IN COMPUTER NETWORKS

CODI: Context Oriented Dissemination of Information brings a new concept to the dissemination of information in computer networks.

CODI was developed to support the application of systems for:

- the automatic dissemination of homogeneous documents /like software packets, bibliographic information, patens, news and so on/ in computer networks from a information producer /software of bibliographic information distributor, Patent Office .../ located in one of the network nodes, to all other nodes of the information consumers /like scientist, engineer/ in the context of their information needs,
- the consumers' interacitve evaluation of documents automatically selected by the producer,
- the organization and re-retrieval of documents in the consumer's own, single-user information retrieval system.

CODI makes use of to-way communication between producer and the consumers in the computer network. The producer provides each consumer with context- relevant documents and relevance information in three new categories.

О НОВОМ СПОСОБЕ ДОСТУПА К ИНФОРМАЦИИ В КОМПЬЮТЕРНЫХ СЕТЯХ

CODI - Context Oriented Dissemination of Information является новой концепцией автоматического распространения информации в компьютерных сетях. CODI является системой проектирования для:

- автоматического распространения гетерогенических документов в компьютерных сетях их производителем потребителю,
- облегчения потребителю оценки посланных документов,
- предоставления потребителю возможности организации и повторного отыскивания документов.

Каждому документу сопутствуют три заново определены класса релевантности выражающие корреляцию между документом и заинтересованностью потребителя.

Janusz BARCZYŃSKI*

sieć komputerowa,
weryfikacja protokołów,
CSP, system dowodowy,
logika niezmienników

OPIS I WERYFIKACJA PROTOKOŁÓW - PODEJŚCIE AKSJOMATYCZNE

Przedstawiono aksjomatyczne podejście do formalnego opisu i weryfikacji protokołów sieci komputerowej oraz omówiono krótko, opracowaną przez autora, technikę szczegółową zgodną z tym podejściem.

1. WSTĘP

Ostatni okres charakteryzuje się znacznym wzrostem zainteresowania rozmaitych środowisk, zarówno akademickich jak i przemysłowych, problematyką formalnego opisu oraz weryfikacji protokołów komunikacyjnych. Stąd, obserwowany burzliwy rozwój tej dyscypliny informatyki. Temat jest zarówno interesujący jak trudny, a uzyskane wyniki dalekie są od ostatecznych rozwiązań. Koncentracja wysiłków dotyczy przede wszystkim technik formalnego opisu konkretnych protokołów. Często chodzi o protokoły dla warstwowego modelu OSI (Open Systems Interconnection) [6], który stał się powszechnie przyjętym punktem odniesienia dla prowadzonych prac badawczych i konkretnych realizacji. Przykładem są tu techniki ESTELLE oraz LOTOS będące przedmiotem prac standaryzacyjnych w ISO [2,4]. Natomiast w mniejszym stopniu rozwinięte są techniki weryfikacji protokołów. W wielu przypadkach, autorzy przedstawiają partykularne techniki weryfikacji oraz ilustrują ich przydatność na prostych przykładach (rozmaite wersje protokołu Alternating Bit). Konieczne jest więc zintensyfikowanie poszukiwań także w zakresie technik weryfikacji konkretnych protokołów. Często trudno jest dobrać technikę weryfikacji do istniejącej techniki opisu. Stąd celowym wydaje się opracowywanie od razu jednolitych podejść do opisu i weryfikacji protokołów, w tym protokołów dla warstwowego modelu OSI [1].

* Centrum Obliczeniowe, Politechnika Wrocławska, Wrocław

2. PODEJŚCIE AKSJOMATYCZNE

Protokoły komunikacyjne są klasą struktur programowych, charakteryzującą się przede wszystkim następującymi cechami:

- równoległe realizowanie procedur wymiany komunikatów przez wiele komunikujących się składowych (modułów protokołu),
- przestrzenne rozproszenie składowych, uniemożliwiające korzystanie ze wspólnych pamięci,
- zawodna wymiana komunikatów spowodowana błędami wprowadzanymi przez media komunikacyjne,
- warstwowa organizacja komunikacji i złożoność procedur wymiany komunikatów.

Ogólna teoria analizy programów, w tym także równoległych, jest dość dobrze rozwinięta. Znałe są tu rozmaite podejścia, w tym przede wszystkim operacyjne, denotacyjne, algebraiczne oraz aksjomatyczne. Wymienione cechy protokołów nie pozwalają bezpośrednio skorzystać z wyników tej teorii, natomiast daje się zaadoptować odpowiednia metodologię.

Przedstawione niżej podejście mieści się w ramach, ogólnie rozumianego, podejścia aksjomatycznego. Mówiąc krótko, polega na opisie protokołów jako pewnej klasy programów równoległych, weryfikacja zaś to dowodzenie własności takich programów. Konstruowanie odpowiedniej techniki odbywa się zgodnie z następującą metodologią:

- (1) Zdefiniować język opisu protokołów - L1.
- (2) Zdefiniować język opisu własności protokołów - L2. Formuły tego języka wyrażają twierdzenia o własnościach protokołów, zapisywanych w języku L1.
- (3) Zdefiniować semantykę języka L1 (np. metodą operacyjną lub denotacyjną) i języka L2.
- (4) Podać system dowodowy T, złożony z aksjomatów i reguł, dla dowodzenia własności protokołów, dokładniej formuł języka L2.
- (5) O systemie T wykazać, że:
 - (a) proces dowodzenia nie wyprowadza z klasy własności prawdziwych w semantyce (3) - niesprzeczność T,
 - (b) każda własność prawdziwa w semantyce (3) i wyrażalna w języku L2 posiada dowód w T - relatywna zupełność T.
- (6) Zademonstrować przydatność formalizmu do opisu i weryfikacji konkretnych protokołów komunikacyjnych. Między innymi pokazać, że język $L1 \cup L2$ oraz semantyka (3) są na tyle bogate, że wyrażenia L1 "dobrze" opisują wybraną klasę protokołów a formuły L2 "dobrze" opisują własności tej klasy protokołów.

Konstruowanie techniki formalnego opisu i weryfikacji protokołów zgodnie z podaną metodologią w ogólnym przypadku, kiedy to mamy do czynienia z

wielowarstwowym modelem protokołów (np. modelem OSI), jest zadaniem niezwykle złożonym. Wtedy w szczególności:

- język L1 powinien posiadać konstrukcje pozwalające na niezależny, modularny opis protokołów, to znaczy warstwa po warstwie z uwzględnieniem złożonej struktury wewnątrzwarstwowej,
- język L2 powinien umożliwiać opis własności usług poszczególnych warstw protokołów, w sposób maskujący nieistotne własności wewnątrzwarstwowe,
- system dowodowy T powinien pozwalać na niezależną, modularną budowę dowodów na kształt uzyskanych opisów.

Następny rozdział zawiera krótkie omówienie, opracowanej przez autora, techniki [1] zgodnej z powyższym podejściem.

3. TECHNIKA

Język opisu protokołów stanowi uogólnienie, uwzględniające specyficzne cechy protokołów, notacji CSP (Communicating Sequential Processes) [3] Hoare'a. Protokoły opisywane są jako cykliczne kaskadowe programy równoległe. Składowe takich programów połączone są simpleksowymi kanałami umożliwiającymi synchroniczną wymianę komunikatów i sygnałów. Dla opisu własności oraz weryfikacji wykorzystano zmienne historii. Wartościami zmiennych historii są sekwencje komunikatów przesyłanych oraz odbieranych, zwane historiami odp. wyjściowymi oraz wejściowymi. Weryfikacja protokołów dotyczy dowodzenia ich własności niezmienniczych. Inne własności zostaną uwzględnione w kolejnych generacjach techniki.

3.1. Język opisu protokołów

Protokoły komunikacyjne opisywane są jako cykliczne kaskadowe programy równoległe, będące programami równoległymi, o następującej ogólnej postaci:

$$Q \equiv [Q_1 \# Q_2 \# \dots \# Q_{n-1} \# Q_n] \quad ,$$

gdzie Q_i jest procesem sekwencyjnym z nazwą $A::P$ (A jest nazwą procesu P) lub programem równoległym. Podstawową zasadą semantyczną jest tu rendez-vous: do wymiany komunikatów (odp. sygnałów synchronizacyjnych) pomiędzy dwoma procesami występującymi w Q może dojść wtedy, gdy jeden z nich chce przesłać komunikat (odp. sygnał synchronizacyjny) a drugi chce go odebrać. Instrukcje komunikacyjne mają postać instrukcji

• przesłania
do procesu A :

• odbioru
od procesu B :

(a) $\langle A!T(\vec{e});S \rangle$, $\langle B?T(\vec{x});R \rangle$, (komunikatów)

(b) $\langle A!T;\text{skip} \rangle$, $\langle B?T;\text{skip} \rangle$, (sygnałów synchronizacyjnych)

gdzie: T jest typem wymienianych komunikatów (a) lub sygnałów synchronizacyjnych (b); $\bar{e} = e_1, \dots, e_m$ jest listą wyrażeń - układ ich wartości jest komunikatem, który ma być przesłany; $\bar{x} = x_1, \dots, x_m$ jest listą zmiennych - układ ich wartości jest komunikatem odebrany; S oraz R są programami, w których występują zmienne historii i mogą mieć postać

$$S \equiv h.out := h.out \circ [\bar{e}] \quad , \quad R \equiv h.in := h.in \circ [\bar{x}] \quad ,$$

gdzie $h.out$ oraz $h.in$ są zmiennymi historii odp. wyjściowej oraz wejściowej a $\circ$ jest operatorem konkatenacji.

Jeżeli (a) oraz (b) są zgodnymi parami (pojęcie to jest podobne do odpowiedniego pojęcia z CSP) to wykonanie (a) (odp. (b)) stanowi atomową akcję równoważną, w sensie transformacji wartości zmiennych, wykonanie programu $\bar{x} := \bar{e}; S; R$ (odp. skip) .

Program kaskadowy Q charakteryzuje się pewną regularnością struktury komunikacyjnej, mianowicie składowa Q_i może komunikować się wyłącznie z jedną, gdy istnieje tylko jedna, lub dwoma sąsiednimi składowymi. Ilustruje to dobrze graf komunikacyjny (por. przykład 1 i rys. 1), w którym wierzchołki reprezentują programy składowe a krawędzie - łączące je kanały; kanał utożsamiany jest ze zgodną parą instrukcji komunikacyjnych. Wreszcie, program kaskadowy Q zwany jest cyklicznym, jeżeli wszystkie występujące w nim procesy sekwencyjne z nazwami mają postać:

$$A :: * [b_i; c_i \rightarrow R_i] \dots [b_m; c_m \rightarrow R_m] \quad ,$$

gdzie: b_i jest wyrażeniem logicznym; c_i jest instrukcją komunikacyjną lub skip - jednak co najmniej jedna z c_i jest instrukcją komunikacyjną; w programie R_i mogą również występować instrukcje komunikacyjne.

Przykład 1. Następujący kaskadowy program równoległy:

$$PC \equiv [USER^1 \# SYSTEM_n \# USER^T] \quad ,$$

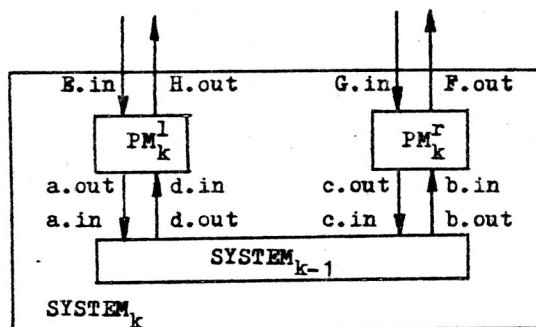
$$\text{gdzie} \quad SYSTEM_k \equiv [PM_k^1 \# SYSTEM_{k-1} \# PM_k^T] \quad ; \quad 1 \leq k \leq n \quad ,$$

jest modelem zespołu złożonego z n warstw protokołów ($SYSTEM_n$) z którego korzystają użytkownicy $USER^1$ oraz $USER^T$. Program równoległy PM_k^z , $z \in \{1, r\}$, jest tu modelem modułu protokołu warstwy k -tej. W kontekście modelu OSI moduł protokołu może być rozpatrywany jako składowa jednostki warstwy k -tej (np. 4-tej, transportowej), która to składowa obsługuje punkt końcowy (connection endpoint) jednego połączenia warstwy k -tej. Formalny opis protokołu warstwy k -tej polega na zdefiniowaniu programów PM_k^1 oraz PM_k^T . W wielu wypadkach można przyjąć, że programy te skomponowane są z trzech procesów sekwencyjnych modelujących: moduł nadawczy protokołu,

moduł czasomierzy oraz moduł odbiorczy protokołu.

Graf komunikacyjny programu $SYSTEM_k$ z zaznaczonymi zmiennymi historii wyjściowych (z sufiksem out) oraz wejściowych (z sufiksem in) przedstawia rys. 1. Zmienne te wykorzystamy dalej.

Rys. 1.



3.2. Język opisu własności protokołów

Formuły poprawnościowe wyrażające twierdzenia o własnościach niezmienniczych programów (protokołów) mają postać:

$$(\#) \quad \langle Q, \varphi \rangle \text{ inv } I \quad ,$$

gdzie: Q jest programem równoległym; φ oraz I są formułami (1-go rzędu) z których pierwsza zwana jest warunkiem początkowym Q . Formalna interpretacja $(\#)$ opiera się o semantykę operacyjną programów równoległych, której tu nie przytoczono. Mówiąc nieformalnie, formuła $(\#)$ jest prawdziwa wtt formuła I jest prawdziwa w każdym stanie (wartościowaniu zmiennych) programu Q osiągalnym podczas jego wykonywania, rozpoczętego w pewnym stanie s_0 w którym prawdziwa jest formuła φ (charakteryzująca zbiór stanów początkowych programu Q). Wtedy mówimy, że I jest niezmiennikiem Q ze względu na φ .

Najbardziej użytecznymi dla opisu własności protokołów są formuły niezmienniki w których występują, jako zmienne wolne, wyłącznie odpowiednie zmienne historii wyjściowych i wejściowych. Formuły te, zwane formułami zewnętrznymi odpowiednich programów, definiują własności zewnętrzne lub własności usług. Przy ich budowaniu wykorzystywane są klasyczne spójniki logiczne i kwantyfikatory oraz między innymi następujące operatory:

- $h \circ h'$ - konkatenacja historii h i h' ,
- $m \in h$ - "komunikat m jest elementem historii h ",
- $h \leq h'$ - " h jest początkową podhistorią historii h' " itp.

Przykład 2. Formuły zewnętrzne programów SYSTEM_{k-1} oraz SYSTEM_k z przykładu 1 (por. rys. 1) mają odpowiednio postać:

$$\text{Ext}_{k-1} = (\forall m)(m \in b.\text{out} \rightarrow m \in a.\text{in}) \wedge (\forall m')(m' \in d.\text{out} \rightarrow m' \in c.\text{in}) ,$$

$$\text{Ext}_k = F.\text{out} \leq E.\text{in} \wedge H.\text{out} \leq G.\text{in}$$

A oto odpowiednie formuły poprawnościowe.

$$(A) \quad \langle \text{SYSTEM}_{k-1}, \varphi_{k-1} \rangle \text{ inv } \text{Ext}_{k-1}$$

SYSTEM_{k-1} jest zawodny; powoduje zagubienia, zmianę sekwencyjności oraz generowanie duplikatów komunikatów.

$$(B) \quad \langle \text{SYSTEM}_k, \varphi_k \rangle \text{ inv } \text{Ext}_k$$

gdzie: $\varphi_k = \varphi_k^1 \wedge \varphi_{k-1} \wedge \varphi_k^r$; φ_k^z , $z \in \{1, r\}$, jest warunkiem początkowym programu PM_k^z .

SYSTEM_k jest niezawodny; protokół warstwy k-tej zabezpiecza przed błędami wprowadzanymi przez SYSTEM_{k-1} .

Celem weryfikacji protokołu warstwy k-tej może być udowodnienie (B) przy założeniu (A). ■

3.3. Dowodzenie własności protokołów

System dowodowy umożliwia dowodzenie formuł poprawnościowych postaci ($\star$) (rozdz. 3.2). Zawiera między innymi reguły komponowania dowodów, które pozwalają na modularną budowę dowodów na kształt analizowanych opisów. Mówiąc dokładniej, dowód własności niezmienniczych programu równoległego uzyskiwany jest poprzez odpowiednie złożenie dowodów dla programów-składowych tego programu. Dowody dla składowych sekwencyjnych uzyskiwane są przy tym wprost, w oparciu o teksty odpowiednich procesów. Wynikającą z powyższego zaletą systemu dowodowego jest możliwość rozpoczęcia dowodu własności programu równoległego już w momencie, gdy dostępna jest jakakolwiek jego składowa.

Przykład 3. Weźmy program SYSTEM_k z przykładu 1 (rys. 1) oraz formuły poprawnościowe (A) i (B) z przykładu 2. Zakładamy, że mamy dowód formuły (A) (można go uzyskać podobnie jak dowód formuły (B)). Wtedy dowodzenie formuły (B) może przebiegać następująco. Początkowo generowane są dowody dla składowych sekwencyjnych programu PM_k^z , $z \in \{1, r\}$. Ze złożenia tych dowodów uzyskiwany jest dowód formuły:

$$(\star\star) \quad \langle \text{PM}_k^z, \varphi_k^z \rangle \text{ inv } \text{Ext}_k^z ,$$

gdzie Ext_k^z jest pewną formułą zewnętrzną programu PM_k^z .

Ze złożenia dowodów formuł (A) oraz (B) uzyskiwany jest dowód formuły (B).

4. KONKLUZJE

Przedstawiono aksjomatyczne podejście do formalnego opisu i weryfikacji protokołów oraz scharakteryzowano technikę szczegółową zgodną z tym podejściem [1]. Zaletą techniki jest jej modularność, co pozwala między innymi na ręczną weryfikację złożonych protokołów. Cecha ta może okazać się niezwykle cenną w przypadku budowania systemu wspomagającego projektowanie i weryfikację protokołów, opartego o tą technikę. Z drugiej strony, język opisu protokołów może być rozpatrywany jako możliwy język ich implementacji. Rozwój technologii mikroprocesorowej i praktyka silikonowania rozproszonych programów równoległych [7], których klasą są protokoły komunikacyjne, wymagają do opisu oraz implementacji tych ostatnich języków zorientowanych na komunikację. Takim jest język opisu protokołów. W trakcie prac implementacyjnych można oprzeć się na rezultatach uzyskanych z implementowania wersji CSP, a mianowicie języka OCCAM [5].

BIBLIOGRAFIA

- [1] Barczyński J.: Formalny opis oraz weryfikacja protokołów komunikacyjnych sieci komputerowej. IPI PAN, Warszawa, 1986.
- [2] ESTELLE - A Formal Description Technique Based on an Extended State Transition Model. ISO/TC 97/SC 21, 1985.
- [3] Hoare C.A.R.: Communicating Sequential Processes. CACM, Vol. 21, No. 8, 1978.
- [4] LOTOS - A Formal Description Technique. ISO/TC 97/SC 21, 1985.
- [5] May D.: OCCAM. SIGPLAN Notices, Vol. 13, No. 4, 1983.
- [6] Open Systems Interconnection - Basic Reference Model. ISO, IS 7498, 1983.
- [7] Martin A.J.: Compiling communicating Processes into delay-insensitive VLSI circuits. Distributed Computing, Vol. 1, No. 4, 1986.

DESCRIPTION AND VERIFICATION OF PROTOCOLS - AN AXIOMATIC APPROACH

An axiomatic approach to a formal description and verification of computer network protocols is presented. Detailed technique developed by the author and consistent with the above approach is briefly discussed. The protocol description language is a generalization of CSP notation, while the verification consists in proving invariance properties of protocols.

ОПИСАНИЕ И ПРОВЕРКА КОРРЕКТНОСТИ ПРОТОКОЛОВ -- АКСИОМАТИЧЕСКИЙ ПОДХОД

Представлен аксиоматический подход к формальному описанию и проверке корректности протоколов вычислительной сети. Охарактеризована разработана автором подробная техника согласно с подходом.

Язык описания протоколов является обобщением нотации CSP. Проверка корректности заключается в доказании свойства инвариантных протоколов.

Lesław BUDZIANOWSKI^{*}
Andrzej FRYŚ^{*}

sieć komputerowa,
węzeł komutacji
pakietów, algorytm
wyboru drogi

WŁASNOŚCI ALGORYTMU WYBORU DROGI W SIECI SKJS/2

W referacie przedstawiono algorytm wyboru drogi zastosowany w sieci SKJS/2. We wstępie scharakteryzowano sieć SKJS/2 i własności, jakie powinien posiadać algorytm wyboru drogi. Następnie zostały omówione główne elementy algorytmu i jego własności. W końcowej części podano krótką ocenę nieformalną przyjętego rozwiązania.

1. WSTĘP

W ostatnim okresie prowadzone są intensywne prace nad sieciami komputerowymi. Jednym z takich przedsięwzięć jest budowa sieci komputerowej SKJS/2 [1]. Jest to sieć homogeniczna składająca się z m.c. serii RIAD oraz procesorów teleprzetwarzania PTD.

Cechą charakterystyczną tego rozwiązania jest to, że funkcje węzłów komutacji pakietów SKJS/2 spełniają procesory PTD przy równoczesnej realizacji pełnych funkcji standardowego programu sterującego NCP. Po szczególne PTD łączone są ze sobą zgodnie z rekomendacją X.75 [4]. Równocześnie możliwe jest też dołączenie systemów zewnętrznych do węzłów komutacji pakietów zgodnie z rekomendacją X.25 [3].

Jednym z ważniejszych elementów sieci mającym duży wpływ na takie parametry sieci jak opóźnienie przesyłania danych, przepustowość, dostępność do usług sieci czy stabilność usług jest algorytm wyboru drogi. Algorytm ten jeszcze bardziej zyskał na znaczeniu, kiedy została zbudowana sieć ARPANET i zostały odkryte rzeczywiste problemy i pułapki w sieci. Dziedziną intensywnych badań teoretycznych jest analiza optymalności, czy przynajmniej poprawności, algorytmu wyboru drogi, ale teoretyczne rezultaty nie zawsze dają się zaimplementować przy rozsądnych kosztach.

^{*} Instytut Cybernetyki Technicznej Politechniki Wrocławskiej, Wrocław

Zalecane w literaturze algorytmy wyboru drogi w sieciach [2] są rozproszone we wszystkich węzłach, nie ma centralnego systemu wyboru drogi. W tablicach marszrutyzacji nie jest pamiętana cała droga do węzła docelowego, lecz tylko do najbliższego sąsiada. Każdy węzeł wylicza najlepszą w danej chwili drogę. Algorytmy te są adaptacyjne, monitorują główne zmiany w sieci i szybko przekładają dane pomiarowe na potrzebną informację do wyboru drogi. Są one wolne od tworzenia pętli i oscylacji, a w razie ich powstania wykrywają je i likwidują. Ponieważ w sieci istnieje realna potrzeba szybkiego przesyłania krótkich transakcji pomiędzy użytkownikami bez konieczności wstępnego ustanawiania połączenia, algorytmy wyboru drogi dopuszczają taką możliwość. Dopuszczają one możliwość interwencji operatora w wybór drogi przez udostępnienie dla niego niektórych mechanizmów dostrajania (tuning).

2. WYBÓR DROGI W SIECI SKJS/2

2.1. Tablica marszrutyzacji

Podstawą działania algorytmu wyboru drogi w sieci SKJS/2 jest odpowiednio skonstruowana tablica marszrutyzacji. Tablica marszrutyzacji jest strukturą danych umieszczoną w każdym PTD (węźle) sieci, decydującą o kierunkach przekazywania żądań nawiązania połączeń i datagramów. Dotyczy to wszystkich połączeń, zarówno inicjowanych z lokalnego komputera obliczeniowego jak i tranzytowych (rozpoczynających się w innych węzłach sieci). W tablicy marszrutyzacji każdego PTD reprezentowane są adresy i drogi dojścia do każdego komputera obliczeniowego w sieci, łącznie z komputerem lokalnym.

Podstawą do utworzenia tablicy marszrutyzacji każdego z węzłów jest określenie dróg pierwotnych i zastępczych wiodących z danego węzła do wszystkich pozostałych. Droga pierwotna będzie używana do tworzenia połączenia między węzłami w sytuacji, gdy wszystkie elementy sieci są sprawne. Drogi zastępcze będą wybierane do tworzenia połączenia po wykryciu niesprawności drogi pierwotnej. O kolejności wybierania dróg zastępczych do próby tworzenia połączeń decyduje kolejność ich specyfikacji w tablicy marszrutyzacji. Określając w tablicy marszrutyzacji drogę między węzłami A i B specyfikujemy w każdym z nich tylko jej początek, tj. numer kierunku przez który ma być wysłane żądanie ustanowienia połączenia, tak aby dotarło ono do celu używając drogi pierwotnej lub zastępczej. Między dowolnymi węzłami A i B musi być określona co najmniej droga pierwotna i ewentualnie drogi zastępcze w miarę potrzeb i możliwości ich rozróżnienia. Tak utworzona tablica marszrutyzacji jest listą jednokierunkową składającą się z elementów określających węzły docelowe w sieci. W każdym z tych elementów znajdują się następujące pola:

- wskaźnik (adres) następnego elementu w liście,
- sieciowy adres docelowego PTD,
- ilość możliwych kierunków prowadzących do wskazanego adresu docelowego,
- pierwszy numer kierunku,
- drugi numer kierunku,
- .
- .
- .
- n-ty numer kierunku.

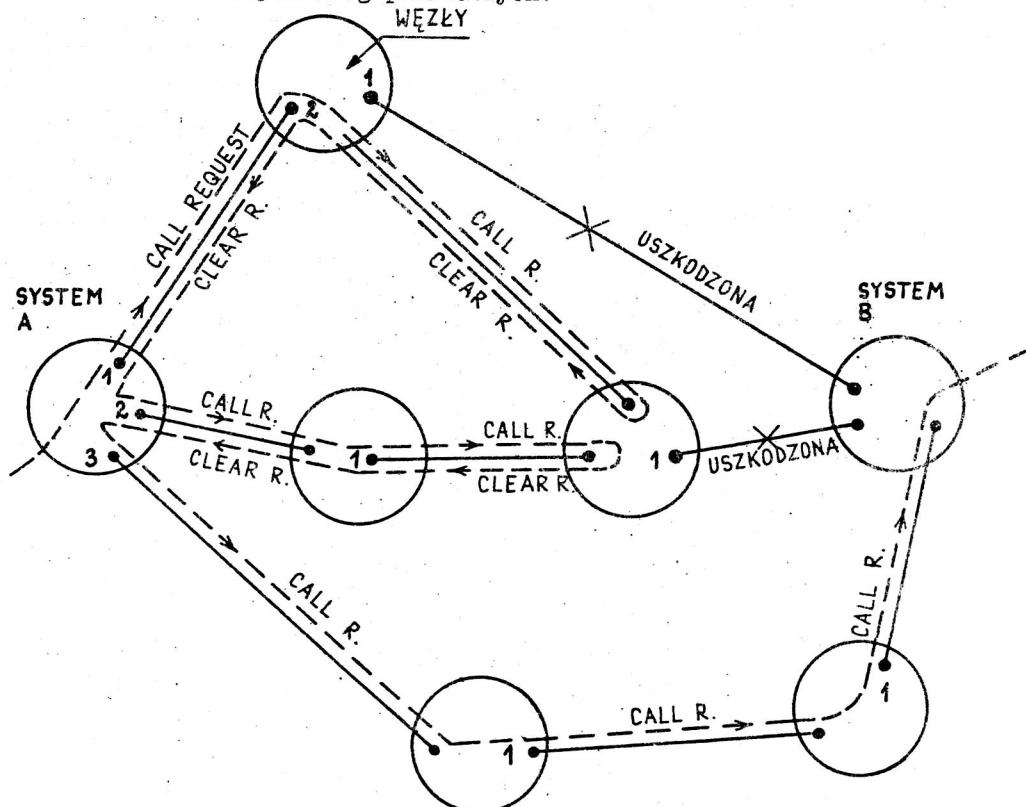
2.2. Zasady wyboru drogi dla połączeń sieciowych

Żądanie ustanowienia połączenia sieciowego niezależnie od pochodzenia (może ono pochodzić od stacji transportowej gdy jest inicjowane lub od stacji sieciowej gdy przychodzi z innego węzła) powoduje wybranie z tablicy marszrutyzacji elementu określającego żądany adres docelowy. Brak takiego elementu powoduje odrzucenie połączenia ze wskazaniem przyczyny - błąd adresacji. Natomiast, gdy istnieje możliwość dojścia do adresu docelowego, wybierana jest odpowiednia droga pierwotna lub zastępcza (numer kierunku), jeżeli tylko dany kierunek jest aktywny i ma możliwość tworzenia nowych połączeń sieciowych (nie jest przekroczony zadany limit połączeń). W przypadku braku odpowiedniego numeru kierunku następuje odrzucenie połączenia ze wskazaniem jako przyczyny - brak dojścia. Jeśli dany węzeł okaże się węzłem docelowym (zgodność adresów w pakiecie CALL REQUEST z adresem danego węzła) następuje potwierdzenie połączenia sieciowego pakietem CALL ACCEPTED, jeżeli tylko stacja transportowa jest aktywna i zdolna do tworzenia nowych połączeń transportowych.

Otrzymanie żądania rozłączenia CLEAR INDICATION w fazie nawiązywania połączenia sieciowego z przyczynami innymi niż powstanie pętli lub brak dojścia jest przekazywane do żądającego ustanowienia połączenia sieciowego. Natomiast w przypadku pętli lub braku dojścia następuje próba znalezienia innej drogi zastępczej przez sprawdzenie pozostałych numerów kierunków, jeszcze nie sprawdzanych, jeżeli tylko istnieją.

W celu zabezpieczenia się przed powstaniem pętli, każde połączenie sieciowe ma nadany numer referencyjny unikalny w całej sieci. W przypadku wystąpienia, wśród istniejących połączeń sieciowych w danym węźle, połączenia z numerem referencyjnym identycznym jak w nowo zgłaszanym połączeniu sieciowym pakietem CALL REQUEST, mamy do czynienia z pętlą. Nowe połączenie jest wtedy odrzucane ze wskazaniem jako przyczyny powstanie pętli. Należy zauważyć, że informacje o poszczególnych połączeniach sieciowych (kanałach logicznych) istniejących w danym węźle są przechowywane w formie skojarzonych tablic. Przykładowa próba zestawie-

nia połączenia przedstawia rysunek 1. Pokazano na nim drogę jaką musi przebyć pakiet CALL REQUEST od systemu A do B w przypadku gdy istnieją uszkodzenia niektórych dróg pierwotnych.



Rys. 1. Przykładowe zestawienie połączenia sieciowego
Fig. 1. Example of network connection establishment

2.3. Zasady wyboru drogi dla przesyłania datagramów

Reguły wyboru drogi przy przesyłaniu datagramów bazują, podobnie jak w przypadku zestawiania połączeń sieciowych, na tablicy marszrutyzacji. Cechą charakterystyczną różniącą przesyłanie datagramów do zestawiania połączenia sieciowego jest brak śladu w danym węźle po wysłaniu datagramu.

W chwili pojawienia się datagramu, w celu jego wysłania, sprawdzają się tablicę marszrutyzacji, aby znaleźć element o adresie docelowym zgodnym z podanym w datagramie. W przypadku znalezienia takiego elementu następuje wybranie numeru kierunku, w którym będzie można wysłać datagram. W tym celu wybrany kierunek (stacja sieciowa) musi być aktywny, a ilość datagramów oczekujących w kolejce na wysłanie nie może przekraczać zadanego limitu. W takiej sytuacji datagram jest dołączony

do kolejki z zaznaczeniem w nim, że jest on wysyłany "do przodu".

W przypadku, gdy w tablicy marszrutyzacji nie ma takiego elementu bądź nie ma dostępnego wyjścia, datagram jest odsyłany w tym samym kierunku z którego przyszedł z zaznaczeniem w nim, że jest wysyłany "wstecz". Datagram dotarł do celu, jeśli stwierdzono zgodność adresu docelowego w datagramie z adresem danego węzła. Dla zaznaczenia kierunku ruchu datagramu służy odpowiedni wskaźnik przesyłany wraz z datagramem.

W przypadku otrzymania datagramu ze wskaźnikiem informującym o ruchu datagramu "wstecz", następuje wybranie następnej nie sprawdzanej drogi zastępczej. Gdy taka droga istnieje, datagram jest wysyłany nią z zaznaczeniem kierunku ruchu "do przodu". Natomiast gdy nie istnieje inna droga zastępcza w tym węźle doprowadzająca datagram do celu, dany datagram jest wysyłany "wstecz" do kolejnego węzła. Osiągnięcie węzła źródłowego przy ruchu datagramu "wstecz", powoduje skasowanie datagramu z powodu niemożności dotarcia do adresu docelowego. Datagram jest kasowany także, gdy wskaźnik ilości prób wyboru drogi "do przodu", zawierający liczbę przejść przez węzły sieci, przekroczy zadany limit. Zapobiega to przed krążeniem datagramów w sieci i powstawaniem pętli.

3. UWAGI KOŃCOWE

Jak można się przekonać, algorytm zastosowany w sieci SKJS/2 spełnia w znacznym stopniu postulaty stawiane w literaturze. Przede wszystkim ma charakter rozproszony, decyzje o wyborze drogi podejmowane są w poszczególnych węzłach na tworzonej drodze. W tablicach marszrutyzacji nie jest przechowywana droga z końca do końca, a jedynie kierunek do najbliższego sąsiada, który może pośredniczyć w osiągnięciu punktu docelowego. Algorytm zapewnia wykrycie i likwidację powstałej pętli i jest wolny od oscylacji w sieci. Jest przystosowany do ustanawiania logicznych połączeń sieciowych jak i do przesyłania informacji bez tworzenia połączeń (zgodnie z X.25 i X.75). Zalecany w literaturze warunek adaptacyjności algorytmu wyboru drogi nie jest spełniony. Cecha ta wiąże się z dodatkowymi kosztami pamięci, gdyż w tym przypadku nieodzownym jest system zbierania, analizy i rozsyłania informacji o zmianach w sieci. Zastosowany jako węzeł procesor PTD charakteryzuje się brakiem własnej pamięci zewnętrznej o bezpośrednim dostępie jak i ograniczoną pamięcią operacyjną. Pewne elementy adaptacyjności zostały uwzględnione przez wprowadzenie mechanizmu aktywacji poszczególnych kierunków.

W przypadku braku aktywności danego kierunku nie jest on brany pod uwagę w rozważaniach przy wyborze drogi. Istnieje też możliwość

interwencji operatorskiej przez włączenie lub wyłączenie pewnych kierunków w celu dostrojenia optymalnych parametrów sieci. Sam algorytm może pracować przy adaptacyjnym modelu wyboru drogi. Wiąże się to tylko z dynamiczną zmianą kolejności numerów kierunków w tablicy marszrutyzacji. Pewną niedogodnością tego rozwiązania związaną z ograniczonymi zasobami pamięci jest konieczność zmiany tablicy marszrutyzacji we wszystkich węzłach w przypadku dołączenia nowego węzła.

Istnieją duże trudności z wykazaniem, że w dowolnym rzeczywistym przypadku dany algorytm jest najlepszy, niektóre algorytmy górują nad innymi w przypadku ustabilizowanego natężenia ruchu, ale bardzo słabo reagują na sytuacje wyjątkowe. Wybór algorytmu powinien uwzględniać wielkość sieci, niezawodność sprzętu i posiadane do dyspozycji zasoby. Wydaje się, że zaproponowany algorytm wyboru drogi spełnia żądane wymagania w sieci SKJS/2 co też zostało sprawdzone eksperymentalnie.

LITERATURA

- [1] Budzianowski L., Markowska U., Wietrznych J.: Projekt techniczny oprogramowania PTD jako procesora czołowego dla EC 1032 i komputera węzła w sieci SKJS/2. ICT PWr, Raport SFR nr 3/85.
- [2] Ossola M.: Routing principles for the eighties: are experience and theory sufficient to settle for distributed and adaptative routing? Performance of Data Communication System and their Applications, North-Holland Publishing Company, 1981.
- [3] Yellow Book, Data communication networks services and facilities, terminal equipment and interfaces. Recommendation X.25, Volume VIII - FASCICLE 2, CCITT, Geneva 1981.
- [4] Yellow Book, Data communication networks transmission, signalling and switching, network aspects, maintenance, administrative arrangements, Recommendation X.75, Volume VIII - FASCICLE 3, CCITT, Geneva 1981.

PROPERTIES OF ROUTING ALGORITHM IN THE COMPUTER NETWORK SKJS/2

In this paper routing algorithm adepted in SKJS/2 has been presented. At the beginning the computer network SKJS/2 and its properties which should own this routing algorithm have been characterized. Then the main elements of this algorithm and its properties have been discussed. In the final part of the paper a short informal appreciation of the accepted solution has been presented.

СВОЙСТВА АЛГОРИТМА ПРОКЛАДКИ МАРШРУТА В СЕТИ SKJS/2

В статье представлен применённый в сети SKJS/2 алгоритм прокладки маршрута. В предисловии дана характеристика сети SKJS/2 и свойст какие должен иметь алгоритм прокладки маршрута. Далее представлены главные элементы алгоритма и его свойства. Дан также краткая неформальная оценка принятых решений.

Piotr COFTA^X

Sylwester KACZMAREK^X

sieci lokalne

systemy wielokomputerowe

centrale telefoniczne

ZASTOSOWANIE SIECI LOKALNYCH W DEDYKOWANYCH SYSTEMACH WIELOMIKROKOMPUTEROWYCH

W pracy przedstawiono rzeczywiste zastosowanie koncepcji sieci lokalnych dla potrzeb dedykowanych systemów wielomikrokomputerowych, ze szczególnym uwzględnieniem central telefonicznych. W systemach tego typu sprawą mniej istotną jest zapewnienie wszelkich własności pracy w sieci, a istotniejsza staje się możliwość dostarczenia prostego sterownika sieci realizującego wybrane funkcje. Opis budowy oraz funkcji takiego sterownika został umieszczony w pracy.

1. WSTĘP

Stosowanie sieci lokalnych, popularne w wypadku mikrokomputerów ogólnego przeznaczenia, napotyka na pewne trudności w wypadku dedykowanych systemów wielomikrokomputerowych. Jedną z nich jest stosunkowo znaczny koszt sterownika sieci lokalnej szczególnie w sytuacji, gdy wykorzystywany jest jedynie nieznaczny zakres jego możliwości.

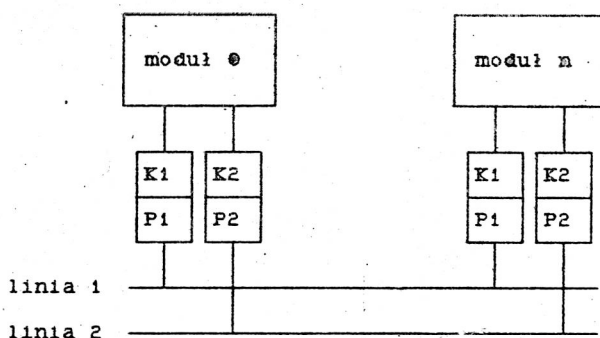
Jest to istotne szczególnie dla urządzeń typu centrale telefoniczne, w których istnieje znaczna liczba stosunkowo prostych mikrokomputerów, a jednocześnie zachodzi konieczność ich ścisłej współpracy. W tej sytuacji pomocnym może być prosty sterownik sieci LAN o możliwościach dopasowanych do rzeczywistych potrzeb danego urządzenia.

Niniejszy referat zawiera zarys koncepcji komunikacji pomiędzy modułami centrali SPC-800 [1,2] w oparciu o strukturę tzw. pseudo - LAN (pLAN), z wykorzystaniem zmodyfikowanego algorytmu CSMA/CD. Konieczność skonstruowania takiego specjalizowanego sterownika wynika wprost z potrzeby zapewnienia pewnego a jednocześnie elastycznego sposobu komunikacji międzymodułowej. Sama logiczna konstrukcja pLAN nawiązuje do stosowanych w modułach mechanizmów komunikacji międzyprocesorowej i międzyprocesowej [3].

^X Politechnika Gdańska, Gdańsk

2. BUDOWA SIECI

Schematyczna budowa centrali została pokazana na Rys. 1. Centrala składa się z szeregu modułów. Każdy z nich jest podłączony do dwóch niezależnych linii przez które może się komunikować z pozostałymi modułami. Podłączenie do linii jest zrealizowane poprzez wyspecjalizowany sterownik pLAN, złożony z lokalnego mikrokomputera (K_i) realizującego właściwy algorytm pLAN oraz z układu pośredniczącego - dopasowującego (P_i).



Rys. 1. Ogólna struktura połączeń modułów.

Fig. 1. General structure of the modules interconnection.

Podstawową rolę w układzie pośredniczącym spełnia zegar transmisji. Generuje on symetryczny przebieg prostokątny o częstotliwości równej pożądanej szybkości przekazywania informacji po linii. Zegar ten posiada ponadto wejście zerujące. Sygnały, które lokalny mikrokomputer przekazuje na linię są synchronizowane z zegarem i wysyłane na linię w chwili narastającego zbocza zegara.

Sygnały przychodzące z linii są próbkowane i zapamiętywane w chwili odpowiadającej środkowi taktu zegara, t. j. przy opadającym jego zboczu. Zapewnia to optymalną poprawność próbkowania przy założeniu synchronicznej pracy zegarów nadawcy i odbiorcy. W celu odpowiedniego zsynchronizowania zegara odbiorcy, pojawienie się na linii jakiegokolwiek zbocza powoduje skasowanie stanu zegara transmisji. Ponieważ w trakcie własnego nadawania taka synchronizacja nie jest potrzebna, możliwe jest zablokowanie tego mechanizmu. Narastające zbocze sygnału zegara powoduje ponadto zsynchronizowanie pracy układu pośredniczącego i lokalnego mikrokomputera.

Tak skonstruowany układ pośredniczący gwarantuje poprawność pracy sterownika przy stosunkowo niskiej częstotliwości zegara ($1 \cdot$ szybkość

pracy linii) oraz przy znacznej prostocie układowej. Tym samym realizacja większości funkcji została przeniesiona na oprogramowanie lokalnego mikrokomputera.

3. FORMAT TRANSMISJI

Jako podstawę przyjęto transmisję szeregową, asynchroniczną. Szybkość transmisji jest wyznaczana poprzez zegar transmisji. Zależy ona od szybkości pracy procesora lokalnego mikrokomputera i jest od niej od 100 do 120 razy mniejsza. Szacunkowo, można przyjąć, iż szybkość transmisji wynosi:

dla procesora i8080 z zegarem 2MHz - 19200 bit/s

dla procesora Z80 z zegarem 2MHz - 19200 bit/s

dla procesora Z80 z zegarem 4MHz - 38400 bit/s

Transmisja odbywa się zawsze pełnymi komunikatami o zmiennej długości. Pomiędzy poszczególnymi bajtami komunikatu nie występują przerwy. Każdy komunikat, oprócz samej przesyłanej treści zawiera ponadto ramkę, która przenosi informacje o jego długości, adresach (identyfikatorach) nadawcy i odbiorcy oraz o numerze kolejnej transmisji pomiędzy daną parą nadawca - odbiorca. Ponadto ramka zawiera sumę kontrolną całego komunikatu.

Odebranie każdego komunikatu jest potwierdzane przez odbiorcę poprzez wysłanie komunikatu - potwierdzenia. Budowa komunikatu - potwierdzenia jest analogiczna do budowy typowego komunikatu.

Ponadto linia transmisyjna może być, przez sterownik któregośkolwiek z modułów wprowadzona w stan tzw. zagłuszania. Pojęcie to oznacza podanie na linię przez odpowiedni, ustalony odcinek czasu stanu "0".

4. KOMUNIKACJA Z MODULEM

Opisany powyżej sterownik sieci lokalnej jest przeznaczony do wykorzystywania jako inteligentne urządzenie zewnętrzne, sterowane przez procesor nadrzędny. Sterownik jest połączony z procesorem nadrzędnym poprzez dwukierunkowe ośmiobitowe łącze równoległe. Komunikacja polega na podawaniu odpowiednio interpretowanych kodów, zależnie od potrzeb. Kody (instrukcje) zarówno przesyłane do jak i z procesora / sterownika posiadają zbliżony, a często wręcz identyczny format.

5. ZAKOŃCZENIE

Niniejsza propozycja układu sprzęgającego procesor do szyny z możliwością pracy jak w sieci lokalnej wydaje się być bardzo prosta, szczególnie w swojej części sprzętowej. Oprogramowanie, z racji bardzo ścisłych wymagań czasowych jest oczywiście o wiele bardziej skomplikowane w wykonaniu. Całość oprogramowania w aktualnej wersji zajęła 2KB pamięci ROM, a wymagany obszar RAM waha się od 256B do 1KB zależnie od pożądanej maksymalnej długości komunikatu.

Proponowane rozwiązanie może znaleźć zastosowanie w szeregu niedużych systemach wielokomputerowych, w których zastosowanie pełnego sterownika sieci lokalnej nie jest uzasadnione.

Dotychczasowe eksperymenty przeprowadzane z użyciem sterownika wykazały jego znaczną niezawodność oraz odporność na sytuacje nietypowe (zanik napięcia, brak modułu itp.).

LITERATURA

- [1] S. Kaczmarek, P. Cofta, A. Stankiewicz, R. Weisbrodt: Mikroprocesorowy układ realizacji funkcji eksploatacji technicznej centrali telefonicznej SPC-800, Materiały II Krajowej Konferencji Naukowo - Technicznej "Przetwarzanie Sygnałów w Telekomunikacji, Sterowaniu i Kontroli", Bydgoszcz 1986, str. 402-410.
- [2] S. Kaczmarek, P. Cofta, A. Stankiewicz, R. Weisbrodt: Funkcje eksploatacji technicznej i ich realizacja w centralach średniej pojemności i sterowaniu programowym, Materiały Krajowego Sympozjum Telekomunikacji, Rynia 1986, str. 604-613.
- [3] S. Kaczmarek, P. Cofta, A. Stankiewicz, R. Weisbrodt: System diagnostyczny - utrzymaniowy centrali SPC-800. Projekt koncepcyjny i techniczny wybranej wersji. Raport 821070/84, Politechnika Gdańska, Gdańsk 1984.

THE USE OF THE LOCAL AREA NETWORK FOR THE DEDICATED MULTIMICROCOMPUTER SYSTEMS

In this paper the real implementation of the local area network concept for the dedicated multimicrocomputer systems is presented. Mainly the implementation for the electronic switching network is analysed. In such systems it is less important to provide all of the LAN facilities but it is more important to be able to use the simple, unexpensive network controller. In this paper the description of construction and functions provided by such controller is also given.

ПРИМЕНЕНИЕ ЛОКАЛЬНЫХ СЕТИ ДЛЯ СПЕЦИАЛИЗИРОВАННЫХ МНОГОМИКРОКОМПЬЮТЕРНЫХ СИСТЕМОВ

Описано применение концепции локальной компьютерной сети для специализированных многомикромпьютерных систем. В таких системах возможность использования дешевых управляющих устройств сети является более существенным чем обеспечение всех возможностей работы в сети. Описано тоже схему и правила действия такого устройства, построенного для телефонной станции.

Wiktor Bohdan DASZCZUK^Xsieć mikrokomputerów
sieć pierścieniowa
protokół
realizacja sieciPIERŚCIENIOWA SIEĆ MIKROKOMPUTERÓW
(zrealizowana w MERA-SYSTEM)

W referacji przedstawiono pierścieniową sieć komputerową zrealizowaną w Przedsiębiorstwie Systemów Komputerowych MERA-SYSTEM, na tle innych rozwiązań. W sieci starano się pogodzić prostotę węzłów sieci z prostotą protokołu wymiany informacji. Przyjęto zasadę unifikacji węzłów.

1. Wprowadzenie

Najczęściej spotykanymi rozwiązaniami w lokalnych sieciach mikrokomputerów są: wspólna magistrala [3] oraz pierścień [1,2,4]. Sieć pierścieniowa jest wolniejsza i bardziej zawodna od organizacji magistralowej, lecz cieszy się dużym zainteresowaniem ze względu na niski koszt instalacji oraz niewielki koszt dołączania kolejnych mikrokomputerów. Wady i zalety wymienionych konfiguracji są dobrze znane [5] i nie zostaną tu omówione.

W niniejszym referacie jest opisana sieć pierścieniowa zrealizowana w MERA-SYSTEM. Implementacja sieci jest kompromisem pomiędzy prostotą protokołu i prostotą węzłów sieci. Wprowadzono zasadę całkowitej unifikacji węzłów.

2. Protokół komunikacyjny

Protokół spełnia w sieci pierścieniowej trzy podstawowe funkcje: nawiązywanie łączności między węzłami, przekazywanie informacji, oraz zwiększenie niezawodności.

Ze względu na duży czas przesyłania informacji (każdy węzeł wprowadza pewne minimalne opóźnienie), dwie pierwsze funkcje są często łączone. Tak jest i w przypadku sieci MERA-SYSTEM. Połączenie obu funkcji polega na zaopatrzeniu informacji użytecznej w nagłówek. Nagłówek ten spełnia rolę "wywoływania" odbiorcy, przy czym informacja jest wysyłana "w ciemno", nie wiedząc, czy dotrze ona do adresata i czy zostanie zaakceptowana.

2.1. Nadawanie informacji

Podstawowym problemem w sieciach pierścieniowych jest potencjalna kolizja informacji przechodzącej przez węzeł (transmitowanej) z informacją nadawaną przez ten węzeł. Znane są trzy podstawowe metody rozwiązania tej niedogodności:

^X

Przedsiębiorstwo Systemów Komputerowych MERA - SYSTEM, WARSZAWA

1. "Sztafeta", czyli przekazywanie sobie przez węzły pewnego elementu jako pałeczki sztafetowej. Pałeczka jest w systemie jedna (lub jest ich stała liczba). Węzeł posiadający pałeczkę może nadawać, a pozostałe mogą transmitować, odbierać i ewentualnie konsumować informacje. Wadami tego rozwiązania są: trudność rekonfiguracji sieci (grozi zanikiem lub powieleniem pałeczki) i małe wykorzystanie kanału przesyłania.

2. "Karuzela", polegająca na permanentnie krążących w pierścieniu "krzeselkach". Jeżeli przez węzeł przechodzi puste krzeselko, może zostać wypełnione informacją. Pewnym problemem tej metody jest konieczność utrzymania odpowiedniej do liczby węzłów ilości krzesłek. Służące temu, algorytmy wymagają przesyłania informacji technicznych. Czasem stosuje się wyróżniony węzeł, który może dodawać lub usuwać krzeselka z karuzeli [2].

3. Buforowanie. Metoda ta polega na wstrzymywaniu informacji przechodzącej przez węzeł w buforze, jeżeli węzeł nadaje. Informacja transmitowana ma pierwszeństwo przed informacją nadawaną, tak jak w ruchu na rondzie (w poprzednim kodeksie drogowym). Węzeł może rozpocząć nadawanie tylko wtedy, gdy nastąpi przerwa w informacji transmitowanej. Pojemność bufora musi zapewnić wysłanie przez węzeł całej przygotowanej informacji. Protokół nie wymaga przesyłania technicznych informacji w sieci. W sieci MERA-SYSTEM zastosowano buforowanie.

Powyższe rozważania ukazują, że prostota protokołu stoi w sprzeczności z prostotą węzła: uproszczenie protokołu kosztuje buforowanie informacji w węźle.

2.2.Usuwanie informacji

W sieciach pierścieniowych istnieje konieczność usuwania krążącej informacji po jej odebraniu. Stosowane są następujące metody:

1. Usuwanie przez wyróżniony w systemie węzeł [2]. Ta metoda nie spełnia założenia o unifikacji węzłów.

2. Usuwanie przez odbiorcę. Odbiorca jest zobowiązany do wysłania odpowiedzi. Jeżeli adresat nie odbierze informacji, usunąć ją musi nadawca.

3. Usuwanie przez węzeł nadawczy (nadajnik). Odbiorca potwierdza odebranie informacji poprzez zaznaczenie pewnego miejsca w strumieniu informacji. Ta metoda nie wymaga przesyłania informacji technicznej (odpowiedzi) i wyznacza zawsze jednoznacznie węzeł, który ma usunąć informację z sieci. Metodę (3) zastosowano w sieci MERA-SYSTEM.

2.3.Usuwanie informacji przekłamanaj

Istotnym problemem w sieciach pierścieniowych jest usuwanie informacji przekłamanaj. Przekłamanie w części użytecznej informacji zostanie omówione w p. 2.5. Tu zajmiemy się takim przekłamanaj, które w nagłówku zmienia numer nadawcy. Jeżeli błąd ten zostanie wykryty przy użyciu danych redundantnych, to informacja może zostać usunięta przez węzeł, który wykryje błąd.

Poważniejszą jest sytuacja, w której błąd nie może być wykryty przez wykorzystanie danych redundantnych, na przykład przekłamanie dwóch bitów nagłówka przy kontroli typu bit parzystości [1]. Informacja z tak przekłamanym nagłówkiem może potencjalnie krążyć w pierścieniu w nieskończoność, jeżeli numer nadawcy powstały wskutek przekłamanaj nie odpowiada żadnemu z numerów węzłów aktualnie włączonych w sieć. Im bardziej wyrafinowana jest kontrola nagłówka, tym mniejsze jest prawdopodobieństwo takiego przekłamanaj nagłówka, że będzie się on prezentował jako poprawny. Przy zastosowaniu kontroli cyklicznej CRC, prawdopodobieństwo to może być pominięte. Koszt przesyłania informacji w sieci wzrasta jednak, dane redundantne CRC zajmują 16 bitów. Tutaj

znow napotykamy sprzeczność pomiędzy prostotą protokołu a prostotą węzła. W sieci MERA-SYSTEM zastosowano kontrolę CRC upraszczając algorytm działania węzła: nie jest implementowany algorytm usuwania "niczyjej", choć zewnętrznie poprawnej informacji.

Istnieje niebezpieczeństwo, że węzeł zostanie odłączony od sieci w momencie, gdy w pierścieniu znajduje się wysłana przez niego informacja. Świadomie dopuszczono możliwość krążenia w sieci tak powstałej "niczyjej" informacji.

2.4. Odbieranie informacji

Odbiornik może być w trzech stanach ze względu na konwersację z nadajnikiem:

- oczekujący, jeżeli akceptuje informacje z nadajnika,
- zajęty, jeżeli nie może przyjąć informacji z nadajnika (na przykład z powodu zajętości bufora),
- zamknięty, jeżeli nie akceptuje informacji z nadajnika.

Każdy ze stanów jest sygnalizowany nadajnikowi odpowiedzią ustawianą w nagłówku. Przy odpowiedziach sygnalizujących stan oczekujący lub zamknięty, odbiornik kończy współpracę odpowiednio z sukcesem lub porażką. Sygnalizacja stanu zajętości jest równoważna z żądaniem oczekania przez nadajnik określonego technicznego odstępu czasu i ponowienia nadawania.

2.5. Zwiększanie niezawodności

W celu zwiększenia niezawodności sieci w stosunku do pewności działania fizycznych łączy, stosuje się retransmisję (powtórne nadawanie) informacji w przypadku jej przekłamania. Błąd może być wykrywany przez odbiornik lub nadajnik informacji.

1. Wykrywanie przez odbiornik. Rozwiązanie to polega na ustawieniu przez odbiornik bitu poprawności. Dane redundantne dotyczące bitu poprawności muszą być umieszczone przez odbiornik (nawet dla przekłamaną informację), aby nadajnik mógł właściwie zinterpretować bit poprawności. Jeżeli nie wystąpi przekłamanie za odbiornikiem, nadajnik ma możliwość określenia, czy informacja była odebrana poprawnie. Nie ma tej możliwości w przypadku przekłamania między odbiornikiem a nadajnikiem (przekłamaniu mógł ulec bit parzystości).

2. Wykrywanie przez nadajnik. W tym rozwiązaniu nadajnik sprawdza poprawność powracającej do niego informacji i w przypadku wykrycia błędu zakłada, że odbiornik otrzymał informację przekłamaną. Odbiornik jest zobowiązany kontrolować informacje "na własny użytek", aby nie przyjąć informacji błędnej. Wykrywanie przez nadajnik jest zastosowane w sieci MERA-SYSTEM.

3. Organizacja węzła

3.1. Przepływ informacji

Organizacja węzła sieci MERA-SYSTEM jest przedstawiona na rys.1.

Zasadniczą częścią węzła są bufor: nadawczy, odbiorczy i transmisyjny.

W buforze nadawczym (BN) informacja jest gromadzona i zaopatrywana w nagłówki i dane redundantne. W buforze transmisyjnym (BT) znajdują się informacje przeczytane z sieci, lecz jeszcze nie wysłane. Przełącznica wyjściowa (A) wybiera jeden z buforów (BN, BT) jako źródło informacji do wysyłania. Przełącznica może zostać przełączona na bufor nadawczy, jeżeli jest informacja do nadania w buforze nadawczym, a bufor transmisyjny jest pusty. Jeżeli bufor transmisyjny przestanie być pusty, przełącznica po wysłaniu informacji bezwzględnie przestawi się na

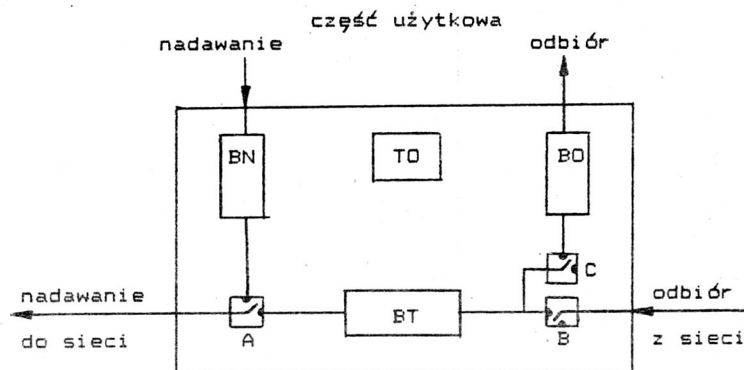
bufor transmisyjny. W ten sposób realizowany jest priorytet informacji będącej w sieci nad informacją, która ma być wprowadzona do sieci. Informacja wysłana z bufora nadawczego jest w nim zamrażana do ewentualnych retransmisji.

W buforze odbiorczym (BO) jest zbierana informacja przeznaczona dla węzła. Przełącznica wejściowa (B) analizuje nagłówek informacji i powoduje przesłanie jej do bufora transmisyjnego, jeżeli nie ma być usunięta z sieci. Informacja jest usuwana z sieci, gdy zostanie wykryty błąd nagłówka oraz gdy do węzła powraca wysłana informacja, (ponowne liczenie sumy kontrolnej). W przypadku wykrycia błędu, informacja w buforze nadawczym jest odmrażana i ponownie wysyłana. Postępowanie to jest powtarzane określoną liczbę (trzy) razy, aż do sukcesu lub stwierdzenia niemożności nawiązania poprawnej łączności.

Przełącznica odbiorcza (C) zwiera bufor odbiorczy do toru informacji, jeżeli (1) węzeł jest odbiornikiem informacji, (2) bufor odbiorczy jest pusty, (3) w tabeli odbiorców (TO) jest zaznaczony stan oczekujący względem nadawcy informacji. Informacja jest wstawiana jednocześnie do bufora transmisyjnego i odbiorczego. Informacja ta jest przekazywana części użytkowej węzła, jeżeli jest poprawna.

W tabeli odbiorców jest zapisywany dla każdego węzła numer retransmisji (zawarty w nagłówku) ostatnio odebranej informacji. Jeżeli numer retransmisji informacji aktualnie odbieranej jest większy od numeru ostatnio poprawnie odebranego, informacja jest ignorowana. Numer w tabeli odbiorców jest ustawiany na numer retransmisji aktualnie odbieranej informacji (niezależnie od jej poprawności). Takie postępowanie zapobiega powtórному odebraniu tej samej informacji, gdy w sieci jest uszkodzenie pomiędzy odbiornikiem a nadajnikiem.

Istnienie jednego bufora nadawczego i wielu pozycji w tabeli odbiorców pozwala węzłowi prowadzić jednocześnie jedną łączność jako nadajnik i wiele łączności jako odbiornik.



Rys.1 Logiczna organizacja węzła sieci MERA-SYSTEM
Fig.1 Logical scheme of MERA-SYSTEM net node

3.2. Budowa informacji

Informacja przesyłana w sieci składa się z nagłówka i informacji użytecznej, co jest typowe dla sieci pierścieniowych (patrz rozdział 2). Nagłówek i informacja użyteczna są wyposażone w oddzielne dane redundantne CRC. Najistotniejszymi elementami nagłówka są: numer nadawcy, numer odbiorcy, numer retransmisji, kod typu informacji i kod odpowiedzi. Kod typu informacji może oznaczać:

- zwyczajną łączność,

- informację typu 'broadcast', czyli do wszystkich (nie zaimplementowane),
- informację "do nikogo", czyli test drożności pierścienia (nie zaimplementowane).

Kod odpowiedzi oznacza:

- odbiornik jest zamknięty,
- odbiornik jest zajęty,
- odpowiedź pozytywna.

Struktura informacji jest bajtowa (wynika to z zastosowanego sprzętu).

4. Implementacja

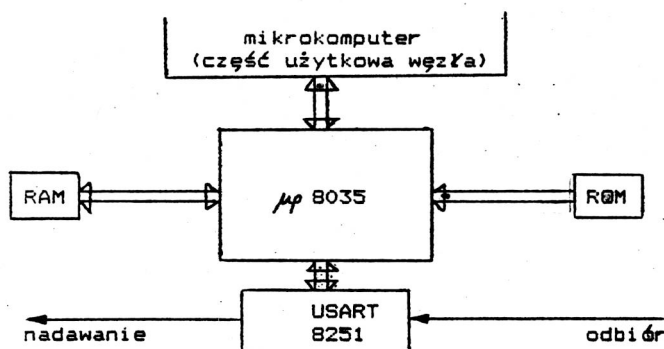
Węzeł sieci MERA-SYSTEM został zrealizowany na bazie układu mikroprocesorowego. Część użytkową implementuje dowolny mikrokomputer połączony z pakietem węzła sprzęgiem równoległym. Wykonano węzeł w postaci pojedynczej eurokarty z łączem krawędziowym 96-stykowym, lecz są rozważane wykonania jako pakietu IBM/PC i jako "czarnej skrzynki" z własnym zasilaczem.

Węzeł jest zbudowany na bazie mikroprocesora 8035 (rozważane jest użycie 8085). Wykorzystano 4k pamięci EPROM oraz 1k zewnętrznej pamięci statycznej RAM. Procesor współpracuje z układem 8251 służącym jako nadajnik/odbiornik sieci.

Programowo algorytm węzła jest implementowany jako zespół jednopoziomowych podprogramów obsługi przerwań.

W sieci zastosowano łącze V-24 z pętlą prądową 20 mA z optoizolacją. Sieć pracowała poprawnie z prędkością transmisji 9600 baud przy odległościach między węzłami rzędu 10 m i z prędkością 150 baud przy odległościach do kilometra.

Awaria zasilania w węźle powoduje elektryczne zwarcie wejścia z wyjściem węzła. Istnieje również możliwość programowego odłączenia węzła od sieci przez zwarcie wejścia z wyjściem.



Rys.2 Schemat budowy węzła

Fig.2 Hardware diagram of net's node

5. Podsumowanie

W sieci pierścieniowej MERA-SYSTEM udało się pogodzić wymagania co do prostoty protokołu i prostoty węzłów sieci. Zaopatrzenie informacji użytkowej w nagłówki pozwala na połączenie funkcji nawiązywania łączności i przekazywania informacji. Zastosowano zasady buforowania informacji przez węzeł nadawczy i usuwania informacji z sieci przez ten

węzeł. Informacja o przekłamanym nagłówku jest usuwana z sieci przez węzeł, który wykryje przekłamanie. Ścisła kontrola nagłówków pozwala na rezygnację z wykrywania permanentnie krążącej "niczyjej" informacji. Zwiększanie niezawodności polega na kontroli cyklicznej CRC i 3-krotnej próbie retransmisji w przypadku wykrycia błędu. Węzły sieci są zunifikowane. Istotną własnością węzła sieci jest fakt, że do poprawnego działania nie jest wymagane poprawne działanie innych węzłów. Ma to szczególne znaczenie przy uruchamianiu i testowaniu węzła oraz przy lokalizacji uszkodzeń w sieci.

Sieć zrealizowano tą technologią przy użyciu powszechnie dostępnych elementów. Temat został zrealizowany przez zespół w składzie : Stanisław Chrobot, Piotr Kowalski, Jan Krawczyk i autor niniejszego referatu.

Literatura

- [1] Farber D. J., Larson K. C.
The System Architecture of the Distributed Computer System - the Communications System
Proc. Symp. on Computer - Communications Networks and Teletraffic, Polytechnic Institute of Brooklyn, April 1972
- [2] Lis R. W.
Lokalna sieć komputerowa o konfiguracji pierścieniowej
Raport IPI PAN Nr 536, Warszawa, styczeń 1984
- [3] Metcalfe R. M., Boggs D. R.
Ethernet : Distributed Packet Switching for Local Computer Networks
Comm. ACM 19,7 (July 1976) str. 395-404
- [4] Wilkes M. V.
The Cambridge Digital Communication Ring
Proc. Mitre/NBS, Local Area Networks Symposium, Boston MA, May 1979, str. 47-61
- [5] Wittie L. D.
Communication Structures for Large Networks of Microcomputers
IEEE Trans. on Computers C-30,4 (April 1981), str. 264-273

THE RING NETWORK OF MICROCOMPUTERS

An implementation of ring network in MERA-SYSTEM Laboratories is presented. The main goal of design was to meet simplicity of information exchange with simplicity of net nodes. The principles of the protocol are :

- information is supplied with address header
- sender node buffers information while sending
- sender removes its own information from the ring
- information with wrong header is removed from the ring by the node that checked an error
- rigorous control of headers prevents infinite circulation of 'ownerless' information with wrong header
- reliability is increased by retransmissions
- unification of nodes in the net is established
- no suspicious node exists in the net
- correct working of node does not depend on correct working of other nodes

Grounds of these principles against a background of other solutions are covered.

КОЛЬЦЕОБРАЗНАЯ СЕТЬ МИКРОКОМПЬЮТЕРОВ (РЕАЛИЗОВАНА В MERA-SYSTEM)

В докладе представлена кольцеобразная сеть микрокомпьютеров реализована в Предприятии компьютерных систем MERA-SYSTEM, на фоне других решений. В сети пытались согласить друг с другом простоту узлов сети с простотой протокола обмена информацией. Принято принцип унификации узлов.

Artur DUBIELEWICZ, Janusz RATAJCZAK*

sieć komputerowa,
transfer zbiorów,
teoria kolejek

OCENA WYKORZYSTANIA CZASU PRACY STACJI TRANSFERU ZBIORÓW
W WARUNKACH WYSTĘPOWANIA USZKODZEŃ

W pracy przedstawiono kolejkowy model Stacji Transferu Zbiorów sieci komputerowej, w którym uwzględniono występowanie uszkodzeń przerywających transmisję oraz nie powodujących zerwania połączenia. Rozwiązanie modelu pozwoliło na wyznaczenie współczynników: wykorzystania czasu pracy Stacji, strat czasu na powtarzanie transmisji i wysyłanie znaczników oraz dobranie częstości wysyłania znaczników maksymalizującej wykorzystanie Stacji.

1. WSTĘP

W złożonych systemach komputerowych, a do takich należy zaliczyć również sieci komputerowe, prawdopodobieństwo wystąpienia uszkodzenia jest stosunkowo duże. Błędne działanie systemu może być spowodowane błędami oprogramowania lub awariami sprzętu. Niektóre błędy np. przekłamanie na liniach teletransmisyjnych, utrata części przesyłanych danych, itp. mogą być poprawione w czasie trwania transmisji przez powtórne przesłanie utraconej informacji [4,6]. Inne, takie jak nieoperatywności podsieci komunikacyjnej lub awaria komputera obliczeniowego powodują przerwanie połączenia między komputerami. W takich przypadkach, po usunięciu przyczyny, konieczne jest wznowienie transmisji. W obu sytuacjach potrzebna jest informacja o ostatnim odebranych poprawnie rekordzie przesyłanego zbioru. Można ją uzyskać wysyłając co jakiś czas specjalny rekord tzw. znacznik. Odebranie znacznika jest potwierdzane. W ten sposób ostatni potwierdzony znacznik wyznacza punkt restartu transmisji.

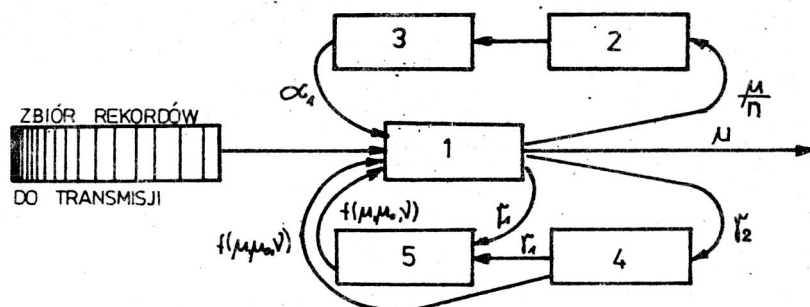
Ponieważ straty czasu, powodowane przez konieczność powtarzania części transmisji oraz wysyłanie i potwierdzanie znaczników są znaczące należy je minimalizować.

* Centrum Obliczeniowe Politechniki Wrocławskiej, Wrocław

W prezentowanej pracy przedstawiono kolejkowy model Stacji Transferu Zbiorów (STZ) [1]. Prezentowany model jest uogólnieniem omawianych w pracach [2,3] modeli. Model uwzględnia błędy przerywające transmisję i błędy, które nie powodują zerwania połączenia. STZ wysyła znacznik co n rekordów i każdy znacznik musi być potwierdzony. Rozwiązanie przyjętego modelu pozwala wyznaczyć: współczynnik wykorzystania STZ, straty czasu na powtarzanie transmisji i wysyłanie znaczników, a także optymalną częstość wysyłania znaczników.

2. MODEL STACJI TRANSFERU ZBIORÓW

Model STZ obsługującej transmisję jednego zbioru przedstawiono na rys. 1.



Rys. 1. Model Stacji Transferu Zbiorów
Fig. 1. The File Transfer Station Model

Założenia:

Procesy wysyłania rekordu zbioru w sieć oraz inicjacji wznowienia transferu mają rozkład wykładniczy o intensywnościach odpowiednio, μ i μ_0 . Po n wysłanych rekordach wysyłany jest znacznik. Każdy znacznik musi być potwierdzony, a więc wysyłanie następnych rekordów zbioru jest wstrzymywane do momentu otrzymania potwierdzenia. Średni czas wysłania znacznika wynosi $\frac{1}{\alpha}$, a średni czas oczekiwania na potwierdzenie $\frac{1}{\alpha_4}$. W modelu uwzględniono dwa strumienie uszkodzeń: przerywające połączenie (o intensywności γ_1) i nie przerywające połączenia (γ_2). Średni czas przywrócenia zerwanego połączenia oznaczono przez $\frac{1}{\beta}$. Po usunięciu uszkodzenia transmisja jest kontynuowana od ostatnio potwierdzonego znacznika.

Na rysunku 1 widać, że STZ może wykonywać jedną z pięciu czynności:

- wysyłanie rekordów zbioru - transmisja pierwotna (1)

- wysyłanie znacznika - po n wysłanych rekordach zbioru (2)
- oczekiwanie na potwierdzenie wysłanego znacznika (3)
- przywracanie połączenia między komputerami biorącymi udział w transmisji (4)
- wysyłanie rekordów zbioru - powtórna transmisja (5).

Stany STZ, wskazujące funkcję realizowaną przez stację w dowolnej chwili są określone przez parę (i, j) następująco:

- $(i, 0)$ dla $i \in \{0, 1, \dots, n-1\}$ - transmisja pierwotna; wysłano i rekordów po ostatnim znaczniku,
- $(n, 0)$ - wysyłanie znacznika;
- $(n+1, 0)$ - oczekiwanie na potwierdzenie znacznika;
- $(i, 1)$ - stan po przerwaniu połączenia; usuwanie uszkodzenia,
- $(i, 2)$ - nawiązanie połączenia; rozpoczęcie powtórnej transmisji;
- (i, j) - dla $i > 0$ i $j \in \{3, \dots, i+2\}$ - powtórna transmisja; przesłano powtórnie $j-3$ rekordy, należy przesłać powtórnie i rekordów.

Graf przejść między stanami STZ przedstawiono na rys. 2.

3. PODSTAWOWE WYNIKI

Oznaczmy przez $P(i, j)$ prawdopodobieństwo zdarzenia, że STZ znajduje się w stanie (i, j) . Rozwiązując układ równań różniczkowych opisujących model STZ zgodnie z przedstawionym na rysunku 2 grafem przejść między stanami otrzymujemy stacjonarne prawdopodobieństwa stanów (5).

Prawdopodobieństwa stanów, w których stacja prowadzi transmisję pierwotną wyrażają się następującym wzorem:

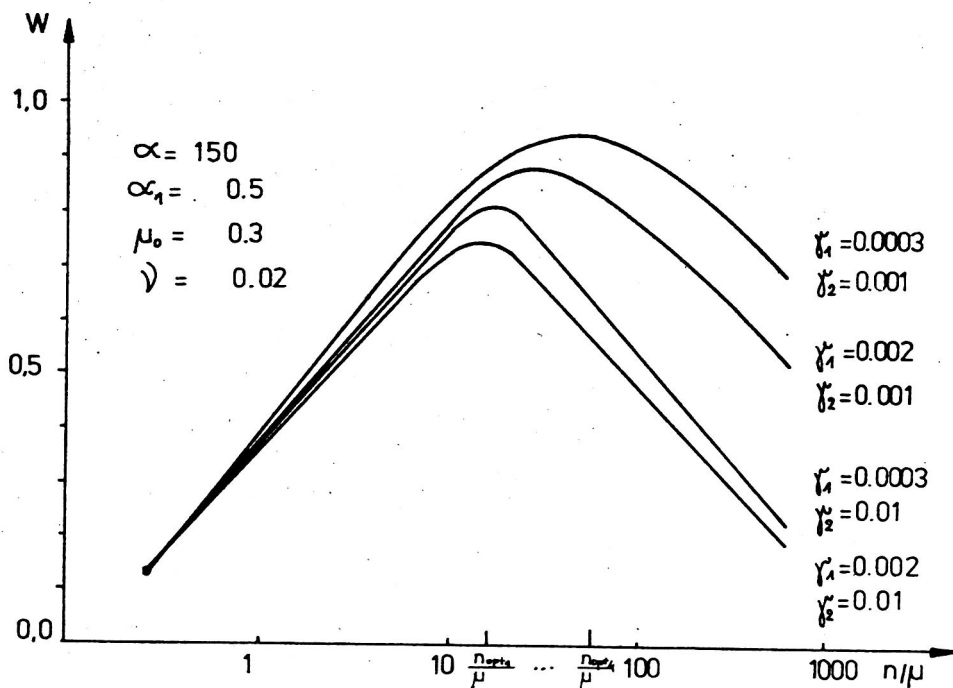
$$P(i, 0) = \frac{\alpha \alpha_0 \mu_0 \nu}{(n\alpha\alpha_0 + \mu(\alpha + \alpha_0))(\gamma_0 + \mu_0)(\gamma_0 + \nu)} \cdot \left(\frac{\mu}{\mu + \gamma_0 + \gamma_2} \right)^i \quad (1)$$

Ponieważ tylko te stany, w których stacja prowadzi transmisję pierwotną odzwierciedlają czas efektywnie wykorzystany na przesłanie zbioru pozostałe stany oznaczają straty czasu pracy spowodowane różnymi czynnikami, możemy określić współczynnik wykorzystania czasu pracy STZ następująco:

$$W = \sum_{i=0}^{n-1} P(i, 0) \quad (2)$$

Jeżeli uwzględnimy wyrażenie na $P(i, 0)$ i otrzymamy zależność:

$$W = \frac{\alpha \alpha_0 \mu_0 (\mu + \gamma_0 + \gamma_2) \nu}{(n\alpha\alpha_0 + \mu(\alpha + \alpha_0))(\gamma_0 + \mu_0)(\gamma_0 + \nu)(\gamma_0 + \gamma_2)} \left[1 - \left(\frac{\mu}{\mu + \gamma_0 + \gamma_2} \right)^n \right] \quad (3)$$



Rys. 3. Wykres zależności współczynnika wykorzystania STZ od względnej częstości wysyłania znaczników

Fig. 3. The W coefficient estimated for the various parameters of the FTS, communication subnetworks and data files.

Na podstawie zależności (3) można wyznaczyć optymalną (maksymalizującą wykorzystanie STZ) liczbę rekordów między znacznikami (n_{opt}) uwzględniającą zarówno charakterystyki niezawodnościowe sieci ($\alpha, \alpha_1, \mu_1, \mu_0$) jak i szybkość działania sieci oraz charakterystykę przesyłanych rekordów ($\gamma_1, \gamma_2, \gamma$). Stosowanie optymalnej częstości przysyłania znaczników pozwala na maksymalne w danych warunkach wykorzystanie możliwości sieci.

4. UWAGI KOŃCOWE

W artykule przedstawiono kolejkowy model działania STZ w warunkach występowania uszkodzeń. Prezentowany model pozwala na wyznaczenie stopnia wykorzystania STZ oraz określenie optymalnej częstości wysyłania znaczników (n_{opt}) z uwzględnieniem takich czynników jak: niezawodność elementów sieci komputerowej, szybkość przesyłania, wielkość rekordów

zbioru, itd. . Dokładna znajomość Nopt pozwala na lepsze wykorzystanie możliwości transmisyjnych sieci.

LITERATURA

- [1] Dubielewicz A. i inni, "Stacje Transferu Zbiorów", część I i II, Raporty CO PWR, Sprawozdania nr 34/82 i 4/83, Wrocław 1982, 1983.
- [2] Dubielewicz A., Ratajczak J. i inni, "Odnowa transferu zbiorów w sieci komputerowej", Materiały konf. VIII Międzynarodowej Konferencji FTSD'85, Katowice 1985, str. 239-246.
- [3] Dubielewicz A., Ratajczak J., "Kolejkowy model Stacji Transferu Zbiorów w warunkach występowania uszkodzeń", I Krajowa Konf. Sieci komputerowe, Protokoły, usługi, Wrocław 1985, Zeszyt Prace Naukowe COPWR, Konferencje nr 4/86, str. 49-54.
- [4] Kaszuba R., "Protokół Transferu Zbiorów dla MSK", Raport CO PWR serii SPR nr 16/83, Wrocław 1983.
- [5] Takeas L., Introduction to the theory of Quens, Oxford University Press, Inc. New York 1962.
- [6] ISO, Information Processing Systems - Open System Interconnection - Basic Reference Model, IS 7498, 1984.

THE FILE TRANSFER STATION TIME UTILIZATION UNDER FAULTS OCCURANCE

In the paper, the File Transfer Station queuing model of a computer network with faults which do not break or break interconnection between computers, is presented.

Basing on the solution of the proposed model the FTS's time utilization coefficient, the waste time for retransmission of failed records and sending marks and the optimal frequency of sending marks are computed. Obtained solution of the model is exemplified.

ОЦЕНКА ИСПОЛЬЗОВАНИЯ ВРЕМЕНИ РАБОТЫ СТАНЦИИ ПЕРЕДАЧИ ФАЙЛОВ В УСЛОВИЯХ НАЛИЧИЯ ПОВРЕЖДЕНИЙ

В статье представлена очередная модель Станции Передачи Файлов СПФ для вычислительной сети, в которой учтено выступание повреждений, не причиняющих к прекращению связи между вычислительными системами. Решение модели даёт возможность осуществить коэффициенты, характеризующие передачу файлов такие как: использование СПФ, потери из-за сбоев сети и т д , а также вычисления частоты передачи маркеров, максимизирующей коэффициенты использования СПФ.

Iwona DUBIELEWICZ^x, Zbigniew FRYŻLEWICZ^x,
Krystyna KOŁEŚNIK^x, Witold KOMOROWSKI^x,
Leszek MISIURA^x

sieć komputerowa,
protokoły, testowanie

APARATURA POMIAROWO-DIAGNOSTYCZNA DLA SIECI KOMPUTEROWEJ

W pracy przedstawiono możliwości i zasady monitorowania protokołów sieciowych w sieciach transmisji danych przy użyciu aparatury pomiarowo-diagnostycznej SPS. Omówiono podstawowe komendy monitorowania, ich parametry oraz miary wyznaczane podczas ich realizacji dla różnych typów protokołów sieciowych. Zamieszczono przykładową postać końcowego raportu monitorowania oraz obserwowalną postać zarejestrowanych danych.

Aparatura pomiarowo-diagnostyczna SPS zrealizowana w Centrum Obliczeniowym Politechniki Wrocławskiej umożliwia podglądanie i rejestrowanie przesyłanych w systemie teleprzetwarzania komunikatów, a także ma możliwość generowania jednostek protokołu i wysyłania ich do systemu w celu zbadania reakcji jakie powinny wywoływać.

Aparatura bazuje na modułowym systemie mikroprocesorowym MSM firmy IMPOL (z µp 8085), wyposażonym w stację dysków oraz specjalnie skonstruowane pakiety sprzęgające (typów A,B,C) umożliwiające włączenie aparatury do sieci w celu monitorowania i testowania.

SPS ma możliwość rejestrowania przepływu danych na styku S2 poprzez monitorowanie:

- stanu obwodów styku S2,
- transmisji według asynchronicznych protokołów znakowych,
- transmisji według synchronicznego protokołu BSC,
- transmisji według synchronicznych protokołów zorientowanych bitowo tj. LAPB, SDLC, X25.

Dodatkowo aparatura ma możliwość prowadzenia testowania standardowego testu wg zalecenia V52, do pomiaru bitowej i blokowej stopy błędów, lub testowania programowanego poprzez wysyłanie w sieć testu zdefiniowanego przy użyciu języka opisu testu, interpretowanego poprzez SPS [2] .

^x Centrum Obliczeniowe Politechniki Wrocławskiej, Wrocław

1. ZASADY MONITOROWANIA I TESTOWANIA

Użytkownik określa /w SPS/ wybrany typ pracy za pomocą komend, których pełna definicja wprowadzana jest do SPS w trybie dialogowym. Dla wszystkich typów monitorowania (z wyjątkiem monitorowania inter-face'u) warunki te definiują: sposób monitorowania dwóch strumieni danych (do/z DCE) łącznie lub rozdzielnie, rodzaje pułapek określających warunki zakończenia monitorowania oraz szczegółowe parametry transmisji podlegającej monitorowaniu, zależne od typu monitorowania a także konieczność opatrywania odbieranych jednostek protokołu czasem ich odbioru. Do pułapek, które mogą być zdefiniowane dla wszystkich typów monitorowania należą: upływ czasu trwania pomiaru, wystąpienie limitu zdefiniowanych wzorcy (sekwencji 1-6 bajtów), wypełnienie bufora rejestrującego. W czasie prowadzenia pomiaru podglądane jednostki danych protokołu są wprowadzane do bufora cyklicznego (o pojemności 100 jednostek 54-ro bajtowych dla każdego strumienia) i są aktualizowane liczniki zdarzeń zdefiniowanych dla danego typu monitorowania. Uzywane wyniki mogą być wyprowadzone w postaci raportu, na życzenie użytkownika SPS w trakcie wykonywania pomiaru, bez jego przerywania, a automatycznie są wyprowadzane po zakończeniu pomiaru. Przykład raportu dla monitorowania protokołu LAPB pokazuje rys. 1.

* LAPB PROTOCOL MONITOR *

START	12:44:17	DURATION	00:00:13:140
	STOPPED	TRAP	
RESULTS	TX		RX
U	0		0
S	0		15
I	0		17
FCS ERRORS	0		0
ABORTS	0		0
PATTERNS	0		0

Rys. 1. Raport końcowy dla monitorowania protokołu LAPB
Fig. 1. LAPB protocol monitoring raport

Zainicjowany pomiar może być w każdej chwili zatrzymany z wyprowadzeniem dotychczasowych wyników oraz powtórzony dla tych samych warunków pomiaru. Po zakończeniu pomiaru ostatecznie zarejestrowana zawartość bufora nadawczo-odbiorczego może być wyświetlana na ekranie przy użyciu komend redagująco-monitorujących. Za ich pomocą można zdefiniować wyświetlany strumień (RX, TX, RX+TX łącznie), tryb wyświetlania proceduralny, nie-proceduralny oraz długość wyświetlanej jednostki (krótka, długa) i wyświetlany obszar. Zawartość bufora jest wyprowadzana na ekran w postaci wydzielonych jednostek protokołu, z których każda opatrzona jest stan-

standardowym nagłówkiem określającym: typ strumienia, stan jednostki (sygnalizujący poprawność jej odebrania) oraz opcjonalnie czas odbioru jednostki. Fragment bufora wyświetlanego w trybie nieproceduralnym pokazuje rys. 2.

```

TX + RX STREAM
TX 05:57:51:060 32 10 61 FF                               .../.
RX 05:57:51:300 32 37 FF FF FF FF FF FF FF FF FF FF FF FF .....
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .....
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .....
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .....
    FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF .....
    65 64 6E FE 65 64 C0 C0 80 80 5A                        ...>...!!..!
TX 05:57:51:860 32 10 70 FF                               ....
RX 05:57:52:140 32 02 27 F1 C7 11 C2 60 13 F1 F2 F3 F4 F5 ...1G.B-12345
    F6 F7 F8 F9 F0 98 A6 85 99 A3 A8 A4 89 96 67890ЯВЕРТЫ
    97 81 A2 84 86 87 88 91 92 93 A9 A7 83 A5 ПАСДФРХЙКЛЗБ
    82 95 94 D8 E6 C5 D9 E3 E8 E4 C9 D6 D7 C1 БНMQWERTYUIO
    E2 C4 C6 C7 C8 D1 D2 D3 E9 E7 C3 E5 C2 D5 SDFGHJKLZXCV

```

Rys. 2. Fragment buforów nadawczo-odbiorczych wyświetlanych w trybie nieproceduralnym

Fig. 2. Fragment of sending-receiving buffer displayed in nonprocedural mode

Wykrycie warunku uniemożliwiającego wykonanie komendy w zadany sposób, powoduje wyprowadzenie odpowiedniego komunikatu informującego i zaniechanie akcji.

2. MONITOROWANIE OBWODÓW STYKU S2

Procedura monitorowania wywoływana komendą MI, przy użyciu pakietu sprzęgającego A powoduje monitorowanie 14 obwodów styku S2 o numerach 103..115 i 125. Monitorowanie odbywa się poprzez rejestrowanie stanu logicznego obwodu (ON, OFF) w zadanych odstępach czasu próbkowania. Po zakończeniu próbkowania (75 próbek) jest wyprowadzany raport końcowy w postaci wykresu stanu linii w momentach próbkowania.

3. MONITOROWANIE TRANSMISJI WEDŁUG PROTOKOŁÓW ASYNCHRONICZNYCH

Przy użyciu pakietu sprzęgającego B (z układem 8251) za pomocą komendy MA monitorowana jest transmisja półduplexowa, której szczególne parametry tj. szybkość transmisji (50..9600), długość znaku (5..8), długość bitu stop (1, 1.5, 2), rodzaj parzystości (EVEN, ODD, NONE) są definiowane w parametrach komendy. Oprócz wymienionych pułapką może być wykrycieadanego limitu błędów parzystości. Podczas monitorowania jest mierzona liczba znaków, liczba błędów parzystości,

liczba wystąpień wzorca.

4. MONITOROWANIE TRANSMISJI WEDŁUG PROTOKOŁU SYNCHRONICZNEGO BSC

Transmisja według protokołu asynchronicznego BSC monitorowana jest także przy użyciu pakietu sprzęgającego B, za pomocą komendy MB. Szczegółowe parametry tj. kod transmisji (ASCII, EBCDIC), szybkość transmisji (600..9600) i rodzaj parzystości (jw.), są ustalane podczas definiowania komendy. Istnieje możliwość zdefiniowania pułapek jak dla komendy MA. Podczas monitorowania mierzona jest liczba błędów parzystości, liczba wystąpień wzorca, liczba komunikatów rozpoczynających się od określonego znaku (dla 10 znaków sterujących tj. ENQ, EDT, STX, SOH, ACK0, ACK1, NACK, RVI, WACK, TTD). Przeglądanie zarejestrowanego bufora nadawczo-odbiorczego może odbywać się w trybie nieproceduralnym (wyświetlanie komunikatów określonego strumienia opatrzonych nagłówkiem w postaci szesnastkowej i w kodzie ASCII) lub w trybie proceduralnym, (w którym znaki sterujące są wyświetlane w postaci ich mnemoników, pozostałe zaś w postaci szesnastkowej).

5. MONITOROWANIE TRANSMISJI WEDŁUG PROTOKOŁÓW SYNCHRONICZNYCH ZORIENTOWANYCH BITOWO

Transmisja prowadzone według protokołów LAPB, SDLC i X25 mogą być monitorowane przy zastosowaniu pakietu sprzęgającego C (z układem 8273) za pomocą komendy MB. Szczegółowe parametry transmisji podawane przy definiowaniu komendy to typ protokołu (LAPB, SDLC, X25), szybkość transmisji (600..9600), miejsce włączenia SPS w system teleprzetwarzania (local, remote) i sposób kodowania (NRZI, NRZ). Pułapką charakterystyczną dla tego rodzaju monitorowania może być limit dopuszczalnych błędów FUS, liczba sekwencji ABORT, liczba jednostek protokołu określonej grupy. Wyróżniono trzy grupy jednostek protokołów tj. ramek dla protokołów LAPB i SDLC (odpowiadające ramkom sterującym nienumerowanym, ramkom nadzorczym numerowanym i ramkom informacyjnym) oraz cztery grupy pakietów (nawiązywania połączenia, informacyjne, sterujące i restartu). Po zakończeniu pomiaru przeglądanie bufora nadawczo-odbiorczego może odbywać się także w trybach nieproceduralnym w postaci jak dla monitorowania synchronicznego oraz proceduralnym uzależnionym od typu protokołu. Dla protokołów LAPB i SDLC w trybie proceduralnym ramka jest rozpakowywana i osobno wyświetlany jest: adres ramki (szesnastkowo), typ ramki (rozkaz, odpowiedź) i wartość bitu P/P, rodzaj ramki (I, S, V), mnemonik ramki a dla ramek numerowanych numer ramki i potwierdzenia odpowiednio po komentarzu N(S)= i N(R)=.

Dla protokołu X25 pakiet jest rozpakowany i osobno wyświetlany jest: GF1, numer kanału logicznego, mnemonik pakietu, numer pakietu i numer

potwierdzenia z komentarzami odpowiednio $P(S)=$ i $P(R)=$. Fragment bufora zarejestrowanego podczas monitorowania protokołu X25 wyświetlanego w trybie proceduralnym pokazano na rys. 3.

EMPTY BUFFER TX RX STREAM

```

RX 12:48:00:540 01 OFFE DTE DATA P(R)=1 P(S)=1
                02 F0 80 00 7D 5B C2 11 5A 7A D7 D9 E9 C5
                E2 D3 C9 D1 40 40 40 40 40 40 40 40 40
                40 40 40 40 40
RX 12:48:01:560 01 OFFE DTE IR P(R)=3
RX 12:48:01:600 01 OFFE DTE DATA P(R)=3 P(S)=2
                04 60 00 26 07
RX 12:48:04:580 01 OFFE DTE DATA P(R)=1 P(S)=0
                04 60 00 26 0D
LAST ENTITY OF RX
RX 12:48:06:060 01 OFFE DTE DATA P(R)=4 P(S)=3
                04 60 00 26 10

```

Rys. 3. Fragment bufora nadawczo-odbiorczego strumienia RX z jednostkami protokołu X25 wyświetlanego w trybie proceduralnym

Fig. 3. Fragment of sending-receiving buffer displayed in procedural mode /X25 protocol/

Literatura

- [1] I. Dubielewicz i inni: "Aparatura pomiarowo diagnostyczna dla sieci SKJS/2". Centrum Obliczeniowe, Raport serii SPR Nr 12/86.
- [2] I. Dubielewicz i inni: "Testowanie programowe w sieci transmisji danych", Centrum Obliczeniowe, Raport serii PRE Nr /87.

SPS-DIAGNOSTICS AND MEASURING SYSTEM IN COMPUTER NETWORKS

The paper presents principle of data streams monitoring using the SPS measuring and diagnostics system connected to DTE - DCE interface in communication network. Basic monitor modes of the SPS and relevant commands for various transmission protocols are described. Some samples of resulting report form are given.

АППАРАТУРА ИЗМЕРИТЕЛЬНО-ДИАГНОСТИЧЕСКАЯ ДЛЯ СЕМИ ПЕРЕДАЧИ ДАННЫХ

В работе представляется функции аппаратуры измерительно-диагностической SPS предназначенной для оценки работы сети передачи данных.

Обсуждается типы подлежащих испытанию с применением SPS, сетевых протоколов, измерительные каналы, их параметры определяющие условия измерения и получаемые результаты. Помещается пример конечного рапорта мониторинга и подлежащая для наблюдения форму зарегистрированных данных.

Iwona DUBIELEWICZ^x, Zbigniew PRYZLEWICZ^x,
Krystyna KOLESNIK^x, Witold KOMOROWSKI^x,
Leszek MISIURA^x

sieć komputerowa,
protokoły, testowanie

TESTOWANIE PROGRAMOWANE W SIECI TRANSMISJI DANYCH

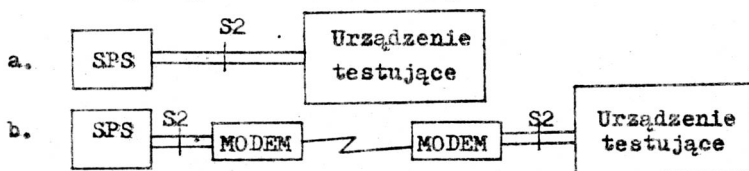
W pracy przedstawiono zasady testowania programowanego w sieci transmisji danych przy pomocy aparatury pomiarowo-diagnostycznej SPS. Podano systemy pracy SPS wraz z parametrami transmisji i typami protokołów, które można definiować dla potrzeb testowania różnych urządzeń. Opisano język programowania testów TPL /struktury danych i instrukcje/ oraz umieszczono przykładowy program testu do komunikacji z węzłem sieci MSK na poziomie protokołu liniowego.

Problemy testowania i weryfikacji poprawności działania urządzeń, które są włączane do pracy w sieciach komputerowych, jak również trudności występujące przy uruchamianiu nowego sprzętu w sieci powodują, że wzrasta zapotrzebowanie na sprzęt specjalizowany, który by te prace usprawnił. Aparatura SPS jest przyrządem, który oprócz monitorowania pracy elementów sieci [2] może prowadzić testowanie działania wybranego urządzenia sieci. Testowanie jest prowadzone na zasadzie dialogu zgodnie z zaimplementowanym w testowanym urządzeniu protokołem.

W pracy przedstawiono systemy testowania z wykorzystaniem SPS, omówiono język do budowy programów testowych i podano przykładowy program testowy do komunikacji z węzłem sieci MSK.

1. KONFIGURACJE SPRZĘTOWE DO TESTOWANIA PROGRAMOWANEGO

Aparatura SPS musi być podłączona do sieci urządzenia testowanego poprzez styk S2. Testowanie może być prowadzone lokalnie (rys. 1 a) lub zdalnie z wykorzystaniem modemów (rys. 1 b).



Rys. 1 Systemy testowania z wykorzystaniem SPS
Fig. 1 Testing systems with SPS

^x Centrum Obliczeniowe Politechniki Wrocławskiej, Wrocław

W testowaniu można wykorzystać linie półdupleksową lub dwupleksową o szybkości z zakresu 600 - 9600 b/s. Komunikacja z testowanym urządzeniem może wykorzystywać protokoły znakowe (np. BSC) lub zorientowane bitowo (HDLC/SDLC), których większość parametrów można zdefiniować w SPS (np. długość znaku, typ parzystości, wzorzec synchronizacji) [1].

2. JĘZYK PROGRAMOWANIA TESTÓW

Język programowania testów (TPL) jest językiem specjalizowanym, który operuje na następujących strukturach danych:

- bufor (zawierające komunikaty/ramki/pakiety),
- liczniki,
- stopery.

Bufory stanowią podstawowe struktury danych, a zawarte w nich informacje są transmitowane do/z testowanego urządzenia, porównywane z wzorcami i analizowane. Bufory mogą być zapisywane przez użytkownika SPS (do wysłania w linie) lub są wypełniane danymi odbieranymi z linii. W aktualnej wersji istnieje 16 buforów, których rozmiary może ustalić użytkownik (4-2400 bajtów/bufor).

Liczniki i stopery są pomocniczymi strukturami danych, które umożliwiają organizację pętli (liczniki) oraz odmierzanie czasu (stopery). W aktualnej wersji istnieje 8 liczników dwubajtowych oraz 8 stoperów dla odmierzania czasu 20 ms do 5 sek.

W języku TPL istnieje 21 instrukcji, w tym:

- 4 rozkazy licznikowe,
- 8 rozkazów skoków,
- 2 instrukcje dla stoperów,
- 2 instrukcje transmisyjne (nadawanie/ odbiór bufora),
- 3 instrukcje operacji na buforach,
- 1 instrukcja zawieszenia programu (na podany czas),
- 1 instrukcja stopu.

Możliwość realizacji transmisji dwupleksowej w programie sekwencyjnym uzyskano poprzez przyjęcie zasady komunikacji bezpotwierdzeniowej. Po zainicjowaniu nadawania i odbioru program może sprawdzić (po podanym czasie) wykonanie każdej z tych operacji (nie czeka się więc w pętli na ich wykonanie).

Na rys. 2 podano przykładowy program do komunikacji z węzłem sieci MSK na poziomie protokołu liniowego. Bufory B0, B2 i B4 zostały wypełnione przez operatora SPS odpowiednio ramkami SAEW, UA i I (z pakietem RESTART INDICATION).

1	10	SEND	B0	; nadaj SARM
2	50	WAIT	5	; czekaj 0,1 s
3	43	JNS	15	; błąd nadaw
4	20	RCEV	B1	; odbieraj do B1
5	50	WAIT	20	; czekaj 0,4 s
6	44	JNR	1	; próbuj od nowa
7	30	CMP	B1, B2	; czy odebr UA
8	42	JNEQ	14	; brak UA
9	10	SEND	B4	; nadaj RESTART
10	50	WAIT	5	; czekaj
11	43	JNS	15	; błąd nadaw
12	20	RCEV	B3	; odbieraj
13	50	WAIT	50	; czekaj 1 s
14	00	HALT		; stop poprawny
15	00	HALT		; stop po błędzie

Rys. 2 Przykładowy program testowy
Fig. 2 Example of testing program

Użytkownik pisze program wykorzystując prosty edytor tekstu wbudowany do systemu, który wykrywa podstawowe błędy składniowe. Napisany program wykonywany jest interpretacyjnie przy czym są sygnalizowane stwierdzone błędy semantyczne.

W przypadku poprawnego zakończenia wykonania programu drukowany jest raport, w którym oprócz podania ustalonych parametrów transmisji są wyprowadzane stany liczników oraz dane o wykorzystywanych buforach i stoperach. Przeglądanie zawartości buforów jest możliwe przy stosowaniu specjalnej komendy SPS.

3. PODSUMOWANIE

Aparatura SPS wykorzystywana w trybie testowania programowanego wydaje się być uniwersalnym i wygodnym narzędziem do uruchamiania pilotowych wersji sieci i poszczególnych jej komponentów, jak również do testowania nowych urządzeń włączanych do sieci pracującej. Należy jednak zwrócić uwagę na konieczność bardzo dobrej znajomości protokołu a w szczególności režimu czasu pracy testowanego urządzenia przy pisaniu programów testowych. Dodatkową zaletą SPS jest możliwość generowania sytuacji błędnych i testowania zachowania się urządzeń w takich warunkach (nie zawsze jest to możliwe w pracujących systemach).

Literatura

- [1] Dubielewicz L. i inni, "Aparatura pomiarowo-diagnostyczna dla sieci SK JS/2", Centrum Obliczeniowe PWr, Raport serii SPR 12/86
- [2] Dubielewicz L. i inni, "Aparatura pomiarowo-diagnostyczna w sieci komputerowej", Materiały konfer. "Sieci komputerowe - teoria, technika zastosowania" Wrocław 1987.

PROGRAMMED TESTING IN COMPUTER NETWORKS

The principles of programmed testing of the communication network equipment using the SPS measuring and diagnostics system is described. It is shown how the SPS can be programmed in an easy - to - learn TPL language to fulfill requirements of various protocols. Example of the test on-line protocol level for MSK network is given.

ПРОГРАММИРУЕМОЕ ТЕСТИРОВАНИЕ В СЕТИ ПЕРЕДАЧИ ДАННЫХ

В работе представляется принципы программируемой тестировки в сети передачи данных при помощи измерительно-диагностической аппаратуры SPS. Сообщается системы работы SPS вместе с параметрами передачи и типами протоколов, которые определяются для тестировки разных устройств. Описывается язык программирования тестов TPL (структуры данных и инструкции) а также помещается в виде примера программу теста для соединений с узлам сети MSK на уровне линейного протокола.

sieć SYNKOM,
synchronizacja

Henryk GUT-MOSTOWY^x

ZAGADNIENIA SYNCHRONIZACJI SIECI "SYNKOM"

Podaje się topologię i strukturę techniczną sieci "SYNKOM". W tak wyróżnionych przekrojach sieci przedstawia się zasady synchronizacji: elementowej, ramkowej i blokowej oraz omawia się metodę synchronizacji węzłowych generatorów podstawy czasu.

1. WSTĘP

W Zakładzie Teleinformatyki Instytutu Łączności oraz Instytucie Telekomunikacji Politechniki Warszawskiej prowadzi się prace nad strukturą techniczną Krajowej Synchronicznej Komutowanej Sieci Transmisji Danych, nazywanej tymczasowo siecią "SYNKOM" [1,3÷9]. W pracach tych nie mało uwagi poświęca się problemom, które dla sieci tego rodzaju mają znaczenie fundamentalne. Tworzą bowiem kompleks zagadnień jej szeroko rozumianej synchronizacji. Prezentacja rozwiązań wybranych elementów tej "warstwy" projektowanej sieci jest zasadniczym celem niniejszego referatu.

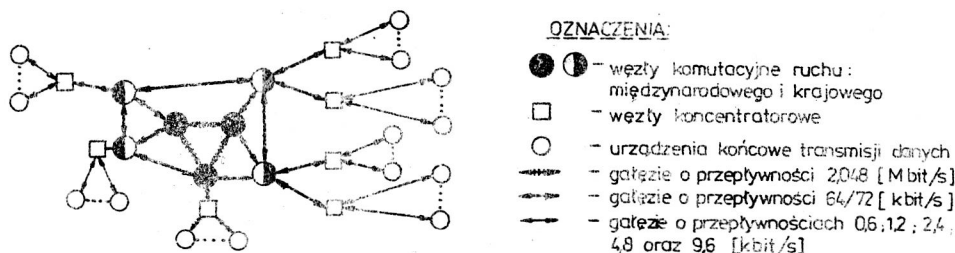
2. TOPOLOGIA I STRUKTURA TECHNICZNA SIECI

Sieć "SYNKOM" można przedstawić jako złożony przestrzennie rozproszony obwód, utworzony ze zbioru M-węzłów, połączonych gałęziami dwukierunkowymi w trójpoziomą strukturę uzależnień; zob. rys. 1. Węzłami tak rozumianej sieci są: centrale komutacyjne z czasową komutacją kanałów, koncentratory oraz urządzenia końcowe transmisji danych. Gałęziami są zaś systemy transmisji sygnałów cyfrowych o przepływnościach: 0.6, 1.2, 2.4, 4.8, 9.6, 64/72/ [kbit/s] oraz 2.048 [Mbit/s].

Dla prawidłowego funkcjonowania układu tego rodzaju jak sieć "SYNKOM", oprócz szeregu innych, muszą zachodzić dwa warunki. Jednym jest wysoka wiarygodność transmisji danych w gałęziach sieci, zaś drugim - zgodność przepływności binarnych w poszczególnych jej węzłach.

^x Instytut Łączności, Warszawa

Realizacja warunku pierwszego wymaga utrzymania urządzeń odbiorczych gałęzi w stanie wyrównania: elementowego, ramkowego i znakowego. Warunek drugi zachodzi zaś wówczas, gdy w całej sieci zachowana jest jednolita skala czasu.



Rys.1. Topologia sieci "SYNKOM"
Fig.1. Topology of "SYNKOM" network

3. PROCESY SYNCHRONIZACJI W GAŁĘZIACH SIECI

Funkcją podstawową gałęzi sieci jest transmisja binarnego, synchronicznego sygnału elektrycznego pomiędzy przestrzennie odległymi punktami jego wytwarzania i odbioru. Warunkiem jej realizacji jest odtworzenie sygnału elementowej skali czasu. W przypadku gałęzi o przepływności 64/72 [kbit/s] dodatkowo pożądanym jest odtworzenie fali nośnej, niezbędnej dla detekcji synchronicznej, gwarantującej optymalne warunki odbioru. W urządzeniach sieci "SYNKOM" do tych celów stosuje się binarne pętle fazowe. Charakteryzują się one: dużą niezawodnością działania, powtarzalnością parametrów oraz łatwością realizacji w różnych wariantach elementowych.

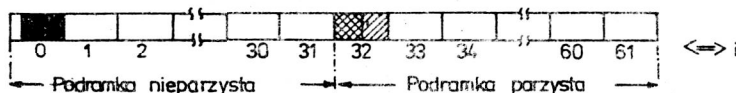
Sygnały przesyłane zarówno w gałęziach 1-go, jak i 2-go poziomu sieci "SYNKOM" mają strukturę ramek synchronizacyjnych ze skupionym wzorem synchronizacyjnym; zob. rys. 2. Daje to możliwość realizacji układów synchronizacji ramkowej działających wg. tego samego algorytmu na wymienionych poziomach sieci.

Przy określonej liczbie bitów "N" tworzących ramkę i danej liczbie bitów "R" tworzących wzór synchronizacyjny następujące trzy elementy w pełni definiują proces odzyskiwania i utrzymywania stanu synchronizacji ramkowej odbiornika. Są nimi: struktura kodowa wzoru synchronizacyjnego, algorytm poszukiwania fazy ramki oraz kryterium utraty i odzyskania synchronizacji ramkowej.

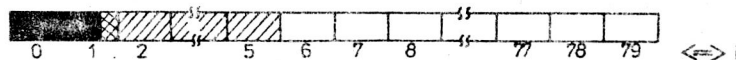
W sieci "SYNKOM" stosuje się nieprzesuwalne sekwencje synchronizacyjne, tzn. takie R-elementowe ciągi binarne, które złożone ze sobą nie dają na styku słów kodowych o takiej jak one strukturze. Poszukiwanie fazy ramki realizuje się wg algorytmu porównywania równoległego i przesuwania jednobitowego [2]. Kryterium utraty i odzyskania syn-

chronizacji ramkowej konstruuje się zaś na bazie testów sekwencyjnych Valda'a [10].

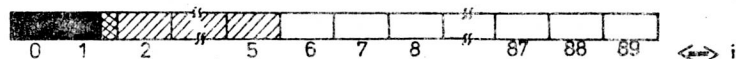
a) Ramka strumienia 2048 [Mbit/s]; $N=512$ [bit], $R=7$ [bit]



b) Ramka strumienia 640 [kbit/s]; $N=640$ [bit], $R=14$ [bit]



c) Ramka strumienia 720 [kbit/s]; $N=720$ [bit], $R=14$ [bit]



Oznaczenia:

- i - numer kolejny 8-mio bitowej szczeliny czasowej ramki
-  - szczeliny kanału sygnalizacyjnego
-  - szczeliny kanałów danych
-  - szczeliny wzoru synchronizacyjnego
-  - szczeliny niewykorzystane

Rys.2. Struktury ramek w sieci "SYNKOM"
Fig.2. Structures of frames in "SYNKOM" network

Od gałęzi poziomu trzeciego wymaga się, aby możliwą była permanentna, równoczesna i wzajemnie niezależna transmisja sygnałów kodowych obwodów "T" i "C" styku X.21 dla kierunku transmisji od terminala do koncentratora oraz transmisja sygnałów obwodów "R" i "I" dla kierunku przeciwnego. Dążąc do maksymalizacji zasięgu transmisji, w sieci "SYNKOM" zastosowano w tym celu alfabetowy kod transmisyjny [8]. Zysk na zasięgu okupiono tu jednakowoż koniecznością wyrównywania znakowego dekodera. Dzięki właściwościom autosynchronizacyjnym zastosowanego kodu - objawiającym się tym, że w dowolnym strumieniu bitów, powstałym z połączenia dowolnych dwóch słów kodowych, prawdopodobieństwo zdekodowania losowo wybranej 16-ki kolejnych bitów jako słowa kodowego jest równe 0.3 - nie stwarza to istotnych trudności realizacyjnych dekodera. Odzyskiwanie i utrzymywanie stanu wyrównania znakowego może być bowiem realizowane wg zasad właściwych dla procesu synchronizacji ramkowej strumieni danych 1-go i 2-go poziomu sieci.

4. SYNCHRONIZACJA ZEGARÓW SIECI

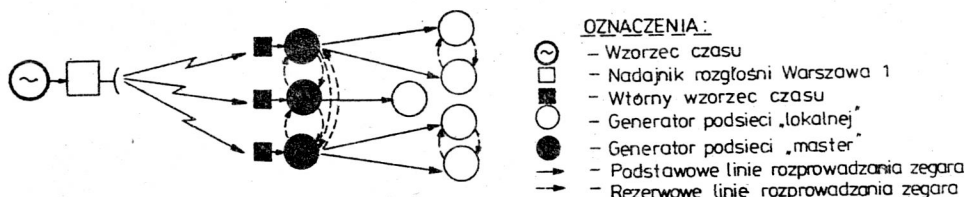
Warunkiem niezbędnym dla prawidłowego funkcjonowania sieci synchronicznych jest utrzymanie stanu zgodności przepływności binarnych w węzłach komutacyjnych sieci, a więc synchronizacja częstotliwości drgań generatorów węzłowych, wytwarzających lokalne skale czasu.

Synchronizację zegarów sieci "SYNKOM" realizować się będzie wg metody "master-slave" z samomodyfikującą się siecią rozprowadzania

informacji synchronizacyjnych [5]. Według tej metody, na poziomie najwyższym znajduje się laboratorium czasu. Wyposażone jest ono w wysokostabilne i niezawodne wzorce czasu. Wytwarza ono sygnały taktowania, które synchronizują częstotliwość drgań generatora fali nośnej 227 [kHz] rozgłośni radiowej "Warszawa program pierwszy". Informacje synchronizacyjne, zawarte w częstotliwości sygnału nośnego tejże radiostacji, odbierane są przez tzw. wtórne wzorce czasu [12], zlokalizowane w centralach międzynarodowych. Wytwarzają one okresowy sygnał odniesienia dla mikroprocesorowych pętli fazowych, generujących sygnały taktowania lokalnych skal czasu węzłów komutacyjnych tzw. podsieci "master" - drugi poziom synchronizacji. Dane o częstotliwości chwilowej generatorów podsieci "master", zawarte w szybkości modulacji sygnałów transmitowanych w gałęziach sieci, docierają do /zob. rys. 3/:

- innych generatorów podsieci "master", gdzie wykorzystywane są do obliczania tzw. kombinowanej fazy odniesienia oraz
- generatora o najwyższym priorytecie w podsieci lokalnej, sterując tam pracą pętli fazowej o takiej samej strukturze jak pętla fazowa drugiego poziomu synchronizacji.

Zgodnie z omawianą metodą, zegary węzłów podsieci "master" regulowane są sygnałem odniesienia z wtórnego wzorca czasu zawsze wtedy, gdy odbiera on wiarygodne radiowe sygnały synchronizacyjne. W przypadku przeciwnym - kryterium dla regulacji pętli stanowi permanentnie modyfikowana kombinowana faza odniesienia. Zasada niniejsza przenosi się na podsieci lokalne - trzeci poziom synchronizacji. Najwyższy priorytet regulacji pętli przypisany jest tu sygnałom odbieranym od węzła "master" nadzorującego daną podsieć.



Rys.3. Metoda synchronizacji zegarów w sieci "SYNKOM"
Fig.3. Method of clock synchronization in "SYNKOM" network

5. PODSUMOWANIE

Sieć "SYNKOM", a tym samym system jej synchronizacji, znajduje się obecnie na etapie zaawansowanych prac projektowo-modelowych. Duży procent nakreślonych wyżej problemów ma zatem charakter faktów dokonanych. Są jednakowoż i takie, jak np. synchronizacja zegarów sieci, które bezpieczniej jest traktować nie jako rozwiązania pewne, lecz raczej - najbardziej prawdopodobne.

LITERATURA

- [1] Z. Baran i inni: "Projekt koncepcyjny adaptacji i modernizacji modemu pierwotnogrupowego V.36S". Oprac. wew. ITPW, W-wa 1985r.
- [2] A. Dobrogowski: "Pewne aspekty probabilistyczne procesu odzyskiwania stanu synfazowości ramki w systemach wielokrotnych o podziale czasowym". Arch. Elektrotech. zesz. 2, 1980r.
- [3] H. Gut-Mostowy: "Urządzenie WT-X21 do nawiązywania połączeń w teleinformatycznej sieci SYNKOM dla terminali o styku V.24, V.28 CCITT". Oprac. wew. IŁ, W-wa 1985r.
- [4] H. Gut-Mostowy: "Wzywak teledacyjny WT-X.21 - charakterystyka ogólna". Przegląd Telekom. Nr 10, 1985r.
- [5] H. Gut-Mostowy: "Synchronizacja zegarów sieci według zmodyfikowanej metody master-slave". Kraj. Symp. Telekom. 86, zesz. 2, Rynia 1986r.
- [6] H. Gut-Mostowy i inni: "Translacja teledacyjna - urządzenie TRT /Projekt wstępny/". Oprac. wew. IŁ, W-wa 1986r.
- [7] M. Jarociński i inni: "Projekt studialny centrali telefoniczno-teleinformatycznej DSE sieci SYNKOM". Oprac. wew. ITPW, W-wa 1985r.
- [8] W. Lewandowski: "Zagadnienia transmisyjne w sieci SYNKOM". Kraj. Symp. Telekom. 86, zesz 2, Rynia 1986r.
- [9] K. Odolak i inni: "Projekt studialny transpleksera TPX". Oprac. wew. IŁ, W-wa 1985r.
- [10] A. Wald: "Posledowatelnyj analiz". Gosudarstwiennoe Izd. Fizyko-matem. Literatury, Moskwa 1960r.
- [11] B. Zbierzchowski i inni: "Projekt wstępny koncentratora". Oprac. wew. ITPW, W-wa 1986r.
- [12] Praca zbiorowa Zakł. Radiokom. Radiofonii i Telewizji: "Wtórny wzorzec częstotliwości - opis techniczny". Oprac. wew. IŁ, W-wa 1986r.

THE SYNCHRONIZATION PROBLEMS OF "SYNKOM" NETWORK

A draft of topology and technical structure of "SYNKOM" network is given. In such distinguished sections, principles of bit-synchronization, character-synchronization and frame-synchronization are shown. A method of clock-synchronization is also described.

ПРОБЛЕМА СИНХРОНИЗАЦИИ СЕТИ - "SYNKOM"

В работе определено топологию и техническую структуру сети "SYNKOM". В так выделенных уровнях сети описано общие принципы позиционной синхронизации, знаковой синхронизации и цикловой синхронизации. Обсуждается здесь тоже метод синхронизации узловых генераторов тактовой частоты.

ELŻBIETA HUDYMA^x
ELŻBIETA KOSMULSKA-BOCHENEK^x

sieć komputerowa
system pomiarowo-
sterujący, symulacja

BADANIE PRZEPŁYWU POMIAROWO-STERUJĄCEGO
W SYSTEMIE TRANSMISJI DANYCH SIECI KOM-
PUTEROWEJ

Tematem artykułu jest analiza ruchu pomiarowo-diagnostycznego dla założonego modelu systemu pomiarowo-sterującego sieci komputerowej. Określa się wpływ tego ruchu na taki parametr, jak czas obsługi pakietów użytkowych. W celu oszacowania parametrów ruchu pomiarowego wykorzystano metodę symulacji.

1. WSTĘP

Każda sieć komputerowa powinna posiadać integralnie wbudowany w jej architekturę system pomiarowo-diagnostyczno-sterujący (SPS). Celem takiego systemu jest optymalne zarządzanie zasobami sieci, rekonfiguracja i odnowa zasobów w warunkach uszkodzenia, rozliczanie użytkowników, pomiary efektywności sieci itp. W chwili obecnej system spełniający tego rodzaju funkcje nie podlega standardom ISO i jest organizowany w sieciach w różny sposób, pod warunkiem zgodności z ustaleniami dotyczącymi protokołów sieciowych. Może być umieszczony w określonej warstwie (np. sieciowej) lub między warstwami. Może być całkowicie rozproszony lub przynajmniej częściowo zcentralizowany. Poniżej przedstawiono pewien model takiego systemu i przebadano jego wpływ na ruch użytkowy w sieci.

2. MODEL SYSTEMU POMIAROWO-STERUJĄCEGO

Prezentowany system pomiarowo-sterujący umieszczono w warstwie sieciowej w ramach podsieci. Charakteryzuje się on częściowym zcentralizowaniem, ponieważ jeden z węzłów jest wyróżniony i spełnia dodatkowe funkcje jako Centrum Pomiarowe (CP).

^x Centrum Obliczeniowe Politechniki Wrocławskiej

Podstawowym narzędziem tego systemu są specjalne pakiety pomiarowe, których zadaniem jest przenoszenie informacji między elementami sieci. Mają one wyższy priorytet w stosunku do pakietów użytkowych, zarówno informacyjnych jak i sterujących. Wstawiane są na pierwsze miejsce do kolejki pakietów wyjściowych z węzła i nie podlegają regułom protokołu pakietowego, chociaż ich budowa jest kompatybilna z pakietami użytkowymi. W omawianym SPS wyróżnia się dwa typy pakietów pomiarowych. Pierwszy z nich to typ pakietów, w stosunku do których wszystkie węzły są jednakowo uprawnione, zarówno z punktu widzenia nadawcy jak i odbiorcy. Do takich pakietów należą: sonda, route, echo, dialog operatorski itp.

Do drugiego typu należą pakiety, które mogą być nadawane tylko przez CP. Są to pakiety cechowania czasu, zbierania danych z podsieci do dzien- nika systemu, zdalnego odczytywania lub ewentualnej zmiany pewnych obszarów pamięci, sygnalizacji sprawności itp. Większość z nich wysyłana jest przez CP periodycznie. Mechanizm ten, między innymi, stanowi gwarancję detekcji ewentualnej niesprawności CP i umożliwia inicjację i rekonfigurację sieci [4], a następnie odnowę zasobów i ponowny restart. Szczegóły dotyczące funkcji wykonywanych przez omawiany SPS, a także dotyczące typów pakietów i ich budowy można znaleźć w [3].

Dla potrzeb problemu rozważanego w artykule interesujący jest przede wszystkim model ruchu pakietów pomiarowych. Obserwujemy dwa rodzaje przepływu pakietów pomiarowych, które związane są z wyróżnionymi wyżej dwoma typami pakietów pomiarowych.

Pierwszy z nich dotyczy ruchu pakietów między dwoma dowolnymi kontaktującymi się węzłami (jednym z nich może być CP). Ruch ten charakteryzuje się następującymi cechami:

- małą intensywnością, ze względu na ograniczony zbiór pakietów, z których może składać się taki ruch; inicjacja tych pakietów odbywa się z reguły przez operatora, co w istotny sposób decyduje o ich intensywności,
- przepływem pojedynczych pakietów o niewielkiej liczbie bajtów,
- losowym momentem generacji pakietów,
- losowym charakterem pary nadawca-odbiorca pakietu, a więc i przypadkowymi trasami przepływu ruchu.

Drugi typ ruchu pomiarowego, nazywanego dalej cyklicznym, odbywa się między poszczególnymi węzłami a CP. Jest on dużo lepiej określony w stosunku do poprzedniego i dużo bardziej intensywny. Charakteryzuje się następującymi cechami:

- dużą intensywnością, ponieważ obejmuje wszystkie istniejące typy pakietów pomiarowych, z których tylko niewielka część inicjowana jest

w trybie zapytaniowym przez operatora; większość pakietów inicjowana jest automatycznie, bądź z węzłów (pakiety zdarzeniowo-informacyjne) bądź z CP,

- przepływem głównie pakietów długich, a nawet serii pakietów (w przypadku przesyłania informacji statystycznych z węzłów, zdalnej wymiany kodu itp),
- cyklicznym charakterem ruchu, na który nałożona jest losowa składowa pojedynczych pakietów,
- ścisłym określeniem drogi przepływu tych pakietów między poszczególnymi węzłami a CP.

Na podstawie powyższego zestawienia cech obu rodzajów ruchu pomiarowego można przyjąć, że ich wpływ na przepływ pakietów użytkowych nie jest jednakowy.

Ruch między równouprawnionymi węzłami jest niewielki, a możliwości sterowania nim praktycznie nie istnieją ze względu na całkowite rozproszenie decydentów tego ruchu.

Cykliczny ruch pomiarowy ma natomiast istotny wpływ na pogorszenie parametrów sieci z punktu widzenia użytkownika. Ruchem tym jednak można w znacznej mierze sterować przez dobór odpowiedniej długości cyklu. Badanie wpływu intensywności cyklicznego ruchu pomiarowego na ruch użytkowy jest tematem następnego punktu.

3. MODEL SYMULACYJNY SIECI DLA POTRZEB ANALIZY RUCHU POMIAROWEGO

Badanie wpływu intensywności ruchu pomiarowego na ruch użytkowy, biorąc pod uwagę dużą liczbę zmiennych losowych w modelu, stosunkowo najłatwiej jest zrealizować metodą symulacji. Uproszczony model sieci komputerowej, w którym uwzględniono najistotniejsze czynniki związane z celem symulacji, jest oparty na modelu przedstawionym w [1].

W modelu użytkownika zakłada się, że:

- czas pomiędzy dwoma kolejnymi zgłoszeniami użytkownika ma rozkład wykładniczy,
- zgłoszenie ma charakter serii pakietów, a liczba pakietów w serii jest liczbą losową o rozkładzie równomiernym;
- dobór węzłów jako pary nadawca-odbiorca pakietu ma charakter zmiennej losowej o rozkładzie równomiernym.

W modelu sieci przyjęto następujące uproszczenia:

- zakłada się, że sieć jest sprawna,
- jeden, z góry określony węzeł spełnia rolę CP,
- zakłada się poprawne przesyłanie pakietów między węzłami, stąd brak potwierdzeń i żądań retransmisji,

- przetwarzanie pakietu w węźle trwa określony odcinek czasu i nie zależy od długości i rodzaju pakietu,
- wybór drogi dla pakietu w sieci jest realizowany wg algorytmu stałych tras
- pojemność pamięci, buforów pośredniczących i buforów wyjściowych jest nieograniczona,
- węzły połączone są ze sobą kanałami dwukierowymi o stałej jednokierowej szybkości przesyłania (przyjęto 2,4 bity ms).

Dokładny opis modelu i jego realizacji w języku pascalopodobnym PASCAL-SIM, zorientowanym na symulację, znajduje się w 5. Intensywność zgłoszeń użytkowników ustalono eksperymentalnie. Przy określonych stałych parametrach modelu, zasymulowano pracę sieci z wyłączonym systemem pomiarowym. W ten sposób znaleziono taką średnią intensywność zgłoszeń, która powoduje, że sieć pracuje na granicy wydolności. W przykładowo symulowanej sieci tak ustalony średni czas między zgłoszeniami wynosił 100ms. Dalsze zwiększanie intensywności objawia się stałym zwiększaniem długości kolejek w węźle, a na skutek tego czas obsługi pakietów użytkowych gwałtownie rośnie. Na takie krytyczne warunki pracy sieci nałożono w dalszych eksperymentach symulacyjnych działanie SPS.

Zmienną wejściową w modelu jest intensywność ruchu cyklicznego między CP a pozostałymi węzłami. Jako parametr wyjściowy mierzono, między innymi, średni czas obsługi pakietu użytkowego w sieci.

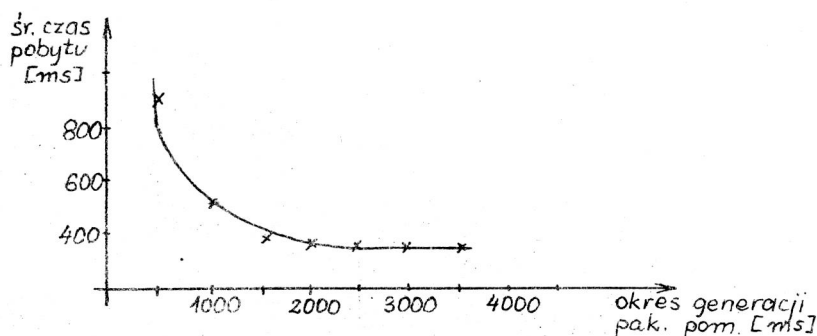
4. STEROWANIE RUCHEM POMIAROWYM W CP

Częstotliwość generacji przez CP pakietów pomiarowych istotnie wpływa na jakość systemu pomiarowego. Im mniejszy okres próbkowania tym:

- krótszy czas wykrycia uszkodzenia każdego z elementów sieci (dotyczy to również CP) i większe prawdopodobieństwo poprawnej rekonfiguracji i odnowy systemu,
- mniejsza pamięć jest wymagana w węzłach na przechowanie lokalnych danych statystycznych,
- szybsza reakcja na wszelkie dynamiczne zmiany w sieci.

Z drugiej strony duża liczba pakietów pomiarowych krążących w sieci wpływa negatywnie na parametry użytkowe systemu, które są nadrzędnym kryterium jakości sieci. Metodą symulacji zbadano jak zmienia się czas obsługi pakietów użytkowych w funkcji intensywności generacji pakietów pomiarowych. Wyniki przedstawiono na rys.1.

Zależność ta, jak widać, nie jest liniowa. Począwszy od pewnej wartości dalsze zmniejszanie intensywności ruchu pomiarowego (lub inaczej zwiększenie okresu generacji) nie wpływa w istotny sposób na obserwowane



Rys.1 Zależność czasu obsługi pakietów użytkowych od wielkości cyklu generacji pakietów pomiarowych.

Fig.1 The time of the user packets flow versus the magnitude of the cycle of the generation of the measurement packets.

parametry użytkowe. Dla symulowanej sieci punkt taki występuje przy okresie generacji pakietów pomiarowych równym około 1500ms.

Analizując krzywą otrzymaną w wyniku symulacji można dobrać maksymalną częstotliwość pracy CP przy zachowaniu minimalnego wpływu pomiarów na ruch użytkowy. W pewnym przybliżeniu jest to punkt przegięcia, w którym krzywa opadająca stabilizuje się prawie do wartości stałej.

5. ZAKOŃCZENIE

Dobór parametrów ruchu pomiarowego jest istotnym problemem, który może wpływać na ocenę pracy całego systemu pomiarowo-diagnostycznego nawet przy prawidłowym zaprojektowaniu jego struktury i mechanizmów. Dotyczy to każdego systemu pomiarowo-sterującego w sieci niekoniecznie opartego na modelu przedstawionym w artykule. W każdym przypadku projektant zmuszony jest do kompromisu między maksymalną skutecznością SPS generującego intensywny ruch pomiarowy, a dobrymi parametrami użytkowymi sieci. Należy podkreślić, że okresy wzmożonej aktywności CI przypadają najczęściej w czasie intensywnego ruchu użytkowego i przy przeciążonej sieci. Przedstawiona powyżej metoda doboru parametrów pracy SPS przy pomocy symulacji jest jednym ze sposobów rozwiązania tego problemu.

6. LITERATURA

- [1] "Wprowadzenie do informatyki" Praca zbiorowa pod red. W. Sobczaka, W-wa, WKiŁ 1979
- [2] Butrymienko A.W.: "Projektowanie i eksploatacja sieci komputerowej", W-wa, MNT 1983
- [3] Bicz T., Bieleninik E., Kosmulska-Bochenek E., Misiura E. "System pomiarowo-sterujący podsieci komunikacyjnej MSK", Centrum Obliczeniowe

Pol.Wrocł.Raport SPR 29/85

- [4] Hudyma E., Kosmulska-Bochenek E.: "The Structure of the Reliable Control and Measurement System in the Computer Network" Proceedings of the IFIP TC 6 Working Conference COMNET'85, Budapest 1985
- [5] Stasiak J., Szala E.: "Sterowanie przepływem danych pomiarowo-sterujących w MSK", praca dyplomowa, Wrocław 1985.

ESTIMATION OF THE CONTROL AND DIAGNOSTIC FLOW IN COMPUTER NETWORK

The subject of the paper is an analyze of a flow of the control and measurement system of the computer communication network. There is assumed a model of the system under consideration and simulation method is used for an estimation of parameters of the flow.

ИСПЫТАНИЕ ИЗМЕРИТЕЛЬНО-УПРАВЛЯЮЩЕГО ПОТОКА В ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

В статье представлен анализ измерительно-диагностического движения для определённой модели измерительно-управляющей системы вычислительной сети. Определяется влияние этого движения на один из эксплуатационных параметров сети, т.е. время обслуживания прикладных пакетов. Для оценки параметров измерительного движения используются методы моделирования.

sieć komputerowa,
transfer zbiorów,
usługi, protokoły

Andrzej KALIŚ[✱], Roman KASZUBA[✱], Jan KWIATKOWSKI[✱]

IMPLEMENTACJA TRANSFERU, DOSTĘPU I ZARZĄDZANIA ZBIORAMI W MODELU SYSTEMU OTWARTEGO

Celem pracy jest przedstawienie założeń implementacyjnych, które były przyjmowane przez grupy wykonawców realizujących stacje FTAM (File Transfer Access and Management). Uzasadniono potrzebę tworzenia założeń oraz wspólnych ustaleń przez grupy wykonawców. Materiał zawarty w pracy dotyczy ustaleń poczynionych głównie przez specjalistów amerykańskich dla FTAM. Bazują one na drugim wydaniu standardu ISO8571.

1. WPROWADZENIE

Zespoły wykonawców realizujących oprogramowanie rozległych sieci komputerowych bazują na dokumentach międzynarodowych organizacji standaryzacyjnych takich jak ISO i CCITT. Dokumenty standaryzacyjne definiują usługi i protokoły każdej z siedmiu warstw systemu otwartego ISO/OSI. Forma dokumentów jest opisowa, chociaż zauważa się szereg prac próbujących sformalizować specyfikację usług i protokołów.

Praktyka i doświadczenie uzyskane przy projektowaniu i implementacji sieci MSK oraz doniesienia o kłopotach zagranicznych wykonawców komponentów składowych rozległych sieci komputerowych wskazują, że stacje warstwowe wykonane przez różne zespoły wykonawców, których projekty bazowały na identycznych dokumentach standaryzacyjnych nie współpracują ze sobą.

Budując sieć MSK oprogramowywano węzły pakietowej podsieci transmisji danych na dwóch minikomputerach - SM-3 oraz MERA-60. Wewnętrzny dokument "standaryzacyjny" opracował zespół Centrum Obliczeniowego Politechniki Wrocławskiej, który jednocześnie implementował oprogramowanie na SM-3. Merę-60 oprogramowywał zespół z Politechniki Śląskiej, który także opierał swoje prace na wspomnianym wyżej dokumencie. Po uruchomieniu oprogramowania na obu minikomputerach okazało się, że węzły ze sobą nie-

[✱]Centrum Obliczeniowe Politechniki Wrocławskiej

współpracują. Aby jednak wywiązać się z podjętych zobowiązań, przywieziono MERĘ-60 do Wrocławia i tutaj dokończono uruchamiania. Przyczyną niepowodzenia były dwuznaczności w interpretacji dokumentu bazowego.

Analogiczna niezgodność wystąpiła przy uruchamianiu stacji transferu zbiorów w sieci MSK, jednej implementowanej na Odrze 1305 i drugiej na EC1032. Obie stacje realizowały dwa odrębne zespoły, opierające się w swych pracach na jednym dokumencie.

Stąd, pozostaje nadal aktualne pytanie - Co robić, by ustrzec się różnej interpretacji tych samych dokumentów standaryzacyjnych? Nasuwa się natychmiastowa odpowiedź - przekazać wykonawstwo stacji warstwowych tego samego poziomu, jednemu zespołowi wykonawców. W tym przypadku partnerskie stacje będą ze sobą współpracować. Jednak pomysł ten jest mało realny. Po pierwsze, nie ma zespołu, który zna wszystkie liczące się na rynku komputery, by móc implementować na nich swoje oprogramowanie. Po drugie, każda firma komputerowa pragnie by jej sprzęt był włączony do rozległych sieci komputerowych, czyli każda z nich będzie się starać implementować poszczególne usługi nie czekając na zakończenie prac nad standardami.

Generalnym rozwiązaniem problemu dwuznacznego rozumienia dokumentów standaryzacyjnych jest formalna ich specyfikacja (LOTOS, ESTELLE, SDL). Zmusza ona, w przeciwieństwie do tekstu w języku naturalnym, do dokładnego zapoznania się ze specyfikacją.

W tym momencie nasuwa się pytanie - Czy należy czekać na poziom IS każdego ze standardów? Zauważyć można, że odpowiedź głównych wykonawców oprogramowania rozległych sieci komputerowych w świecie jest następująca - nie czekać! Ale i nie ryzykować. Wiadomo, że produkcja oprogramowania stanowi największą część składową kosztów poniesionych na wytworzenie systemu komputerowego. Otóż, daje się zauważyć, że producenci sprzętu komputerowego starają się wypracować wspólne ustalenia implementacyjne, które bazują na dokumentach standaryzacyjnych zazwyczaj jest to najnowsze wydanie poziomu DP standardu, umawiając się co do szczegółów tej części dokumentu, którą uznają za "ustabilizowaną", tzn. która nie ulegnie zmianie lub ewentualnej niewielkiej modyfikacji, gdy dokument osiągnie poziom IS poprzedzony przez DIS.

Przykładem takiej umowy jest prezentowany w niniejszej pracy zbiór założeń i Uściśleń dotyczących usług i protokołu transferu, dostępu do i zarządzania zbiorami w systemach otwartych (FTAM), opracowany na podstawie [8].

2. SPECYFIKACJA IMPLEMENTACJI

2.1 Usługi

Jako obowiązkowe przyjęto następujące jednostki usługowe: P_INITIA-

LIZE, F_SELECT, F_OPEN, F_READ, F_WRITE, F_DATA_END, F_TRANSFER_END, F_CLOSE, F_DESELECT, F_U_ABORT, F_P_ABORT, F_TERMINATE. Implementacja F_CANCEL jest opcjonalna. Przyjęto, że równocześnie implementowana jest F_READ i F_WRITE, w czasie trwania wyboru zbioru; zachodzi przynajmniej raz F_OPEN i co najmniej raz F_READ lub F_WRITE; w okresie połączenia możemy dokonać więcej niż jednego wyboru zbioru ale kolejno po sobie; połączenie sesyjne jest zakładane na początku działania FTAM i rozłączane, gdy połączenie transferu jest rozłączane; całkowicie implementowany zespół funkcjonalny ograniczonego zsrządzania zbiorami, korygowany poziom usług oraz atrybuty podzbioru pamięci.

2.2 Atrybuty zbiorów

Implementacje będą negocjować następujące atrybuty zbiorów: nazwę zbioru, typ struktury dostępu, nazwę kontekstu prezentacji, bieżący rozmiar zbioru, dostęp, bieżący typ struktury dostępu, bieżący kontekst prezentacji, przewidywany rozmiar zbioru oraz datę i czas utworzenia zbioru. Uzgodniono, że nie ma ograniczenia na maksymalną długość nazwy zbioru; system, w którym nie mogą być przetwarzane rozszerzone nazwy zbioru, powinien odrzucić (select, create, delete) żądanie skierowane do takiego zbioru; nazwa zbioru składa się z dużych liter oraz cyfr, ale pierwszym znakiem nazwy nie mogą być cyfry.

Implementowana będzie jedynie bezpostaciowa struktura dostępu tzn. struktury jednowymiarowe i hierarchiczne nie będą na razie brane pod uwagę.

Przyjęto implementować dwa konteksty prezentacji: tekstowy formatowany oraz bitowo zorientowany. W pierwszym każda linia tekstu jest zmiennej długości i zakończona parą znaków sterujących CRLF; długość ograniczono do 250 znaków plus CRLF; CR i LF nie mogą występować przedzielone innymi znakami; wykorzystywany repertuar znaków definiuje ISO646 [5].

Obowiązkowymi wartościami atrybutu dostęp są: read, replace, read attribute oraz delete file.

Bieżący kontekst prezentacji jest identyczny z PRESENTATION CONTEXT tzn. ASCII lub bitowo zorientowany. Pozostałe atrybuty zbioru są implementowane zgodnie z [3].

2.3 Ustalenia szczegółowe

- implementacja F_CANCEL w przypadku nienaprawialnych błędów zespołu czytania/pisania jest opcjonalna. Stacja otrzymująca F_CANCEL może odpowiedzieć F_ABORT.
- nie implementuje się zespołu funkcjonalnego grupowania.

- wartość poziomu usługi w F_INITIALIZE ustawia się na poziom korygowany
Klasa usług - klasa transferu; zespół funkcjonalny - czytanie, pisanie
ograniczone zarządzanie; podzbiory atrybutów - podzbiór pamięci.
- wartość parametru "tryb przetwarzania" w F_OPEN może być read lub re-
place; kontekst prezentacji ustawiony, wskazujący kontekst transferu
(ustalenie to przyjęto, ze względu na założenie braku implementacji
warstwy prezentacji).
- dekodowanie F_DATA jest następujące:
IDENTIFIER: wartość specyficznego znacznika kontekstu [55],
LENGTH : (długość w oktetach),
CONTENTS : (oktety).

Definicja F_DATA wykorzystuje notację ASN.1 :

F_DATArequest ::= data [55] IMPLICIT OCTETSTRING

- syntaktyka abstrakcyjna wykorzystuje typ danych "ISO646STRING".
- atrybut kontekst prezentacji stosowany w żądaniu SELECT/CREATE ogra-
niczony jest do sekwencji definiującej tylko jedną nazwę kontekstu
prezentacji. Dekodowanie nazwy kontekstu prezentacji następuje zgodnie
z regułą:
PresentationContextName ::= Application [13] PrintableString .
- nazwa kontekstu prezentacji w F_OPENrequest i F_OPENresponse jest obo-
wiązkowa (w standardzie opcjonalna).
- ustalono syntaktykę abstrakcyjną F_INITIALIZE, nie przytaczamy jej ze
względu na ograniczoną ilość miejsca .
- zgodzono się, że wszyscy implementujący stacje FTAM zgodnie z tablicą
stanów podaną w [4] uwzględniając to co podano wyżej .

2.4 Dalsze uzgodnienia szczegółowe

Przytoczone niżej rekomendacje implementacyjne nie są specyfikowane
w dokumencie standaryzacyjnym. Przyjęto je jednak, aby być pewnym, że
różne implementacje FTAM będą ze sobą współpracować:

- przyjęto, że wszystkie implementacje FTAM będą bezpośrednio odwzorowa-
ne w usługi sesyjne i nie będą wykorzystywane w tym czasie usługi CASE
[6,7]. F_INITIALIZerequest jest osadzony w S_CONNECTrequest oraz
F_INITIALIZEResponse w S_CONNECTresponse, czyli nie jest on wysyłany
jako S_DATArequest po ustanowieniu sesji. Jeżeli F_INITIALIZEResponse
jest pozytywnie potwierdzone, to F_INITIALIZEResponse odwzorowane jest
całkowicie w S_ACCEPT. W przypadku przeciwnym na S_REFUSE.
- F_U_ABORT i F_P_ABORT są odwzorowane w S_DATArequest. Odbiorca F_U_
ABORT i F_P_ABORT musi wysłać S_U_ABORTrequest .
- każda implementacja powinna być zdolna do rozbiórki gramatycznego
wszystkich, poprawnych parametrów opcjonalnych [3,4] , jeżeli parametry

te występują w jednostce danych protokołu, jednak tylko te parametry, które wymienia się w umowie, są brane pod uwagę przy żądaniach i odpowiedziach. Niektóre parametry opcjonalne zamieniono na obowiązkowe. Jeżeli nie występują one w jednostce danych protokołu, żądanie nie może być odrzucone.

- umowa nie ogranicza działań podejmowanych w sytuacjach błędu i diagnostyce, a zaproponowanych w ISO8571. Implementacje mogą zwrócić w odpowiedzi "negatywne" wartości diagnostyczne lub wysłać F_U_ABORTrequest.
- po błędzie, tzn. po F_U_ABORT lub F_P_ABORT, zaleca się pozostawić zbiór w stanie, który umożliwia kolejny dostęp do niego zarówno z oddalonego jak i z lokalnego systemu.
- nie umówiono się odnośnie reguł rządzących współużywalnością zbiorów (atrybuty nie są implementowane). Postanowiono problemy współużywalności pozostawić lokalnym systemom do rozwiązania. Jednak by zmniejszyć dodatkowe źródło błędów zalecono stosowanie następujących reguł implementacyjnych:
 - a/ zbiór może być współużywalny przez kilka transferów jednocześnie jedynie, gdy dostęp do niego posiada tryb czytania,
 - b/ jeżeli zbiór udostępniono w trybie pisania, to dowolne żądanie dostępu do zbioru powinno być odrzucone,
 - c/ zbiór nie może być modyfikowany przez użytkowników lokalnego systemu podczas trwania transferu.
- reguły dekodowania nagłówków jednostek danych protokołu będą zgodne z syntaktyką ISO/ASN.1 (jednak przyjęto, że maksymalna długość jednostki protokołu nie przekroczy 1024 oktetów).
- parametr DIAGNOSTIC zdefiniowano jako ImplicitSequence.
- jeżeli operator otrzyma S_CONNECT, a dane użytkownika przenoszone w tej jednostce wskazują na niepoprawialny błąd, wtedy:
 - a/ jeżeli dane użytkownika rozpoznano jako F_INITIALIZerequest, operator powinien odpowiedzieć F_INITIALIZEResponse - lub F_P_ABORT z parametrem diagnostycznym wskazującym na nieporawialny błąd. Jednostki te będą przenoszone w S_CONNECTresponse - . Parametry w F_INITIALIZEResponse powinny być zwierciadlanym odbiciem parametrów F_INITIALIZerequest; jeżeli nie jest to możliwe, operator ustawia wartości zastępcze.
 - b/ jeżeli dane użytkownika nie mogą być rozpoznane jako F_INITIALIZerequest, wtedy operator powinien odpowiedzieć S_U_ABORT z kodem odpowiadającym przyczynie błędu.

3. PODSUMOWANIE

Przedstawione w pracy uzgodnienia poczynione pomiędzy użytkownikami

różnych systemów komputerowych i zauważalny pośpiech w implementacji przede wszystkim usługi zbiorów w sieciach komputerowych wskazuje, że istnieje duże zapotrzebowanie na tego typu usługi, które zmusiły producentów do prac, pomimo braku ustabilizowanych dokumentów standaryzacyjnych. Producenci starają się oprzeć swoje wyprzedzające prace na najnowszych dokumentach ISO, uzgadniając szczegóły implementacyjne w taki sposób by "okrojone" implementacje mogły ze sobą współpracować; kolejne nadzbiory uzgodnień będą również ustalane. Przyjęcie takiej strategii rozwojowej zapewni im gotowe produkty w momencie niezbyt oddalonym od chwili ukazania się pełnego dokumentu poziomu IS.

Można potraktować przedstawione w pracy uściślenia jako dobry punkt wyjścia do dyskusji nad implementacją usług zbiorów w polskich rozległych sieciach komputerowych.

LITERATURA

- [1] ISO DIS8571/1, TC97/SC21 N-2372, AUG. 1986,
- [2] ISO DIS8571/2, TC97/SC21 N-2373, AUG. 1986,
- [3] ISO DIS8571/3, TC97/SC21 N-2374, AUG. 1986,
- [4] ISO DIS8571/4, TC97/SC21 N-2375, AUG. 1986,
- [5] 7-bit Coded Character Set for Information Processing Interchange, ISO646, 1973,
- [6] ISO DIS8649/2, TC97/SC21 N-1493, May 1986,
- [7] ISO DIS8650/2, TC97/SC21 N-1494, May 1986,
- [8] NBSIR 86-3385-1, Implementation Agreements for Open Systems Interconnection Protocols, NBS, Revised July 24, 1986

FILE TRANSFER, ACCESS and MANAGEMENT IMPLEMENTATION FOR OPEN SYSTEMS INTERCONNECTION

The paper deals with implementation agreements for Open Systems Interconnection undertaken by the File Transfer, Access and Management implementor groups. The need for such an agreement is justified. Presented agreements are made by american experts for the second edition of the ISO/OSI FTAM standard.

РЕАЛИЗАЦИЯ ПЕРЕДАЧИ ФАЙЛОВ ДОСТУПА И УПРАВЛЕНИЯ ФАЙЛОВ В МОДЕЛИ ОТКРЫТЫХ СИСТЕМ

В статье представлены основы реализации станции FTAM, которые были принимаемые группами исполнителей. Обосновано необходимость создания основ и общих установлений в группе исполнителей. Материал статьи главным образом относится к установлениям сделанным американскими специалистами для FTAM. Базируют они на втором издании стандарта ISO 8571.

Andrzej KALIŚ^x
Roman KASZUBA^x
Jan KWIATKOWSKI^x

sieć komputerowa,
usługi sieciowe,
transfer zbiorów

USŁUGA TRANSFERU ZBIORÓW W SIECI KOMPUTEROWEJ

Zakres usług informatycznych sieci komputerowej wyznaczony jest przez sumę usług świadczonych przez poszczególne systemy komputerowe wchodzące w jej skład oraz możliwości łącznego wykorzystania usług różnych komputerów. W pracy skoncentrowano się na usłudze transferu zbiorów, umożliwiającą przenoszenie informacji w postaci zbiorów pomiędzy komputerami obliczeniowymi sieci. Rozważono zagadnienia mające wpływ na niezawodność oraz "szybkość" transferu. Omówiono sposób dostępu do usług.

1. WPROWADZENIE

Sieć komputerowa łącząc wiele systemów komputerowych umożliwia użytkownikowi dostęp do usług świadczonych przez każdy z komputerów obliczeniowych (KO) sieci oraz do tzw. usług sieciowych, w których realizacji uczestniczy co najmniej dwa KO. Jedną z usług sieciowych jest usługa transferu zbiorów umożliwiająca przenoszenie zbiorów pomiędzy KO sieci. Dla jej realizacji wymagana jest współpraca dwóch programów systemowych, tzw. Stacji Transferu Zbiorów (STZ), zlokalizowanych w różnych KO sieci. Programy te w imieniu zleceniodawcy użytkownika usług realizują usługę transferu zbioru, wykorzystując w tym celu komendy protokołu transferu zbiorów. Łącznie programy STZ zlokalizowane w każdym z KO sieci tworzą tzw. System Transferu Zbiorów, dostarczający usługi wszystkim użytkownikom sieci. W pracy przedstawiono wybrane zagadnienia związane z implementacją usługi, które zdaniem autorów mają największy wpływ na jej jakość [1,2].

2. PRZEBIEG TRANSFERU

Jak wspomniano powyżej dla realizacji usługi transferu zbiorów wymagana jest współpraca dwóch STZ zlokalizowanych w różnych KO sieci. Jedną ze stacji, lokalna względem użytkownika wydającego polecenie tran-

^x Centrum Obliczeniowe Politechniki Wrocławskiej

sferu jest odpowiedzialna za jego przebieg i nazywana jest kontrolerem, druga ze stacji uczestnicząca w transferze wykonuje tylko polecenia kontrolera i nazywana jest operatorem [3].

Przebieg każdego transferu można podzielić na pięć faz:

- fazę nawiązywania połączenia, w której nawiązywane jest połączenie między współpracującymi STZ z wykorzystaniem usług warstw niższych,
- fazę negocjacji, w której ustalony jest sposób prowadzenia transferu (negocjowane są parametry zbiorów, parametry transmisji, itp),
- fazę transferu danych, w której odbywa się transmisja według zasad uzgodnionych w fazie negocjacji,
- fazę zakończenia transferu, w której uzgadniany jest status zakończenia transferu,
- fazę rozłączenia połączenia, w której rozłączane jest połączenie między współpracującymi STZ.

W zależności od przyjętego rozwiązania implementacyjnego, niektóre z faz mogą zachodzić na siebie lub może być ich brak dla konkretnego transferu. Jednakże w każdym transferze występują trzy środkowe fazy. Sposób prowadzenia transferu, w szczególności fazy negocjacji uzależniony jest od liczby zaimplementowanych warstw modelu odniesienia [5]. Pierwsze cztery warstwy są praktycznie implementowane we wszystkich sieciach, natomiast funkcje pozostałych trzech warstw niejednokrotnie są spełniane przez jedną warstwę (warstwę aplikacji). Powoduje to konieczność negocjowania pewnych warunków przeprowadzania transferu, które normalnie są ustalane przez warstwy sesji oraz prezentacji (np. kod transferu, czas oczekania). Jednakże bez względu na liczbę zaimplementowanych warstw, można przeprowadzić identyczną klasyfikację usług transferu ze względu na jakość oferowanej usługi. Dla użytkownika najistotniejszymi cechami usługi jest jej niezawodność, czas realizacji oraz wygoda w posługiwaniu się usługą, i w tym też kierunku powinny zmierzać podstawowe działania implementatorów usługi transferu zbiorów.

3. CZAS REALIZACJI TRANSFERU

Poza czynnikami czysto fizycznymi, takimi jak przepustowość podsięci transmisji danych, szybkość obliczeniowa systemu, itp. na czas realizacji transferu mają wpływ sposób nawiązywania połączenia [4] oraz przyjęta strategia planowania kolejnością transferu.

W pierwszym przypadku zasadnicze pytanie brzmi: czy używać komend protokołów niższego poziomu dla przenoszenia danych poziomu wyższego (tzw. zagnieżdżane). Przyjęcie takiego rozwiązania jest niebezpieczne z co najmniej dwóch powodów. Po pierwsze powoduje ono wydłużenie komend niższego poziomu, a po drugie stwarza problemy w przypadku ponownego

wykorzystania tego samego połączenia dla realizacji innego transferu lub w przypadku multipleksowania połączenia poziomu niższego. Jednakże w przypadku, gdy nie występuje multipleksowanie oraz w komendzie poziomu niższego przekazywana jest tylko część parametrów, to korzyści ze stosowania wymienionej techniki mogą być duże nawet w przypadku próby przeprowadzenia innego transferu na tym samym połączeniu. Zmniejsza się liczba wymaganych interakcji pomiędzy współpracującymi systemami oraz do ogólnego czasu trwania transferu nie jest wliczany czas ustanawiania połączenia. Oczywiście jest w tym przypadku konieczność oczekiwania z zaakceptowaniem połączenia niższego poziomu do momentu zaakceptowania połączenia na poziomie wyższym. Dlatego też parametry zagnieżdżone w komendzie niższego poziomu powinny z dużym prawdopodobieństwem stanowić o pozytywnym zakończeniu późniejszej negocjacji na poziomie transferu zbiorów.

Drugim wymienionym elementem mającym wpływ na czas realizacji transferu jest odpowiednio zestrojony system planowania transferami. Pomijając zagadnienia związane z długością przesyłanego zbioru, można przyjąć, że poprawny system planowania powinien spełniać następujące kryteria:

- brak monopolizacji kierunku transferu,
- brak monopolizacji typu realizowanych zleceń lokalne, zdalne,
- brak monopolizacji połączeń transportowych.

Przyjęcie powyższych założeń prowadzi do bardzo prostego algorytmu planowania transferami, który zapewnia:

- wykonanie na jednym połączeniu dwóch transferów w przeciwnych kierunkach,
- wykonanie na jednym połączeniu jednego transferu będącego wynikiem polecenia lokalnego i jednego zdalnego.

Po wykonaniu takiej pary transferów, następuje rozłączenie połączenia i pobranie następnego transferu do realizacji zgodnie z warunkiem braku monopolizacji połączenia transportowego. Z drugiej strony w celu minimalizacji liczby nieefektywnych wywołań zdalnych STZ z sieci, należy przyznać priorytet transferom zdalnym; niestety może to spowodować monopolizację typu wykonywanych zleceń transferu. Pewnym rozwiązaniem w tym przypadku jest stosowanie komendy "Przejmij funkcję kontrolera", która daje możliwość realizacji transferu zdalnego. Technika ta może być stosowana tylko w przypadku monitorowania nieefektywnych wywołań STZ z sieci. Minimalizacja liczby wywołań z sieci umożliwia również stosowanie tzw. zbioru informacyjnego transferu zawierającego rekordy zapisu stanu transferu. W rekordach tych zapisywane są informacje dotyczące aktualnego stanu transferu jak położenie ostatnio przyjętego znacznika, liczba

przyjętych znaczników, jak również informacje określające powód odrzucenia wywołania zdalnej STZ. Bazując na tych informacjach, system planowania może wybierać do realizacji transfery, dla których próba nawiązania połączenia powinna zakończyć się pozytywnie.

Reasumując, stosowanie zagnieźdżenia komend oraz odpowiednio zestrojony system planowania, poprzez zmniejszenie liczby interakcji pomiędzy systemami oraz minimalizację liczby nieefektywnych wywołań, skracca czas realizacji transferu (w zależności od sytuacji średni lub maksymalny).

4. NIEZAWODNOŚĆ TRANSFERU

Transfer zbioru w sieci komputerowej odbywa się w środowisku zawodnym, możliwe są awarie komputerów biorących udział w transferze jak i przekłamania w informacji przesyłanej. Jednakże protokoły transferu zbiorów nie zawierają mechanizmów umożliwiających wykrycie a następnie korekcie błędów, dlatego też jest to realizowane przez warstwy niższe bądź też błędna sytuacja jest raportowana do stacji aplikacyjnych (poza zniekształceniami zbiorów wynikającymi z różnych zestawów znaków graficznych systemów biorących udział w transferze). Dlatego też, na poziomie STZ wymagane są mechanizmy umożliwiające kontynuację przerwanych bądź błędnych transferów. Błędy transferu mogą występować w każdej z fazy jego realizacji. W zależności od fazy transferu wymagane są inne mechanizmy umożliwiające jego kontynuację. I tak w przypadku błędów w fazie nawiązywania połączenia wymagane jest powtórzenie transferu od początku. W przypadku błędów w fazie negocjacji wymagane jest następujące postępowanie - wznowienie transferu od początku i w przypadku odrzucenia wznowienia, jego powtórzenie. Błędy w fazie transferu danych wymagają wznowienia transferu od ostatniego poprawnie odebranego znacznika informacja zapisana w rekordzie zapisu stanu transferu. Dlatego też w strumieniu przesyłanych danych należy umieszczać znaczniki oraz wymagana jest możliwość negocjacji punktu startu wznowianego transferu. Alternatywnym rozwiązaniem jest powtórzenie transferu od początku, jednakże w tym przypadku czas realizacji transferu ulega znacznemu wydłużeniu. Błędy w fazie zakończenia transferu mogą doprowadzić do błędnej interpretacji statusu zakończenia transferu. Dlatego też jest wymagane wznowienie transferu w celu wynegocjowania poprawnego statusu zakończenia. W przypadku negatywnej odpowiedzi na wznowienie należy przyjąć, że ostatnio nadany status transferu jest ważny. Wynika to z faktu, że status zakończenia transferu zostaje praktycznie ustalony w fazie transferu danych poprzez wymianę odpowiednich komend pomiędzy odbiorcą i nadawcą. Stąd

ponowne wysłanie komendy kończącej transfer wraz z poprzednią wymianą stanowi podwójne potwierdzenie ustalonego statusu zakończenia. Fakt powyższy ma szczególne znaczenie, gdy usługa transferu zbiorów jest wykonywana przez inne stacje aplikacyjne, np. system transferu zadań, gdzie zagadnienia niezawodnościowe są bardzo ważne. W przypadku braku powyższych mechanizmów praktycznie naprawialne mogą być tylko błędy występujące w fazie nawiązywania połączenia oraz fazy zakończenia (pomijając mechanizm powtórzenia transferu od początku, który może być stosowany przy braku wznowień).

Odmienne przedstawia się sytuacja w przypadku wystąpienia przekłamania informacji w trakcie transferu danych bądź częściowe jej zagubienie. Przydatny wtedy staje się mechanizm powtórek umożliwiający ponowne przesłanie tej samej informacji w ramach tego samego transferu. Należy zauważyć, że zaimplementowanie tylko mechanizmu powtórzeń jest wystarczające w celu poprawnego przeprowadzenia każdego transferu, jednakże jest to możliwe tylko w przypadku, gdy implementacja dostępu do zbiorów umożliwia przywrócenie stanu początkowego w przypadku błędu. Zwiększa to jednakże czas realizacji transferu.

5. JĘZYK UŻYTKOWNIKA

Ze względu na wygodę użytkownika System Transferu Zbiorów powinien być systemem transakcyjnym, tzn. systemem, który polecenie użytkownika przyjmuje jako zlecenie, wykonywane niezależnie od dalszej działalności użytkownika. Jednocześnie sposób wydawania poleceń do systemu powinien być jak najprostszy. Najlepszym rozwiązaniem jest włączenie komunikacji użytkownika z systemem do Języka Opisu Zadań JOZ każdego z lokalnych systemów operacyjnych. Jest to konstrukcyjnie i użytkowo wygodne rozwiązanie, gdyż pozwala ono użytkownikowi na przekazywanie dyrektyw do systemu za pomocą komend JOZ oraz otrzymywania informacji od systemu w sposób analogiczny do innych informacji otrzymywanych od systemu operacyjnego. Komunikacja użytkownika z systemem może się odbywać poprzez zbiory; komendy systemu STZ dopisują lub usuwają komunikaty ze zbiorów komunikacyjnych zwanych "skrzynkami pocztowymi", a więc nie obowiązuje żadna synchronizacja działań użytkownika z działalnością systemu STZ. Oznacza to, że komendy systemu STZ mogą być wydawane w dowolnym czasie, niezależnie od siebie, niezależnie od wykonywanych transferów i niezależnie od czasów eksploatacji systemu STZ. Z drugiej strony użytkownik powinien mieć przez cały czas możliwość monitorowania wydanych poleceń, poprzez komendy umożliwiające uzyskanie informacji o stanie zaawansowania wykonywanej usługi oraz możliwości zaniechania jej wykonywania. Powyższe powinno być możliwe niezależnie od trybu pracy użytkownika (konwersa-

cyjnie lub wsadowo).

6. PODSUMOWANIE

Omawiana w pracy usługa transferu zbiorów umożliwia użytkownikowi:

- przenoszenie zbiorów między różnymi KO,
- zaniechanie wykonywania wydanego polecenia,
- uzyskanie informacji o stanie zaawansowania wykonywanej usługi.

W pracy przedstawiono najistotniejsze zdaniem autorów problemy występujące w implementacji omawianej usługi, wpływające na jej "jakość". Pracę należy traktować jako propozycję, a jednocześnie głos w dyskusji, jak powinna wyglądać poprawna implementacja usługi transferu zbiorów.

LITERATURA

- [1] Dubielewicz A. i inni, Stacja Transferu Zbiorów sieci SKJS2 w. 1, część 1, Dokumentacja programowa, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 22/86, 1986,
- [2] Jacukowicz M. i inni, Opis stacji transferu zbiorów i zadań dla komputera Odra 1305, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 03/84, 1984,
- [3] Kaszuba R., Protokół transferu zbiorów dla MSK, Centrum Obliczeniowe Politechniki Wrocławskiej, Raport SPR 26/84, 1984,
- [4] Kaszuba R., Kwiatkowski J., Experience in File and Job Service Implementation Using the British "Coloured Books" Protocols, Network in Office Automation, International Symposium IFIP '84, str. 50-61, Sofia 1984,
- [5] ISO 7498, Information Processing Systems - Open Systems Interconnection, Basic Reference Manual.

FILE TRANSFER SERVICE IN COMPUTER NETWORK

Computer networks provide information processing services composed of services offered by individual host computers, when operating in stand-alone mode, and services created by the possibility of using more than one computer to perform a composed task. The paper is focused on file transfer services, which user-friendly tool for information transfer between computers. Issues on transfer reliability and performance are considered. User interface to the services is described.

СЕРВИС ПЕРЕНОСА НАБОРОВ ДАННЫХ В ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

Диапазон услуг вычислительной сети определяется суммой услуг предоставляемых отдельными вычислительными системами из которых она состоит, а также возможностями совместного использования услуг различных ЭВМ. В работе рассматривается сервис переноса информации в виде наборов данных между вычислительными машинами сети. Обсуждаются проблемы влияющие на надежность и "скорость" переноса. Описывается исход доступа к сервису.

Grażyna KATANA^{*}, Leszek KOTULSKI^{**}
Andrzej SOBALA^{***}

lokalna sieć komputerowa,
system COSID,
alokacja oprogramowania

MIEJSCE ALOKACJI
W SYSTEMIE GENERACJI OPROGRAMOWANIA WSPÓLBIEŻNEGO
DLA LOKALNYCH SIECI KOMPUTEROWYCH^{***}

Praca dotyczy metodologii oprogramowywania systemów rozproszonych. Proponuje określenie i wyróżnienie poziomu oprogramowania, na którym można by operować jednolitą terminologią niezależną od rozwiązań sprzętowych, a który służyłby do manipulowania komponentami składającymi się na oprogramowanie np. lokalnej sieci komputerowej. Podejście takie umożliwiłoby budowę i użytkowanie oprogramowania systemów rozproszonych w sposób modularny. Operowanie wcześniej przygotowanymi modułami-komponentami byłoby, natomiast zadaniem specjalnego programu zwanego "alokatorem".

Rozwój lokalnych sieci komputerowych stworzył zapotrzebowanie na opracowanie specjalnych narzędzi programistycznych użytecznych w pracach związanych z oprogramowaniem systemów rozproszonych. Uogólniając można powiedzieć, że należy znaleźć poziom, na którym możliwe byłoby w miarę jednolite widzenie i obsługa sytuacji związanych z różnorodnym sprzętem komputerowym. Na takim poziomie działałby programista systemowy decydujący o ostatecznym kształcie warstwy software'owej udostępnionej użytkownikowi wraz ze sprzętem.

W odpowiedzi na to zapotrzebowanie zrodził się pomysł skonstruowania systemu software'owego COSID (COncurrent Software Implementation for Distributed systems) wspomagającego implementację oprogramowania w lokalnych sieciach komputerowych [4]. Ma on służyć do wyprodukowania (kompilator) i manipulowania (alokator) komponentami, z których programista systemowy mógłby konstruować system oferowany konkretnym użytkownikom, a także, w miarę potrzeby modyfikować go.

^{*} Katedra Informatyki Uniwersytetu Jagiellońskiego, Kraków.

^{**} Instytut Fizyki Jądrowej, Kraków.

^{***} Praca wykonana w ramach programu badawczego Min.SzWNiP RP.I.09 "Rozwój języków, metod i podstaw formalnych programowania".

W niniejszym referacie zasygnalizowane zostaną rozwiązania sferowane przez alokator w systemie COSID. Ten, pewnego rodzaju program organizacyjny, ma do spełnienia dwa zadania. Pierwsze z nich to pierwotne rozmieszczenie w systemie komponentów, tak, aby stworzyły pożądaną strukturę. Ponieważ lokalna sieć komputerowa złożona jest z kilku (czasami różnorodnych) maszyn cyfrowych, które kolejno włączane są do pracy, to działający w czasie fizycznego uruchamiania sprzętu alokator winien umożliwić sukcesywne uruchamianie ich, czuwając nad organizacją powiązań w całym systemie. Drugie zadanie, funkcjonalnie analogiczne do pierwszego, to manipulacja komponentami w trakcie eksperymentów związanych z przygotowywaniem ostatecznej postaci części software'owej systemu, jaki ma być oferowany użytkownikowi. Należy tu uwzględnić zarówno proste przenoszenie komponentów pomiędzy węzłami, jak i ewentualne ich usuwanie spowodowane wyłączeniem z pracy całych węzłów, bądź poszczególnych urządzeń zarówno w normalnym trybie pracy (zakończenie pracy przez jednego współużytkownika sieci) jak i awaryjnym [5]. Podobne zadania stawia zresztą modyfikacja czy rozbudowa sieci.

Alokator musi widzieć sieć jako jeden system programowo-sprzętowy, na którego stanie dokonuje operacji. Konieczność realizacji tych zadań wymaga zdalnej alokacji t. j. możliwości oddziaływania z jednego węzła sieci na przebieg alokacji w innym węźle. Dla poprawnego działania systemu COSID oraz programu współbieżnego wytworzonego przy jego pomocy potrzebne jest istnienie oprogramowania podstawowego lokalnej sieci komputerowej. Musi ona logicznie odpowiadać zadaniom systemu transportowego według standardu ISO OSI Reference Model [2], który organizuje przesyłanie komunikatów pomiędzy różnymi komponentami. Wybór konfiguracji sieci nie ma więc wpływu na logiczną definicję alokatora, będzie natomiast go mieć na efektywność pracy. Ze względu na konieczność realizacji funkcji rozgłaszania zalecana jest konfiguracja Ethernet.

Jednym z najbardziej istotnych pytań dotyczących systemu alokacji jest pytanie "Który z węzłów i w jakim zakresie może alokować komponenty systemu?". System COSID pozwala zróżnicować poszczególne węzły sieci komputerowej w zależności od ich "samodzielności" w czasie alokacji. Wyodrębniamy więc:

- węzły nadzorcze - przechowujące informacje i komponenty niezbędne do zaalokowania całego systemu,
- węzły lokalnie nadzorcze - przechowujące informacje i komponenty wystarczające do zaalokowania podsystemu w jednym węźle,
- węzły lokalnie samodzielne - przechowujące komponenty lokowane w danym węźle,
- węzły uzależnione - mogące współpracować tylko z istniejącym oprogramowaniem.

Powyższy podział jest dodatkowo uzasadniony różnymi możliwościami sprzętowymi komputerów wchodzących w skład sieci.

Jednym z podstawowych zadań realizowanych przez węzeł nadzorczy jest możliwość przechowywania pliku opisu stanu sieci. Plik ten aktualizowany jest w wyniku działania alokatora. Należy zauważyć, że ponieważ możliwe jest istnienie aktualnej kopii pliku opisu sieci w więcej niż jednym węźle to istnieje możliwość zrezygnowania z wyróżnienia tylko jednego z nich jako węzła nadzorczego i mówienie o kilku węzłach nadzorczych. W takiej sytuacji każda akcja alokacji musi informować wszystkie węzły nadzorcze o zmianach w pliku opisu sieci, które się dokonały (bardzo dogodna jest w takim przypadku funkcja rozgłaszania). Należy ponadto zapewnić, że więcej niż jedna akcja alokacji nie będzie wykonywała się w danym momencie).

Kolejnym pytaniem będzie "Jak opisywać dopuszczalne powiązania wzajemne pomiędzy komponentami oraz sposób przypisania komponentów do konkretnych węzłów?". W systemie COSID przewiduje się istnienie specjalnego języka opisu alokacji, który, dzięki zastosowaniu jako aparatu formalnego gramatyk grafowych [1], pozwala na łatwą modyfikację struktury powiązań oraz doprecyzowanie tych powiązań na żądanie alokatora. Taki konwersacyjny tryb pracy (wspomagany dodatkowo przez specjalnie zaprojektowany system konwersacyjny) jest wyjątkowo dogodny dla projektanta systemu, tworzącego jego kolejne, możliwie coraz bardziej doskonałe wersje. Nie jest on jednak zadowalający dla użytkownika systemu, który wcale nie musi być zainteresowany znajomością wszystkich szczegółów dotyczących systemu (a wprost przeciwnie zwykle chce od nich abstrahować). W tym przypadku system COSID oferuje wsadowy tryb pracy alokatora, który pozwala zaalokować system według propozycji pozostawionych przez projektanta systemu.

Jedną z podstawowych zalet systemu alokacji wspomaganego przez COSID jest możliwość opóźnienia definiowania nazw globalnych kreowanych komponentów do etapu kompilacji, dzięki wprowadzonemu w [6], mechanizmowi językowemu, który pozwala na etapie kompilacji wyznaczyć tylko typy współpracujących ze sobą komponentów. Pozwala to na dynamiczne definiowanie podsystemów, w kolejności zgłaszania się do pracy poszczególnych węzłów sieci.

Ponieważ celem niniejszego artykułu jest zasygnalizowanie udogodnień proponowanych przez system alokacji nie omawiano w nim kilku problemów, których rozwiązanie jest niezwykle istotne dla systemu alokacji, mających jednakże bardziej techniczny charakter. Są nimi:

- sposób realizacji oddzielnej kompilacji,
- sposób transmitowania oprogramowania pomiędzy węzłami,
- struktura systemu alokacji,

- stopień rozproszenia funkcji alokacji,
- Są one bardziej szczegółowo omówione w [3].

Prace implementacyjne rozpoczęte w bieżącym roku w ramach projektu COSID pozwalają żywić nadzieję na praktyczną weryfikację naszkicowanej koncepcji generowania oprogramowania dla lokalnych sieci komputerowych.

Literatura

- [1] Flasiński M. "Characteristic of an Atributed Z-edNIO Graph Grammar for Description Distributed Systems".
- przyjęte do druku w kwartalniku AGH "Elektrotechnika"
- [2] ISO "Information processing system - Open system interconnection - Basic Reference Model"
Internal Standard ISO/OSI 7498, 1983r.
- [3] Katana G., Kotulski L., Sobala A.: "Alokacja w systemie COSID".
- złożone w redakcji Zeszytów Naukowych EJ
- [4] Kotulski L. "Wprowadzenie do systemu COSID".
Raport KIUJ nr 1/87
- [5] Kotulski L. "The introduction demands for the Support of Implementation Distributed Software System".
- w przygotowaniu -
- [6] Kotulski L., Pietraszko J. "Językowe uwarunkowania postkompilacyjnej alokacji".
- w przygotowaniu -

SOME NOTES CONCERNING DISTRIBUTED SYSTEMS SOFTWARE CONSTRUCTION

This paper presents some notes concerning distributed systems software construction. Problems referring to modules manipulation for this particular software are discussed more thoroughly, and creation of an allocator - a special program for servicing such a manipulation is proposed.

МЕСТО АЛОКАЦИИ В СИСТЕМЕ ГЕНЕРАЦИИ СОГЛАСОВАННОГО ПРОГРАММИРОВАНИЯ ДЛЯ ЛОКАЛЬНЫХ КОМПЬЮТЕРНЫХ СЕТЕЙ

Работа касается сложения программного обеспечения местных вычислительных сетей. Подробно обсуждены проблемы манипулирования модулями этого программного обеспечения. Предложено тоже создание специальной программы, названной "алокатор" для обслуживания этого манипулирования.

Irena KLIMEK-GRZESIAK^x

lokalna sieć komputerowa,
informatyczny system
szpitalny

WYMAGANIA STAWIANE LOKALNYM SIECIOM KOMPUTEROWYM PRZEZ INFORMATYCZNY SYSTEM SZPITALNY

W artykule omawia się informatyczny system dla szpitala wieloprofilowego, dokonuje się podziału na podsystemy, charakteryzuje się je, precyzuje wymagania funkcjonalne systemu oraz strukturę bazy danych. Informatyka na świecie, a szczególnie w krajach o wysokiej technice, znajduje szerokie zastosowanie w medycynie i organizacji ochrony zdrowia. W Polsce także wiele Ośrodków zajmuje się tą problematyką.

1. WSTĘP

Informatyka jako niezależna dziedzina nauki powstała stosunkowo niedawno.

Na świecie znajduje ona szerokie zastosowanie w różnych dziedzinach nauki a w tym w medycynie i organizacji ochrony zdrowia ze względu na ogromną ilość informacji, która jest potrzebna do opisywania stanu zdrowia społeczeństwa.

Historycznie rzecz biorąc, informatyka do medycyny i organizacji ochrony zdrowia wkroczyła na świecie u schyłku lat sześćdziesiątych. Do najczęściej realizowanych tematów w zakresie informatyzacji należą:

- medyczne systemy informatyczne w:
 - szpitalach,
 - ambulatoriach,
 - opiece pielęgniarstwa,
- kliniczne wspomaganie procesu leczenia,
- zastosowanie mikro- i minikomputerów w:
 - ultrasonografii,
 - medycynie nuklearnej,
 - radiologii,
 - analizie EKG,
 - epidemiologii i statystyce medycznej,
 - analizie EEG,
 - laboratorium analitycznym,
 - intensywnej terapii,
 - formacji i gospodarce lekami, itd.

^x Ośrodek Informatyki Medycznej, Wrocław, Pl. Dzierżyńskiego 6

Światowy dorobek w zakresie stosowania informatyki w Służbie Zdrowia prezentowany jest co 3 lata na Światowych Kongresach MEDINFO /ostatni Kongres odbył się w 1986 r. w Waszyngtonie, następny odbędzie się w 1986 r. w Chinach/.

W Polsce prace informatyczne prowadzi większość placówek Służby Zdrowia ale w większości są to systemy specjalistyczne uwzględniające potrzeby lokalne a eksploatacja ich odbywa się na różnym sprzęcie komputerowym.

Pod auspicjami Wydziału Zdrowia i Opieki Społecznej - Urzędu Wojewódzkiego we Wrocławiu powołano Ośrodek Informatyki Medycznej, który zajmuje się analizą, opracowywaniem i wdrożeniem systemów informatycznych szpitalnych. Dorobek tych prac również prezentowany jest na Kongresach Światowych /1/ /2/.

Lata osiemdziesiąte stworzyły realną szansę na stosowanie informatyki w medycynie i jednostkach Służby Zdrowia ze względu na dostępny w Polsce sprzęt mikrokomputerowy, który może zapewnić rozproszoną strukturę systemu informatycznego z lokalnymi punktami przetwarzania.

Taki sprzęt komputerowy ma przewagę nad innym: nie zajmuje dużej powierzchni i nie wymaga klimatyzacji.

2. KONCEPCJA INFORMATYCZNEGO SYSTEMU SZPITALNEGO /HIS/

2.1. Analiza struktury informacyjnej szpitala

W Wydziale Zdrowia i Opieki Społecznej we Wrocławiu opracowano "Koncepcję komputerowego systemu informatycznego dla szpitala wieloprofilowego" /3/, w której uwzględniono podział na dwa główne piony:

- medyczny,
- administracyjno-ekonomiczny.

W pionie medycznym wyodrębniono pewne autonomiczne funkcjonujące jednostki:

- izbę przyjęć,
- dział rejestracji, statystyki i dokumentacji,
- oddziały,
- poradnie,
- pracownie diagnostyczno-lecznicze,
- aptekę,
- punkt krwiodawstwa,
- dział sterylizacji.

W pionie administracyjno-ekonomicznym wyodrębniono między innymi:

- dział służb pracowniczych,
- dział żywienia,
- dział ekonomiczno-finansowy,
- dział administracyjno-gospodarczy,
- dział eksploatacyjno-gospodarczy.

W ramach funkcji, jakie spełniają w/w jednostki w pionie medycznym można dokonać następującego podziału:

A - jednostki organizacyjne obsługujące i wspomagające proces;

- badania,
- diagnozowania,
- leczenia pacjenta,

B - jednostki zabezpieczające proces leczenia;

C - jednostki analizujące działalność pionu medycznego pod kątem zarządzania i statystyki.

W grupie A znajdują się:

- poradnie /ambulatorium/, oddziały, pracownie diagnostyczno-lecznicze, punkt krwiolecznictwa.

Do grupy B należą:

- apteka, dział sterylizacji.

Do grupy C należą:

- dział rejestracji, statystyki i dokumentacji,

który jest ogniwem łączącym pion medyczny z administracyjno-ekonomicznym.

Powyższy podział szpitala ma dwa piony pociąga za sobą podział informacji na:

- informacje związane z pacjentem, oraz
- informacje administracyjno-ekonomiczne.

Te dwa rodzaje informacji mogą prowadzić w konsekwencji do budowania dwóch niezależnych, autonomicznie pracujących systemów informatycznych:

- KOMPUTEROWEGO SYSTEMU INFORMATYCZNEGO - SZPITAL /HOSPITAL INFORMATION SYSTEM/
- KOMPUTEROWY SYSTEM INFORMATYCZNY - ADMINISTRACJA

System ADMINISTRACJA, w którym przetwarzane byłyby informacje dotyczące kadr, płac, finansów, księgowości, gospodarki materiałowej, środków trwałych, ze względu na jego powszechne dotychczasowe stosowanie w różnych gałęziach gospodarki społecznej, nie był przedmiotem rozważań koncepcyjnych.

2.2. Podsystemy informatyczne szpitalne

- ich autonomiczności i zintegrowanie

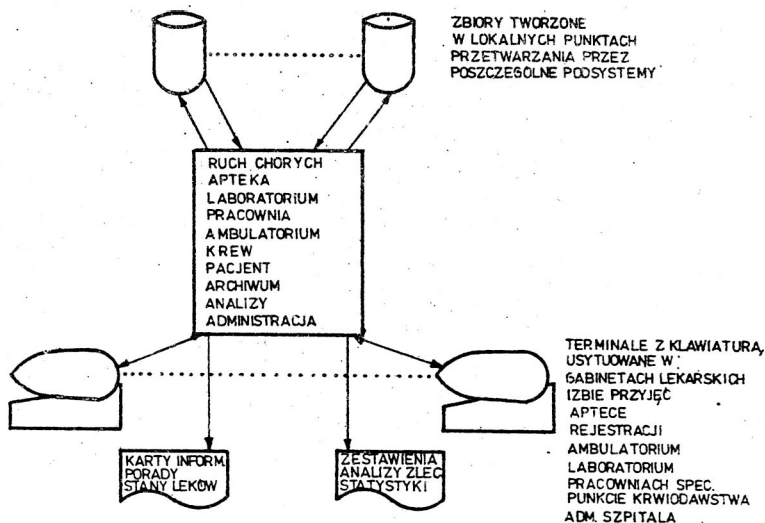
Źródłem powstawania informacji grupy medycznej są:

Izba Przyjęć, dział Rejestracji, Statystyki i Dokumentacji, Ambulatorium, Oddział szpitalny, Zakład Diagnostyki Laboratoryjnej, pracownie specjalistyczne, apteka, punkt krwiolecznictwa.

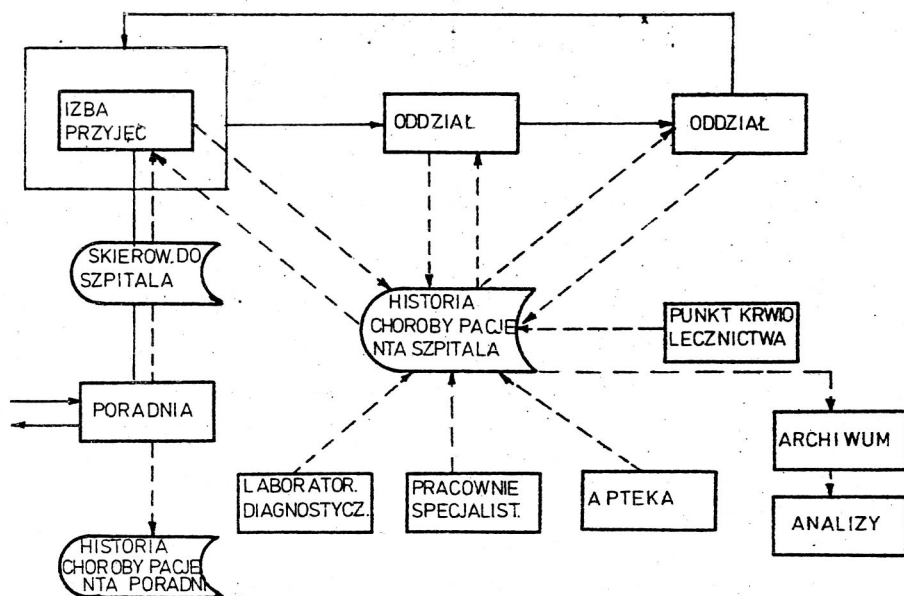
Analizując informacje o pacjencie powstające w wielu punktach szpitala wyodrębniono podsystemy informatyczne z lokalną bazą danych, pracujące automatycznie zilustrowane rys. 1.

Ilustruje to rys. 1.

Rys.1 SCHEMAT SYSTEMU INFORMATYCZNEGO SZPITALA



Izba przyjęć oraz dział rejestracji, statystyki i dokumentacji wyznacza podsystem RUCH CHORYCH, który można zobrazować rys. 2.



Rys. 2 UPROSZCZONY OBIEG INFORMACJI O PACJENCIE

Oddział szpitalny, ambulatorium punkt krwiolecznictwa, zakład diagnostyki laboratoryjnej i pracownie specjalistyczne dostarczają informacji do historii choroby pacjenta i wyodrębniają następny podsystem PACJENT /komputerowa wersja historii choroby/.

Niezależnie funkcjonują autonomicznie pracujące podsystemy: PRACOWNIA, AMBULATORIUM, APTEKA, KREW.

Wszystkie nagromadzone informacje o pacjencie muszą być archiwowane, aby po pewnym czasie stanowiły bogaty materiał do wszelkiego rodzaju analiz i prognozowania - bardzo ważnych elementów z punktu widzenia epidemiologii.

Te informacje wyznaczają podsystem ARCHIWUM i ANALIZY. Najważniejszym podsystemem, który z punktu widzenia bogatego empirycznego materiału archiwalnego przechowywanego na magnetycznych nośnikach informacji, najbardziej interesuje lekarzy, jest podsystem PACJENT - komputerowa wersja historii choroby. Wyodrębniono w niej następujące obszary informacyjne:

- nr księgi głównej,
- dane demograficzne i identyfikacyjne,
- rozpoznanie wstępne,
- wywiad,
- badania przedmiotowe /ogólne, miejscowe/,
- badania laboratoryjne i specjalistyczne,
- obserwacja,
- różnicowanie,
- metody leczenia,
- prognozowanie,
- korelacja z innymi schorzeniami,
- powikłania,
- rozpoznanie końcowe,
- epikryza,
- zalecenia poszpitalne,
- zgon.

Do obszarów tych powinno nanosić się informacje, dopisywać je i archiwować w celu późniejszego ich odzyskania w postaci wydruków lub obrazów na monitorze ekranowym w trybie teletransmisji.

Doświadczenia lekarzy klinicystów wskazują, że duża część informacji o pacjencie jest niewykorzystana z powodu niedokładnego ich zbierania i przechowywania.

Archiwizowany materiał zawarty w historii choroby prowadzonej tradycyjnie jest po pewnym czasie nieprzydatny z powodu braku wypracowanego sposobu zbierania danych z wywiadu i badania przedmiotowego, ich notowania i standaryzacji.

Nadzieja lekarzy klinicystów, co do oczekiwanego systemu PACJENT może być spełniona po okresie całkowitego wdrożenia podsystemów:

RUCH CHORYCH, PRACOWNIA, APTEKA, KREW. Współbieżnie z podsystemem PACJENT można opracowywać podsystem AMBULATORIUM.

3. WYBÓR ŚRODKÓW TECHNICZNYCH

3.1. Wymagania funkcjonalne systemu

Zgodnie z przedstawioną powyżej ogólną koncepcją, system szpitalny ma być specjalizowanym systemem wielodostępnym o następujących cechach:

- stosunkowo duża ilość terminali^{1/},
- orientacje na gromadzenie, wyszukiwanie, wyświetlanie i drukowanie informacji,
- bieżąca baza danych o średniej objętości 40 MB i strukturze logicznej dopuszczającej rozproszenie,
- system zbiorów archiwalnych o rosnącej pojemności /około 200 MB rocznie/ przetwarzany off-line,
- praca ciągła /przez całą dobę/ w zakresie niektórych funkcji lub dwuzmianowe - dla pozostałych,
- wysoki stopień niezawodności i gotowości do pracy,
- system operacyjny zapewniający wielodostęp i wielozadaniowość,
- krótki czas reakcji przy realizacji działań programowych,
- konwersacyjny tryb pracy.

3.2. Wybór struktury bazy danych

W krajach o zaawansowanej technice istnieją informatyczne systemy szpitalne ze scentralizowaną bazą danych, o szybkim czasie reakcji i systemem CICS obsługującym i kontrolującym pracę terminali. Lokalne sieci komputerowe narzucają strukturę rozproszoną.

Istotną zaletą struktury rozproszonej jest podział funkcji systemu pomiędzy wiele komputerów i możliwość rozproszenia bazy danych.

Dzięki temu uzyskuje się na ogół skrócenie czasów reakcji systemu, szczególnie przy realizacji funkcji lokalnych, tzn. funkcji wywołanych z terminala pewnego komputera i dotyczących zbiorów zlokalizowanych w tymże samym komputerze.

System o strukturze rozproszonej może - w odróżnieniu od systemu scentralizowanego - funkcjonować, mimo awarii niektórych jego elementów, chociaż zwykle prowadzi to do częściowego ograniczenia funkcji użytkowych systemu. Struktura rozproszona ma pod względem niezawodności wyraźną przewagę nad strukturą scentralizowaną, jest jednak trudniejsza w realizacji.

^{1/} Jeden lokalny punkt przetwarzania zawiera 3-4 terminale; w szpitalu wieloprofilowym o średniej ilości łóżek około 600 szacuje się 25 lokalnych punktów przetwarzania.

BIBLIOGRAFIA

- [1] Irena Klimek-Grzesiak - Computer System for Hospital Management - MEDINFO 86 - Waszyngton, 1986, edited by R. Salamon, B. Blum and M. Jorgensen
- [2] Irena Klimek-Grzesiak, Józef B. Lewoc, Jerzy Lorenz, Wojciech Minta - "An Approach to the development of a Complex Information System for the Wrocław Province Hospital, Medical Informatics EUROPE 85, Helsinki, Finland, August 85.
- [3] Koncepcja komputerowego systemu informatycznego dla szpitala wieloprofilowego - Praca zbiorowa przy koordynacji Ireny Klimek-Grzesiak, Urząd Wojewódzki - Wydział Zdrowia i Opieki Społecznej, Wrocław 1985.

THE HOSPITAL INFORMATION SYSTEM REQUIREMENTS FOR LOKAL NETWORKS

This paper presents an Information System for Hospital of many medical specialities. The system presented is divided into subsystems according to the functional demands as well as data base in its structure is fitted to the characteristics of hospital activities and usual medical specialities. The application of the information systems in health service has significantly increased especially in developed countries. In Poland there are also many computing centers involved in those problems.

КОНЦЕПЦИЯ ИНФОРМАТИЧЕСКОЙ СИСТЕМЫ ДЛЯ БОЛЬНИЦЫ

В статье представлена концепция информатической системы для многопрофильной больницы, представлены функциональные требования к системе, подана характеристика подсистем этой системы, представлена структура базы данных. Информатика в мире, в особенности в странах с высоко развитой техникой находит широкое применение в медицине и организации охраны здоровья. Также в Польше многие научные центры занимаются этой проблематикой.

Leszek KOTULSKI *

lokalna sieć komputerowa,
język COLNET

Możliwości zastosowania języków współbieżnych do implementacji
oprogramowania w lokalnych sieciach komputerowych **

Propozycja spojrzenia na oprogramowanie lokalnej sieci komputerowej jako na jeden program współbieżny wymaga wprowadzenia definicji nowego języka programowania, gdyż istniejące języki programowania nie spełniają wymagań stawianych przez lokalne sieci komputerowe. Możliwość spełnienia wymagań stawianych przed językami pozwalającymi zaimplementować oprogramowanie lokalnych sieci komputerowych jest zilustrowany na przykładzie nowo zaprojektowanego języka COLNET.

1. Wstęp

Koncepcja opisu oprogramowania lokalnej sieci komputerowej (skr. LSK) przy pomocy języka programowania współbieżnego zwiększa czytelność systemu i ułatwia jego pielęgnację. Przy założeniu, że algorytmy realizujące warstwę transportową (wg. ISO OSI Reference Model [3]) będą procedurami standardowymi albo wprost rozkazami języka zgodność oprogramowania z protokołami warstwy aplikacyjnej powinna odpowiednio wynikać z reguł semantycznych tego języka. Trudności z realizacją takiego podejścia wynikają głównie z niedostosowania istniejących języków programowania współbieżnego do wymagań LSK. Celowym wydaje się więc projektowanie specjalizowanych języków "sieciowych" umożliwiających implementację oprogramowania w LSK. Minimalny podzbiór wymagań, które muszą spełniać te języki jest omówiony w rozdziale 2. Realność spełnienia tych wymagań ilustruje język COLNET (COncurrent Language for NETworks) [9].

2. Wymagania stawiane językom sieciowym

Podstawą dla analizy użyteczności języków sieciowych w porównaniu z językami współbieżnymi będzie przedstawione poniżej porównanie własności LSK i wieloprocesorowych maszyn cyfrowych z rozproszoną pamięcią.

1 - Szybkość transmisji w LSK jest zdecydowanie niższa niż w maszynach wieloprocesorowych ze względu na długość i organizację łącza.

* Katedra Informatyki Uniwersytetu Jagiellońskiego, Kraków Kopernika 27

** Praca została wykonana w ramach programu badawczego MNiSzW RP.I.09
"Rozwój języków, metod i podstaw formalnych programowania".

- 2 - Realizacja transmisji jest z reguły programowa lub programowo-sprzętowa (sterowniki) w przypadku LSK i mikroprocesorowa lub sprzętowa w przypadku wieloprocessorowych maszyn cyfrowych.
- 3 - LSK charakteryzuje się większą różnorodnością sprzętu, niż maszyny wieloprocessorowe, gdzie nawet różnorodne procesory powiązane są pewną ideą logiczną.
- 4 - Oddalenie od siebie węzłów LSK utrudnia, a nawet uniemożliwia, uaktywnianie całej sieci w tym samym czasie co powoduje logiczne zróżnicowanie struktury sieci.
- 5 - Użytkownicy pracujący w danym węźle ma manualnie łatwiejszy dostęp do zasobów tego węzła niż do zasobów innego węzła np. drukarki.

Zauważmy, że dwie pierwsze spośród wymienionych cech LSK wykluczają możliwość dekompozycji programu sekwencyjnego na podobliczenia współbieżne tak jak przykładowo ma to miejsce przy obliczeniach wektorowych w maszynie CRAY. Należy więc założyć, że programista musi wprost zasygnalizować współbieżne zachowanie systemu wyznaczając fragmenty programu, które będą wykonywały się sekwencyjnie (nazwane procesami sekwencyjnymi). Rozproszone środowisko wyklucza użycie języków bazujących na koncepcji zmiennej wspólnej takich jak Concurrent Pascal [1]. Troska o efektywność projektowania i pielęgnacji systemu zmusza nas do rezygnacji z języków wykorzystujących wymianę komunikatów jako jedyny mechanizm komunikacji i synchronizacji takich jak Communicating Port [11] bazujący na koncepcji CSP. W językach CHILL [13], Concurrent C [12] i MOD [2] ze względu na efektywność założono, że procesy zgromadzone w jednym węźle będą wykorzystywały zmienne dzielone udostępniane przez monitor, oraz wymianę komunikatów w innym przypadku. Pomimo, że wyżej wymienione języki jako jedyne umożliwiają wskazanie węzła, w którym powinien być umieszczony dany moduł programowy co umożliwia dostosowanie oprogramowania do konfiguracji sprzętowej (patrz uwaga 3) to czynią to już na etapie kompilacji, co w momencie rekonfiguracji sieci (patrz uwaga 4) zmusza nas do modyfikacji programu źródłowego i ponownej kompilacji systemu. Stosowanie dualnego mechanizmu komunikacji w zależności od wzajemnego położenia procesów czyni taką modyfikację nieomal nieopłacalną.

Krótki przegląd możliwości zastosowania języków programowania współbieżnego doprowadził nas do wniosku, że żaden z nich nie jest użyteczny dla oprogramowania LSK. Celowym wydaje się podanie podstawowych wymagań stawianych przed językiem sieciowym:

1. Jawna specyfikacja obiektów pracujących współbieżnie.
2. Możliwość dystrybucji oprogramowania w rozproszonym środowisku.
3. Jednorodny mechanizm komunikacji i synchronizacji niezależny od wzajemnego położenia komunikujących się obiektów.

4. Możliwość opóźnienia definiowania połączeń międzymodułowych do etapu alokacji przy jednoczesnym zapewnieniu kontroli poprawności przekazywania parametrów już na etapie kompilacji.
5. Możliwość odróżnienia algorytmów sprzętowo-zależnych od sprzętowo-niezależnych.

3. Zarys koncepcji języka COLNET

Oddzielenie etapu kompilacji od alokacji jest podstawowym założeniem systemu COSID [10] wspomagającego implementację oprogramowania w LSK. W niniejszym rozdziale zagadnienie to omówione zostanie od strony mechanizmów języka COLNET [9], który umożliwia zdefiniowanie algorytmów poszczególnych składowych oprogramowania LSK. Zagadnienie łączenia tych składowych zostanie omówione w materiałach konferencji [4].

W celu umożliwienia alokowania poszczególnych obiektów niezależnie od wcześniejszej alokacji innych obiektów koniecznym jest by mechanizm komunikacji był jednorodny z punktu widzenia programu, tj. na etapie kompilacji oraz różnicowanie go na etapie alokacji, aby podnieść efektywność systemu. Powyższe założenie pozwala spełnić wprowadzona w języku COLNET idea modułu pośredniczącego [8], która rozszerza koncepcję monitora w ten sposób, że staje się ona użyteczna również w systemach z rozproszoną pamięcią. Moduł pośredniczący zamienia wywołanie procedury zewnętrznej monitora na wysłanie komunikatu z parametrami wejściowymi oraz odbioru komunikatu z parametrami wyjściowymi. Ten schemat pracy jest jednakże ukryty przed procesem wywołującym (który w dalszym ciągu wykonuje operację wywołania monitora) i wykonywany jest albo automatycznie albo w sposób opisany wewnątrz monitora. W tym drugim przypadku, koncepcja monitora dalej nazywana managerem, rezerwując nazwę monitor tylko dla kolekcji procedur zewnętrznych musi być wzbogacona o opis nowych procedur (specyfikowanych przez słowo kluczowe *interface*), które opisują sposób kodowania, wysyłania i odbierania komunikatów do lub z danej procedury zewnętrznej monitora.

Pomimo, że tekst każdego modułu pośredniczącego jest zdefiniowany wewnątrz managera to monitor grupujący procedury zewnętrzne oraz moduły pośredniczące do tych procedur są generowane jako osobne obiekty. Moduły pośredniczące są przypisywane do tych węzłów, z których potencjalne wywołanie danej procedury może się pojawić. Tak więc proces po prostu wywołuje odpowiedni moduł pośredniczący. Jeżeli taki moduł pośredniczący do danej procedury nie jest zdefiniowany wewnątrz managera to akcja modułu pośredniczącego jest symulowana przez jądro wspomagające system. Podczas tej symulacji następujące operacje są wykonywane: dekodowanie parametrów wejściowych do postaci komunikatu, wysłanie komunikatu, odebranie i rozkodowanie komunikatu z danymi wyjściowymi. Kodowanie

komunikatu musi oczywiście uwzględniać możliwość różnorodnej reprezentacji tych samych typów danych w różnych maszynach cyfrowych. Jeżeli manager definiuje różnorodną obsługę wywołań w zależności z którego komputera pojawia się wywołanie wtedy zbiór modułów pośredniczących musi być wyprodukowany i właściwy z nich jest alokowany. Jeżeli dany proces zmieni swą lokalizację to tylko inny z modułów pośredniczących musi być alokowany razem z nim.

Szczegółowy opis konstrukcji języka COLNET takich jak:

- tryb rozgłaszania, tzn. możliwość wysyłania kilku komunikatów równolegle lub sekwencyjnie jako reakcję na żądanie obsługi,
 - konstrukcji akcji, umożliwiającej współbieżny dostęp do zasobu kontrolowanego przez manager,
 - sprzętowo zależne moduły pośredniczące oraz sprzętowo zależne akcje, umożliwiające obsługę algorytmów zależnych sprzętowo np. przerwań,
- wykracza poza zakres referatu i jest omówiona odpowiednio w [8,6,7].

4. Konkluzja

Wykorzystanie w celu synchronizacji i komunikacji uogólnienia pojęcia monitora dzięki wprowadzeniu koncepcji modułu pośredniczącego pozwala w łatwy sposób wykorzystać bogaty dorobek teoretyczny dotyczący weryfikacji poprawności oprogramowania współbieżnego co stanowi podstawą obok łatwości pielęgnacji i czytelności programu zaletę stosowania języków współbieżnych do opisu całego oprogramowania lokalnej sieci komputerowej.

Kolejną zaletą wydaje się możliwość specyfikowania z poziomu języka programowania sposobu dokonania konwersji w przypadku różnic w reprezentacji tych samych typów danych w poszczególnych komputerach węzłowych. Należy przy tym zwrócić uwagę, że kodowanie komunikatu przez moduł pośredniczący zwalnia jądro systemu operacyjnego realizującego transmisję od wykonania tej czynności, więc sumaryczny czas kodowania i transmisji powinien być zbliżony do czasu transmisji w sieciach komputerowych. Możliwość zakodowania algorytmów obiektów w języku wysokiego poziomu posiadającym odpowiednie mechanizmy modularyzacji w połączeniu z elastycznością systemu alokacji [4] oraz wspomaganie śledzenia programów zaimplementowanych w języku COLNET [5] pozwalają żywić dodatkową nadzieję na konkurencyjność proponowanego podejścia.

5. Literatura

- [1] Brinch Hansen P. "The programming language Concurrent Pascal". IEEE Trans. on Soft. Eng., SE-1, 2, June 1975, 199-206.
- [2] Cook R.F. "MOD - A language for distributed programming". IEEE Trans. on Soft. Eng., SE-6, 6, Nov 1980, 563-571.
- [3] ISO "Information processing systems - Open System Interconnection - Basic Reference Model". Internal Standard ISO/OSI 7498, 1983.

- [4] Katana G., Kotulski L., Sobala A. "Miejsce alokacji w systemie generacji oprogramowania współbieżnego dla lokalnych sieci komputerowych". W materiałach konferencyjnych.
- [5] Kędzierski T., Kosowicz B., Kotulski L. "Środki wspomagania uruchamiania modułów w systemie COSID". W materiałach konferencyjnych.
- [6] Kotulski L., Moszner P. "Akcja - komponent umożliwiający równoległą pracę wewnątrz monitora". Złożone w redakcji Podstaw Sterowania.
- [7] Kotulski L., Onderka Z., Zabołotny A. "The manager and its cooperation with the hardware in local area networks". Przyjęte do druku w kwartalniku AGH "Elektrotechnika".
- [8] Kotulski L. "An Interface Procedure Concept - extension of the monitor concept to distributed programming". Złożone w redakcji IPL
- [9] Kotulski L. "Raport języka COLNET". Raport KIUJ nr 3/87.
- [10] Kotulski L. "Wprowadzenie do systemu COSID". Raport KIUJ nr 1/87
- [11] Mao W.T., Yeh R.T. "Communicating Fort: A language concept of Concurrent Programming". IEEE Trans. on Soft. Eng., SE-6, 2, 1980, 194-204.
- [12] Tsujno Y., Ando M., Araki T., Tokura N. "Concurrent C: A Programming Language for Distributed Multiprocessor System." Software Practice and Experience, 14, 1984, 1061-1078.
- [13] Z200 Recommendation CCITT HIGH LEVEL LANGUAGE CHILL, Geneva 1981.

POSSIBILITIES OF USING CONCURRENT LANGUAGES TO IMPLEMENTATION OF SOFTWARE ENVIRONMENT OF LANs

The advantages of implementation the whole software environment by one concurrent language are selfevident during modification and maintenance of the system. However a few new properties of the above language must be added to the properties of the concurrent languages for the distributed environment only. As an illustration of the problem the main idea of the COLNET language COncurrent Language for NETworks are described.

Возможности использования паралельных языков с целью имплементации средств опрограмирования для местных вычислительных сети.

Удобство имплементации целого средства опрограмирования для местных вычислительных сети через один паралельный язык програмирования особенно заметно во время консервации опрограмирования. Такой язык должен обладать несколькими дополнительными свойствами. В статье названо свойства, которые не выступают в паралельных языках. Язык COLNET, который може в общих чертах описано, иллюстрирует возможность выполнения этих свойств.

Leszek Kotulski^x

**język COLNET,
baza danych**

O możliwościach zastosowania języka COLNET do
implementacji rozproszonych baz danych

Koncepcja "modułu pośredniczącego" umożliwia wykorzystanie konstrukcji monitora w środowisku programowym z rozproszoną pamięcią. Koncepcja "akcji" umożliwia współbieżną obsługę żądań procesu przez kilka niezależnych modułów programowych wykreowanych przez monitor i używających jego struktur danych wyłącznie w trybie odczytu. Obie te koncepcje umożliwiają implementację oprogramowania rozproszonej bazy danych w bardzo naturalny sposób. Główna idea takiego rozwiązania wykorzystującego język COLNET jest proponowana.

Architektura lokalnych sieci komputerowych jest odpowiednia do realizacji koncepcji "rozproszonych baz danych". Lokalne bazy danych są w takim przypadku lokowane w poszczególnych węzłach sieci komputerowej. Dwa problemy należy rozwiązać w takim przypadku:

- opracować optymalny algorytm wyszukiwania informacji w lokalnych bazach danych,
- zsynchronizować pracę lokalnych baz danych, tak by bezkolizyjnie w jak najkrótszym czasie otrzymać dostęp do poszukiwanego zbioru informacji w bazie danych.

W niniejszym opracowaniu tylko drugi przypadek będzie nas interesował, zakładając że jesteśmy w stanie wyznaczyć kilka wystarczająco dobrych algorytmów wyszukiwania informacji w lokalnej bazie danych.

Zwiększenie szybkości dostępu do wyszukiwanej informacji można osiągnąć przez wprowadzenie równoległego przeszukiwania we wszystkich lokalnych bazach danych tworzących rozproszoną bazę danych lub przez wprowadzenie współbieżnego przeszukiwania w konkretnej lokalnej bazie danych. Konstrukcje programowe modułu pośredniczącego (interface) [1] i akcji (action) [2] wprowadzone w języku COLNET (COncurrent Language for NeTworks) [3] pozwalają zaimplementować obie te metody równocześnie.

^x Katedra Informatyki Uniwersytetu Jagiellońskiego, Kopernika 27, Kraków

Język COLNET pozwala opisać system współbieżny w terminologii procesów sekwencyjnych synchronizowanych przy pomocy managera. Z punktu widzenia synchronizacji procesów manager jest odpowiednikiem monitora, tzn. oferuje do użytku zestaw procedur przy pomocy których procesy mogą operować na globalnej strukturze danych utrzymywanej przez menagera. Tylko jedna z tych procedur może być wykonywana w danym momencie czasu.

Problem synchronizacji dostępu do lokalnych baz danych będzie w niniejszej propozycji rozwiązyany poprzez powierzenie ochrony lokalnych baz danych poszczególnym managerom.

Wprowadzona w [1] koncepcja modułu pośredniczącego umożliwia odwołanie się przez procesy do poszczególnych managerów opiekujących się bazami danych niezależnie od wzajemnego ich położenia, ponieważ moduł pośredniczący lokowany jest zawsze w węźle, z którego wskazana procedura managera może być wywołana a komunikacja pomiędzy nim a procedurą managera odbywa się przy pomocy wymiany komunikatów. Poszczególne moduły pośredniczące związane są ściśle z definicją poszczególnych procedur wejściowych managera (tj. dla każdej procedury managera musi istnieć przynajmniej jeden moduł pośredniczący) i z tego powodu definiowane są na poziomie managera (tj. w jego kodzie źródłowym). Standardowy algorytm pracy modułu pośredniczącego zakłada:

- zakodowanie parametrów wejściowych do procedury w formie komunikatu,
- wysłanie komunikatu do managera,
- odebranie komunikatu o wykonaniu żądanej procedury managera,
- rozkodowanie odebranego komunikatu i zwrócenie parametrów powrotnych do procesu.

Wykonanie kolejno operacji decode, send answer i encode możemy więc traktować jako standardową akcję modułu pośredniczącego. W przypadku, gdy moduł pośredniczący dla danej procedury nie został zdefiniowany przyjmujemy, że standardowy moduł będzie pośredniczył w obsłudze.

W przypadku rozproszonych baz danych tylko niestandardowa obsługa będzie nas interesowała. Ponieważ jednym z parametrów operacji decode jest identyfikator managera, który jest "proszony" o obsługę żądania procesu to moduł pośredniczący może poprosić o tę obsługę kilka procesów równocześnie. Jedynym ograniczeniem (wynikającym z konieczności zapewnienia wykrywalności na etapie kompilacji niezgodności typów przekazywanych parametrów) jest wymaganie by managery te były tego samego typu. W omawianym zastosowaniu nie jest ono jednak istotne, gdyż niezależnie od niego oczekujemy, że sposób obsługi poszczególnych lokalnych baz danych będzie identyczny. Przykład 1 ilustruje sytuację, w której jedno wywołanie procedury SZUKAJ(...) uaktywnia kilka managerów (LBD1, LBD2 i LBD3).

```

interface to SZUKAJ(WHAT:key; var OK:boolean; var ANSW:info) ;
var BUFF: buffer;
    ANSW_NUMB:=0..3;
begin
decode(LBD1; BUFF);      send(BUFF);
decode(LBD2; BUFF);      send(BUFF);
decode(LBD3; BUFF);      send(BUFF);
OK := false;
ANSW_NUMB := 0;
while not OK and (ANSW_NUMB < 3) do
    begin
        answer(BUFF);      encode(BUFF = OK, ANSW);
        ANSW_NUMB := ANSW_NUMB + 1
    end;
return;      {zwrot parametrów do procesu i uaktywnienie go}

LBD1.KONIEC;      {wysłanie sygnałów do managerów      }
LBD2.KONIEC;      {LBD1, LBD2 i LBD3, że należy        }
LBD3.KONIEC;      {zakończyć przeszukiwanie          }
end;

```

Przykład 1.

Pierwsza odpowiedź o pomyślnym zakończeniu poszukiwania albo wszystkie odpowiedzi o negatywnym poszukiwaniu powodują zwrot informacji do procesu (operacja return) aby umożliwić mu dalszą pracę. W przypadku odpowiedzi pozytywnych celowym wydaje się zlecenie pozostałym managerom by zaniechali dalszego poszukiwania wywołując odpowiednio procedurę wejściową KONIEC w poszczególnych managerach. Dla umożliwienia managerowi wykorzystania tej informacji nowa konstrukcja językowa akcja musi być wprowadzona. Pod pojęciem akcji rozumiemy moduł programowy, który na zlecenie managera (i pod jego kontrolą) wykonuje operacje na kontrolowanej przez manager strukturze danych. Zlecenie akcji pewnego zadania do wykonania wykonując operację do powoduje wstrzymanie wykonywania danej procedury managera do czasu, aż akcja zakończy swoją pracę. Na ten czas dostęp do managera jest odblokowany analogicznie jak w przypadku rozkazu delay w monitorze. Umożliwia to wykonanie procedury KONIEC współbieżnie z wykonaniem akcji SZUKAJ i skasowanie tej akcji przy pomocy operacji delete, która umożliwia kasowanie akcji mających dostęp do struktury danych jedynie w trybie odczytu.

Koncepcja akcji umożliwia również organizację współbieżnego dostępu do zasobu w sytuacji gdy zasób ten jest wykorzystywany biernie np. odczyt informacji. Dzieje się tak gdy procedura managera zleci współbieżne wykonanie kilku akcji (operacja condo). Mogą one poszukiwać informacji wg różnych algorytmów. Sytuacja taka jest rozważana w [2]. Praktyczne korzyści powyższego rozwiązania będą widoczne jeżeli manager jest alokowany w węźle obsługiwanym przez wieloprocessorową maszynę cyfrową albo

gdy dostęp do chronionego zasobu jest wykonywany przez urządzenie zewnętrzne co umożliwia efektywną realizację techniki podziału czasu.

Ponieważ celem niniejszego opracowania jest raczej zasygnalizowanie możliwości rozwiązania problemu niż pokazanie jego efektywnej implementacji (ze względu na czytelność) w przykładzie 1

- założono, że nazwy managerów znane są już na etapie kompilacji, pomimo istnienia mechanizmów językowych umożliwiających definiowanie tych nazw dopiero na etapie alokacji,
- nie uwzględniono faktu, że niewiele bardziej skomplikowany algorytm modułu pośredniczącego pozwoliłby nam uniknąć wysyłania komunikatów do managerów, które zakończyły już przeszukiwanie. Nie omówiono też mechanizmów językowych wykluczających możliwość powstania błędów uwarunkowanych czasowo w czasie współbieżnego poszukiwania informacji w lokalnej bazie danych.

Literatura

- [1] Kotulski L. "An Interface Procedure Concept - extension of the monitor concept for Distributed Programming.
Złożone w redakcji Information Processing Letters.
- [2] Kotulski L., Koszner P. "Akoja - Komponent umożliwiający równoległą pracę wewnątrz monitora.
Złożone w redakcji Podstaw Sterowania.
- [3] Kotulski L. "Raport języka COLNET".
Raport Katedry Informatyki UJ nr 3/37.

Possibilities of using COLNET language to implementation Distributed Data Bases.

The interface procedure concept allows to use the monitor concept in distributed environment. The action concept allows to concurrent service of the process request by a few independent modules created by a monitor and use its data structure in read-only mode. Both the concepts allow to implement an distributed data base in very natural way. The main idea of such solution in COLNET language is proposed.

О возможности использования языка COLNET к имплементации раздробленных баз данных.

Идея "интерфейса" даёт возможность использования конструкции монитора в местных вычислительных сетях. Идея "акции" делает возможным параллельное обслуживание одного требования процесса через несколько независимых модулей созданных монитором и читающих его данные. Обе эти идеи делают возможным легко имплементировать средства опrogramирования в раздробленных базах данных. Главная идея решения этой проблемы в языке COLNET описана в статье.

sieć lokalna, szacowanie przepustowości, transmisja zbiorów

Józef B. LEWOC^x, Leszek Z. MISIURA^{xx}

ANALIZA PRZEPUSTOWOŚCI W SIECIACH LOKALNYCH Z TRANSMISJĄ ZBIORÓW

W artykule przedstawiono opis metody analitycznej dla przybliżonego szacowania przepustowości sieci lokalnych o strukturze pierścieniowej lub magistralowej. Przeanalizowano i przedstawiono wyniki uzyskane przy wykorzystaniu metody dla sieci nienasyconej i nasyconej. Szczególną uwagę zwrócono na zachowanie sieci podczas transmisji zbiorów.

1. WSTĘP

Obserwacje pracy bardzo popularnej sieci lokalnej Ethernet wykazują, że w przypadkach jednoczesnej transmisji więcej niż jednego zbioru, czas przesyłania każdego zbioru wydłuża się kilkudziesięciokrotnie w stosunku do czasu potrzebnego do przesyłania tego zbioru, gdy nie jest transmitowany żaden inny zbiór.

Brak dobrych metod analitycznych do rozwiązywania konkretnych problemów sieci komputerowych był przyczyną poszukiwania przybliżonych metod oceny przepustowości [1-6]. Opracowana metoda daje bardzo dobre wyniki dla sieci nienasyconych, tzn. takich, w których łączne zapotrzebowanie na przepustowość poszczególnych linii nie przekracza dostępnej przepustowości. Natomiast, jak wykazują badania symulacyjne, z chwilą, gdy ruch oferowany przekracza przepustowości linii, przyjęte przybliżenia stają się niedokładne i należy poszukiwać innych przybliżeń.

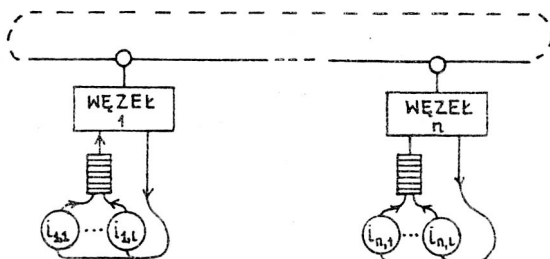
Niniejsza praca jest poświęcona najprostszemu przypadkowi transmisji zbiorów i transmisji interrogacyjnej na uproszczonym modelu sieci lokalnej. Przypadek ten ma istotne znaczenie dla zrozumienia zachowania bardziej złożonych sieci lokalnych.

^x Centrum Obliczeniowe Uniwersytetu Wrocławskiego, Wrocław
^{xx} Centrum Obliczeniowe Politechniki Wrocławskiej, Wrocław

2. MODEL PRZYBLIŻONY

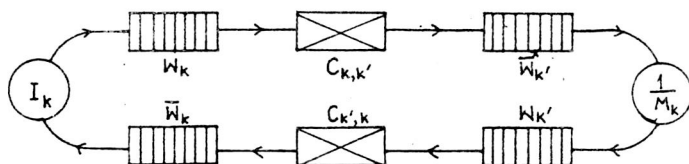
2.1. Ogólny model sieci lokalnej

Ogólny model sieci lokalnej badany cytowaną metodą wygląda następująco:



Dla rozważań niniejszej pracy nie ma istotnego znaczenia fakt, czy rzeczywista sieć ma strukturę pierścieniową, czy magistralową.

We wspomnianej metodzie przybliżonej sieć jest dekomponowana na oczka - odpowiedniki połączeń wirtualnych w sieci X.25. Oczko sieci (Rys.1.) składa się z użytkownika, kolejki do podsieci komunikacyjnej, podsieci komunikacyjnej, kolejki do obsługującego, właściwego (komputer obliczeniowy), stanowiska obsługi, kolejki do podsieci komunikacyjnej, podsieci komunikacyjnej i kolejki do użytkownika. Zachowanie oczka opisują zmienne losowe oznaczone na rys. 1.



Rys. 1. Oczko w sieci lokalnej

Fig. 1. Local area network loop

Oznaczmy wartości oczekiwane tych zmiennych losowych przez odpowiednie małe litery. Przy takich oznaczeniach wartość oczekiwana przepustowości (delivered traffic) w oczku wynosi:

$$d_k = \frac{1}{t_k} = \frac{1}{i_k + w_k + c_{k,k'} + \bar{w}_k + \frac{1}{M_k} + \bar{w}_{k'} + c_{k',k} + \bar{w}_k} \quad (1)$$

Dla sieci nienasyconych dobre wyniki daje przybliżenie [3]:

$$w = \frac{(n-2)\bar{t}}{(n-1)(t-\bar{t})} \left(\frac{1}{2+2(n-1)(t-\bar{t})} \left(\frac{1}{1+2(n-1)(t-\bar{t})} \left(\dots \left(\frac{1}{1+2(n-1)(t-\bar{t})} \right) \dots \right) \right) \right) \quad (2)$$

$$\bar{t} = c_{k,k'} + c_{k',k}$$

Nadaje się ono dla sieci o jednakowych oczkach, co usprawiedliwia odrzucenie indeksu k i k' . W równaniu (1) oblicza się t_k metodą iteracyj-

ną po podstawieniu wartości czasów oczekiwania (2). Jednak w przypadku nasycenia tj. gdy istnieje taka linia, że:

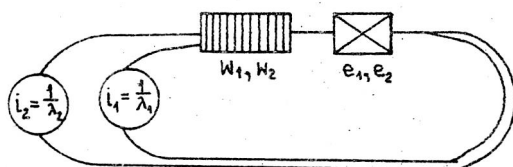
$$\sum_{k \in Z_i} \frac{c_{k,k'} + c_{k',k}}{c_{k,k'} + c_{k',k} + i_k} \geq 1$$

Z_i - zbiór numerów oczek przechodzących przez i -tą linię.

przyjęte przybliżenia prowadzą do dość dużego błędu oszacowania. Szukając nowych przybliżeń analizowano najprostszy przypadek modelu sieci lokalnej. W modelu tym przyjęto, że opóźnienia czasowe wnoszą do oczka tylko podsieć komunikacyjna i to tylko podczas transmisji pakietów informacyjnych, a nie pakietów potwierdzeń. Zatem mamy:

$$\bar{w}_{k'} = \frac{1}{u_k} = w_{k'} = \bar{w}_k = 0 \quad (3)$$

Strukturę rozpatrywanej sieci pokazano na rys. 2.



Rys. 2. Struktura badanej sieci
Fig. 2. A structure study network

Dla takiej sieci przeprowadzono szereg badań symulacyjnych. Część wyników przedstawiono na rys. 3.a-c. Podczas badań ustalono zmienne niezależne jednego oczka i zmieniano czas transmisji w drugim oczku przy ustalonym rozkładzie i wartości oczekiwanej czasu myślenia. Rysunki przedstawiają wartości t_1 w funkcji e_2 obliczone z przybliżenia (1) i znalezione metodą symulacji. Widać, że przy przyjętych założeniach w warunkach nasycenia wartości, zwłaszcza t_1 , obliczone metodą przybliżoną są w istotnym stopniu mniejsze od wartości otrzymanych z symulacji. Przykładowe wyniki uzyskane dla sieci nienasyconej (Rys.3.a) wykazują wysoką dokładność przybliżenia.

2.2. Analiza przepustowości przy transmisji zbiorów

Dla sieci o strukturze pokazanej na rys. 2. można analitycznie wyznaczyć wartości t_1 i t_2 . Zauważmy bowiem, że zgodnie z definicją wartości oczekiwanej t_1 równa się sumie czasu transmisji w oczku numer 1 (interrogacyjne):

$$e_1 + e_2 P(i_1 \geq 0) + e_2 P(i_1 \geq e_2) + e_2 P(i_1 \geq 2e_2) + \dots$$

$$e_k = c_{k,k'}; \quad k = 1, 2$$

Uwzględniając fakt, że zmienna I_1 ma rozkład wykładniczy, mamy:

$$\begin{aligned}
 t_1 &= e_1 + e_2 + e_2 \lambda \int_{e_2}^{\infty} e^{-\lambda x} dx + e_2 \lambda \int_{2e_2}^{\infty} e^{-\lambda x} dx + e_2 \lambda \int_{3e_2}^{\infty} e^{-\lambda x} dx \dots \\
 &= e_1 + e_2 (1 + e^{-\lambda e_2} + e^{-2\lambda e_2} + e^{-3\lambda e_2} + \dots) \\
 &= e_1 + e_2 \frac{1}{1 - e^{-\lambda e_2}} \\
 \lambda &= \frac{1}{i_1}
 \end{aligned} \tag{4}$$

Aby obliczyć t_2 zauważmy, że dla naszej sieci zachodzi:

$$\frac{e_1}{t_1} + \frac{e_2}{t_2} = 1 \tag{5}$$

Dzieje się tak dlatego, że przy transmisji zbiorów ($i_2=0$) podsieć komunikacyjna będzie wykorzystana w 100% (wszelkie rezerwy przepustowości będą natychmiast wykorzystywane na transmisję kolejnych pakietów zbiorów). Tak więc z (4) i (5) mamy:

$$t_2 = e_2 + e_1 (1 - e^{-\lambda e_2}) \tag{6}$$

Wzory (4) i (6) zweryfikowano metodą symulacji. Wyniki dla $e_1=1$, $\lambda=1$ i $\lambda=0.5$, $i_2=0$ i $e_2 = 1, \dots, 10$ przedstawiono w tablicy 1 i 2. Te i inne wyniki potwierdzają poprawność przedstawionej wcześniej metody analitycznej (niewielkie odchylenia wynikają z niedokładności metody symulacyjnej).

Tablica 1.

Odwrotność przepustowości w badanej sieci - wartości obliczone wg (4) i (6) oraz znalezione metodą symulacji.

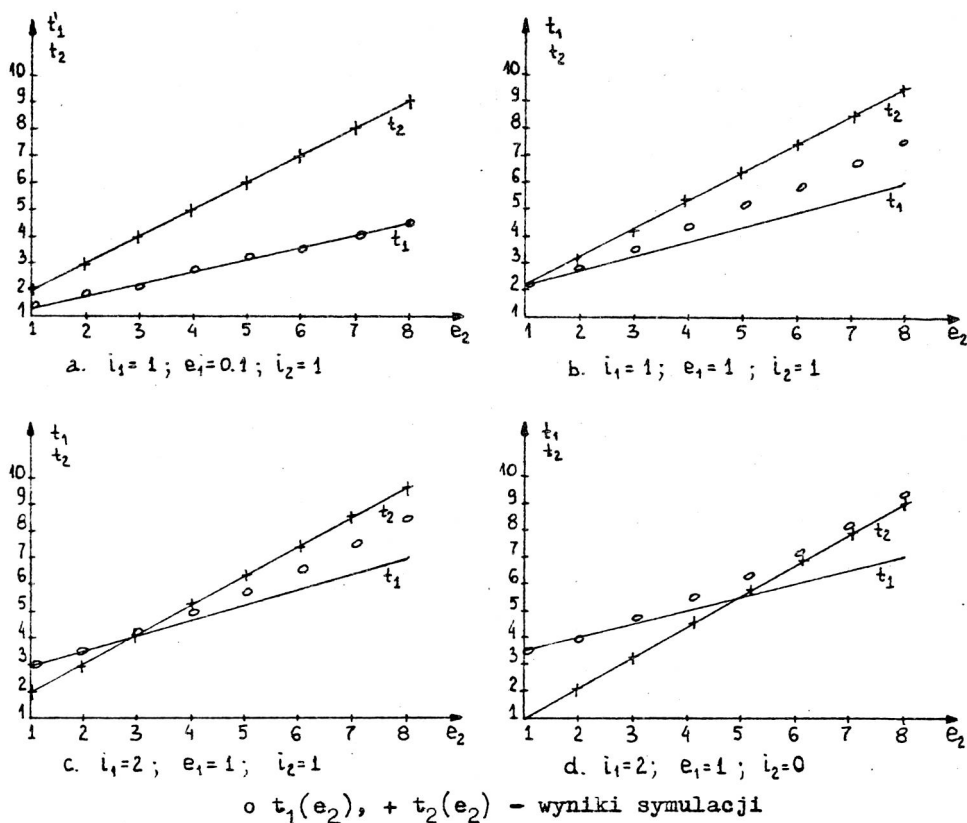
$$\lambda = 1 \text{ (1/s)}, e_1=1, i_2=0$$

e_2	1	2	3	4	5	6	7	8	9	10
t_{1anal}	2.58	3.31	4.16	4.92	6.03	7.01	8.01	9.00	10.00	11.00
t_{1sym}	2.61	3.27	4.14	5.06	6.03	7.03	8.01	9.00	10.00	11.00
t_{2anal}	1.63	2.87	3.95	5.02	5.99	7.00	8.00	9.00	10.00	11.00
t_{2sym}	1.62	2.88	3.96	4.98	5.99	6.99	8.00	9.00	10.00	11.00

Tablica 2.

$$\lambda=0.5, e_1=1, i_2=0$$

e_2	1	2	3	4	5	6	7	8	9	10
t_{1anal}	3.54	4.16	4.86	5.63	6.45	7.31	8.21	9.15	10.10	11.07
t_{1sym}	3.45	3.97	4.93	5.61	6.49	7.35	8.21	9.10	10.13	11.08
t_{2anal}	1.39	2.63	3.78	4.81	5.92	6.95	7.97	8.98	9.99	10.99
t_{2sym}	1.41	2.67	3.77	4.87	5.91	6.95	7.97	8.99	9.99	11.00



Rys. 3. Porównanie wyników analitycznych z wynikami symulacji
Fig. 3. A comparison analytical and simulation results

2.3. Kierunek dalszych prac

Dalsze prace zmierzają do znalezienia przybliżeń (lub rozwiązań dokładnych) dla sieci lokalnych z niezerowymi czasami obsługi i transmisji potwierdzeń dla więcej niż dwóch użytkowników. Dla pewnych przypadków (sieci homogeniczne z wykładniczym myśleniem) można skorzystać z wcześniej uzyskanych wyników np. [7], jednak pełne rozwiązanie zagadnienia nie jest jeszcze znane.

Literatura

- [1] Bicz T. i in., Wstępna ocena opóźnień czasowych przy pracy konwersacyjnej w systemach komputerowych R-32 i Odra 1325, Biuletyn MERA, listopad 1983.
- [2] Lewoc J.B., Misiura L.Z., Rolski T., Metoda oceny przepustowości Cambridge Ring, Przetwarzanie Sygnałów w Telekomunikacji, Sterowaniu i Kontroli, Bydgoszcz 1986.

- [3] Lewoc J.B., Leszek Z. Misiura, Assessment of Round-Trip Delay for Ethernet Local Network, Computer Networks, zgłoszone do druku.
- [4] Lewoc J.B., Misiura L.Z., Analytical performance comparison of Ethernet and Cambridge Ring, Modelling and Simulation, Neapol 1986.
- [5] Lewoc J.B., Misiura L.Z., Porównanie przepustowości sieci lokalnych Ethernet i Cambridge Ring, Infogryf '86, Kókołbrzeg 1986.
- [6] Bicz T., Bieleninik E., Lewoc J.B., Analytical throughput assessment for an X.25 computer network, Modelling and Simulation, Neapol 1986.
- [7] Kopocińska I., Zagadnienie konserwatora, Matematyka Stosowana, Seria III, nr II, 1974.

THROUGHPUT ANALYSIS IN LOCAL AREA NETWORKS WITH FILE TRANSFERS

The paper presents the problem of performance evaluation in local area networks. The case of file transfers and interrogating mode transfers on a simple local area network model is described. The case is of major importance for understanding of behaviour of more complex networks.

ПРОПУСКНАЯ СПОСОБНОСТЬ МЕСТНЫХ СЕТЕЙ С ПЕРЕДАЧЕЙ МНОЖЕСТВ

Из-за отсутствия аналитических методов решения задач о пропускной способности вычислительных систем нужно искать аппроксимационные методы. В докладе представлен аппроксимационный метод вычисления пропускной способности местной сети. После вычислительного моделирования определен класс сетей, в которых этот метод действует удовлетворительно.

sieć lokalna, przepustowość,
analityczna metoda przybliżona

Józef B. LEWOC^x, Leszek Z. MISIURA^{xx}

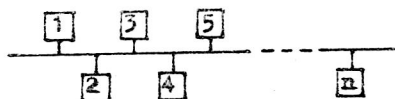
OSZACOWANIE OPÓŹNIENIA PO OCZKU DLA SIECI LOKALNEJ ETHERNET

Wynikiem badań zainspirowanych przez artykuł [1] było opracowanie metody analitycznej dla szacowania przepustowości sieci lokalnych (LAN). Metoda została sprawdzona z wynikiem pozytywnym na przykładzie sieci CSMA/CD (Ethernet) [2] i Empty Slot Time (Cambridge Ring) [3].

Artykuł zawiera skrócony opis metody oraz porównanie wyników przepustowości sieci Ethernet otrzymanych na drodze analitycznej z wynikami zebranymi metodą symulacji publikowanymi w [1].

1. OPIS SIECI

Sieć CSMA/CD (Rys.1.) to pojedyncza szyna, do której podłączonych jest n stacji (węzłów).



Rys. 1. Topologia sieci CSMA/CD
Fig. 1. A network CSMA/CD topology

Strumień wejściowy do każdej stacji jest Poissonowski o średnim czasie pomiędzy zgłoszeniami:

$$\rho = \frac{8 \cdot m \cdot n}{\lambda}$$

m - długość przesyłanych danych (bajt),

λ - intensywność globalna (Mb/s).

Strumienie do indywidualnych stacji są niezależne. Zgłoszenie jest żądaniem transmisji m bajtów danych (ramek, pakietów), a obsługa kończy się, gdy stacja inicjująca odbierze 1-bajtowe potwierdzenie zawarte wewnątrz innej ramki (pakietu). Podczas transmisji może dojść do kolizji i zniszczenia przesyłanych danych. Następuje wówczas retransmisja danych. Kolizja ma miejsce w dwóch przypadkach:

1^o stacja rozpoczyna nadawanie przed odebraniem sygnału informującego,

^x Centrum Obliczeniowe Uniwersytetu Wrocławskiego, Wrocław
^{xx} Centrum Obliczeniowe Politechniki Wrocławskiej, Wrocław

że inna stacja też rozpoczęła nadawanie,

2° podczas transmisji jednej ze stacji co najmniej dwie stacje zgłoszą się po transmisję.

Po kolizji stacja biorąca w niej udział czeka odpowiedni czas (wg algorytmu binary back-off) na dokonanie retransmisji.

2. MODEL ANALITYCZNY

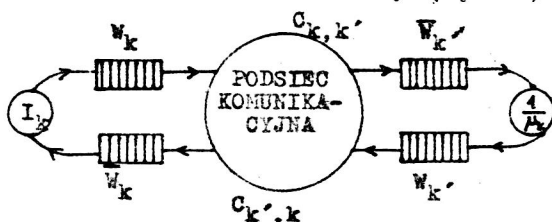
2.1. Założenia modelu

W celu weryfikacji metody przyjęto te same założenia co w artykule [1], a mianowicie:

- ruch w sieci określony zadaniem czasem pomiędzy zgłoszeniami, θ , a nie średnim czasem myślenia, τ ,
- jałowe oczekiwanie stacji inicjującej na potwierdzenie,
- dla jednej stacji w kolejce po transmisję czeka maksymalnie jedno zgłoszenie do danego odbiorcy,
- zerowy czas obsługi pakietu w stacji,
- brak kolejek przed podsiecią,
- pomijalny wpływ błędów transmisji (stopa błędów rzędu 10^{-7}).

2.2. Opis modelu

Model analityczny badania przepustowości LAN opiera się na pojęciu oczek, tworzonych w celu dokonania transmisji (Rys.2.).



Rys. 2. Oczko w sieci komputerowej
Fig. 2. Computer network loop

Oczko składa się z nadawcy k , odbiorcy k' ($k, k' \leq n, k \neq k'$) oraz drogi między nimi. Na pełny czas cyklu (opóźnienia po oczku), T_k , składają się czasy:

- myślenie pomiędzy zgłoszeniami, I_k ,
- oczekiwanie informacji na transmisję, W_k ,
- transmisja informacji, $C_{k,k'}$,
- oczekiwanie odpowiedzi na transmisję, $W_{k'}$,
- transmisja odpowiedzi, $C_{k',k}$.

Zgodnie z przyjętymi założeniami $\bar{W}_k, \bar{W}_{k'}, \frac{1}{\mu_k} = 0$.

Otrzymujemy więc czas cyklu:

$$T_k = I_k + W_k + C_{k,k'} + W_{k'} + C_{k',k}$$

(1)

Na podstawie teorii kolejek i po przejściu na wartości oczekiwane, na podstawie [3] otrzymujemy ostateczną postać wzoru dla przybliżonego szacowania przepustowości:

$$\begin{aligned} t &= 2\bar{t} - c \\ \bar{t} &= w + c \end{aligned} \quad (2)$$

t - średni czas pobytu zgłoszenia w sieci,

$\bar{t}$ - średni czas obsługi zgłoszenia,

w - średni czas oczekiwania na obsługę,

c - średni czas transmisji.

2.2.1. Oszacowanie czasu oczekiwania

Dla policzenia czasu oczekiwania, w , wymagana jest znajomość prawdopodobieństwa zastania ramki w transmisji w chwili, gdy przychodzi zgłoszenie do jednego z o ($1 \leq o \leq n-2$) oczek. Szacujemy je jako:

$$p_o = \frac{o \cdot \bar{t}}{(n-1) \left(\theta - \frac{n-2}{n-1} (\bar{t} + c) \right)} \quad (3)$$

c' - czas transmisji odpowiedzi

Mnożąc to prawdopodobieństwo przez połowę czasu obsługi zgłoszenia, $\frac{\bar{t}}{2}$, otrzymujemy wartość średnią czasu oczekiwania, gdy w kolejce po transmisję czeka jedno zgłoszenie. Ponieważ w kolejce może czekać z zgłoszeń ($2 \leq z \leq n-1$), aby otrzymać czas oczekiwania na zakończenie ich transmisji należy policzyć wg (3) czasy oczekiwania dla kolejnych zgłoszeń w kolejce i pomnożyć przez siebie. We wzorze końcowym na średni czas oczekiwania zgłoszenia na obsługę należy jeszcze uwzględnić oczekiwanie zgłoszenia na zakończenie transmisji z odpowiedzi. Prawdopodobieństwo zastania w kolejce odpowiedzi liczymy wg (3) zastępując w liczniku czas $\bar{t}$ czasem c' . W efekcie wzór na średni czas oczekiwania zgłoszenia na transmisję składa się z czterech członów i ma postać:

$$\begin{aligned} w &= \frac{(n-2)\bar{t}^2}{2M} \left(1 + \frac{(n-3)\bar{t}}{M} \left(1 + \frac{(n-4)\bar{t}}{2M} (\dots) \right) \right) + \frac{(n-2)c}{M} \left(1 + \frac{(n-3)\bar{t}}{2M} (\dots) \right) \\ &\quad \frac{(n-2)c'^2}{2M} \left(1 + \frac{(n-3)\bar{t}}{M} \left(1 + \frac{(n-4)c'}{2M} (\dots) \right) \right) + \frac{(n-3)c'}{M} \left(1 + \frac{(n-4)c'}{2M} (\dots) \right) \\ M &= \frac{1}{(n-1) \left(\theta - \frac{n-2}{n-1} (\bar{t} + c) \right)} \end{aligned}$$

2.2.2. Oszacowanie czasu transmisji

W sieci CSMA/CD przed lub w czasie transmisji może dojść do kolizji co pociąga za sobą odpowiednią karę czasową (o jej wielkości decyduje algorytm binary back-off) i transmisja wydłuża się. Z tego względu całkowity czas transmisji jest sumą czystej transmisji, $e+e'$, oraz kary za wszystkie kolizje, $K+K'$. Otrzymujemy:

$$\begin{aligned} c &= e + e' + K + K' + v + 2d \\ c' &= e' + K' + v + d \end{aligned} \quad (4)$$

v - czas do zakończenia aktualnie trwającej transmisji,
 d - średnia odległość bitowa między transmitującymi stacjami.

Dokładne wzory na poszczególne składniki (4) podano w [2].

2.3. Porównanie wyników analitycznych z symulacyjnymi

Na rys. 3. przedstawiono wyniki otrzymane przy wykorzystaniu modelu analitycznego oraz uzyskane z symulacji [1].

Linia przerywana to wyniki analityczne otrzymane na podstawie nierzezywistych danych - otóż w celu weryfikacji metody przyjęto za autorami z [1] zanizone czasy transmisji e i e' . Porównując otrzymane w ten sposób wyniki można stwierdzić, że przybliżona metoda analizy szacunkowej jest poprawna i daje dobre, a nawet bardzo dobre wyniki (nieznaczne różnice wynikają z błędów symulacji [2,3]).

Znajdujące się na wykresach gwiazdki przedstawiają wartości czasów pobytu zgłoszenia w sieci dla poprawnych wartości e i e' - widać, że czasy te są kilkakrotnie większe (wpływ kolizji).

3. TENDENCJE ROZWOJOWE

W momencie zgłaszania artykułu na konferencję prowadzone są przez autorów prace nad modelem analitycznym dla sieci CSMA/CD i Empty Slot Time w oparciu o nieco inne (stosowane w działających sieciach) założenia. Mianowicie:

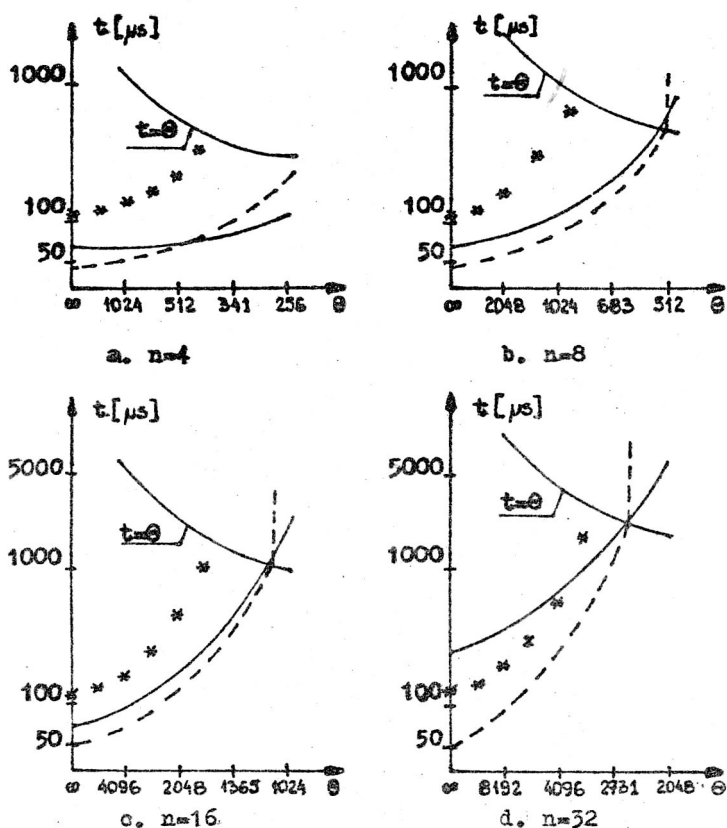
- ruch w sieci określony zadany czasem myślenia pomiędzy zgłoszeniami,
- brak jałowego oczekiwania na potwierdzenie - w tym czasie mogą być transmitowane inne zgłoszenia.

Model będzie uwzględniał również możliwość transmisji zbiorów.

Następnym etapem pracy autorów będzie model analityczny dla pozostałych typów sieci lokalnych tzn. Token Ring i Token Bus.

Literatura

- [1] Blair G.B., Shepherd D., A Performance Comparison of Ethernet and the Cambridge Digital Communication Ring, Computer Networks 6, 1982, North Holland.
- [2] Lewoc J.B., Misiura L.Z., Assessment of Round Trip-Delay for Ethernet Local Area Network, Computer Networks, 1986, zgłoszone do druku.
- [3] Lewoc J.B., Misiura L.Z., Assessment of Round Trip-Delay for Cambridge Ring, Computer Networks, 1986, zgłoszone do druku.



linia ciągła - wyniki z symulacji
 linia przerywana - wyniki analityczne dla złych danych
 gwiazdka - wyniki analityczne dla dobrych danych

Rys. 3. Porównanie wyników analitycznych z symulacyjnymi
 Fig. 3. A comparison analytical and simulation results

ROUND-TRIP DELAY EVALUATION FOR THE LOCAL AREA NETWORK ETHERNET

A method was developed for analytical evaluation of throughput in local networks. The method was tested with positive results on the example of Ethernet and Cambridge Ring. The paper covers comparison of Ethernet evaluation results produced by the analytical method and simulation results published elsewhere.

ОЦЕНИВАНИЕ ПРОПУСКНОЙ СПОСОБНОСТИ МЕСТНОЙ СЕТИ ЭТHERNET

Представлен аналитический метод определения пропускной способности местной сети. Контрольная проверка показывает, что этот метод хорошо работает для сетей CSMA/CD и EST. Сравниваются вычислительные способности этого метода и метода имитационной модели, которая представлена в работе [1].

Teresa LUBIŃSKA*

system roz-
proszonego przetwarzaniaSYSTEM ROZPROSZONY - PROBA PONOWNEGO
PRZYJRZENIA SIĘ ZASOBOM INFORMACYJNYM PRZEDSIĘBIORSTWA

W artykule zdefiniowano system rozproszonego przetwarzania danych. Rozproszoną bazę danych wskazano jako rozwiązanie przyszłościowe. Przyjęto, że dla pomyślnej realizacji systemu rozproszonego dla zastosowań gospodarczych, kapitalne znaczenie ma silna orientacja projektantów na zagadnienia ekonomiczne oraz koncepcja tzw. modelowania semantycznego.

Koncepcja systemów rozproszonego przetwarzania danych zrodziła się w dużych jednostkach gospodarczych, rozproszonych geograficznie, operujących na dużych zasobach danych. Reprezentanci tej koncepcji mają zazwyczaj znaczne doświadczenia w administrowaniu dużymi bazami danych. Zdefiniowanie systemu rozproszonego stwarza szereg trudności. Wynika to z ogromnej różnorodności modeli zaliczanych do systemu rozproszonego. Bliższego wyjaśnienia wymaga korelacja między pojęciami: system rozproszony a decentralizacja przetwarzania danych. Pojęcie decentralizacji na ogół dotyczy sfery zarządzania, gdzie oznacza delegowanie władzy na niższe szczeble. Abstrahując od złożonych zależności stopnia centralizacji i decentralizacji od systemu zarządzania, można w uproszczeniu podać, że w małych przedsiębiorstwach z reguły preferowana jest centralizacja decyzji. W miarę wzrostu stopnia złożoności jednostki gospodarczej pojawia się tendencja delegowania władzy na niższe szczeble, szczególnie w przypadku zróżnicowanej działalności gospodarczej. Decentralizacja w przetwarzaniu danych dotyczy również uprawnień decyzyjnych, realizowanych przecież w ramach ogólnego zarządzania jednostką gospodarczą, związanych z problematyką przetwarzania danych. Uprawnienia decyzyjne odnoszą się tu przede wszystkim do wyboru obszaru zastosowań systemów użytkowych oraz metod ich projektowania, co możnaby określić jako politykę informatyczną. Decentralizacja przetwarzania danych związana jest z tworzeniem w określonych oddziałach, wydziałach, zakładach dużych jednostek gospodarczych, zespołów profesjonalistów z zakresu przetwarzania da-

* Uniwersytet Szczeciński

nych, tzw. personelu informatycznego. W przypadku gdy poszczególne zespoły informatyczne jednostki gospodarczej będą posiadały znaczną samodzielność decyzji w polityce informatycznej, to ma miejsce decentralizacja przetwarzania danych. Zespoły te mają swobodę wyboru funkcji przeznaczonych do skomputeryzowania i samodzielnie decydują jak użyć potencjału informatycznego. Na ogół delegowaniu władzy informatycznej towarzyszy rozproszenie sprzętu informatycznego. Podkreślić należy, że samo rozproszenie potencjału sprzętowego bez delegowania uprawnień w zakresie przetwarzania danych nie stanowi systemu zdecentralizowanego. W takim przypadku bardziej wskazane byłoby określenie sytuacji jako *d e k o n c e n t r a c j a s p r z ę t u i n f o r m a t y c z n e g o*, co uwytknęła fizyczny charakter tego zjawiska. Oczywiście, że dekoncentracja mocy obliczeniowych sprzyja procesowi decentralizacji przetwarzania danych. Trudno jednak precyzyjnie określić jaki jest związek między stopniem decentralizacji a stopniem dekoncentracji sprzętu. Dodać jeszcze należy, że decentralizacja nie oznacza całkowitej autonomii poszczególnych oddziałów informatycznych, obowiązuje przede wszystkim konieczność działania na jednolitej bazie kodów, nomenklatur, norm.

Wspomniana dekoncentracja sprzętu, dotyczy to głównie procesorów, jest zjawiskiem coraz częstszym w praktyce gospodarczej. Zjawisku temu na ogół towarzyszy rozproszenie zasobów danych. Uzyskujemy wówczas rozwiązania stanowiące właśnie system rozproszony. W żadnej definicji *s y s t e m u r o z p r o s z o n e g o* nie podaje się wyraźnie wymogu delegowania uprawnień decyzyjnych. Podkreśla się raczej *f a k t d e k o n c e n t r a c j i s p r z ę t u - p r o c e s o r ó w - z d o l n y c h d o s a m o d z i e l n e j p r a c y i z a - s o b ó w d a n y c h*. Decentralizacja w informatyce, podobnie jak w zarządzaniu, to proces długotrwały, trudny, kryjący wiele niebezpieczeństw. Związany jest on z takimi problemami jak: postawy ludzkie, sposoby myślenia, doświadczenie.

System rozproszonego przetwarzania danych może przyjąć bardzo różnicowane konfiguracje. Praktyka gospodarcza wskazuje na to, że obecnie podstawowym, możliwym do realizacji typem systemu rozproszonego jest zdalne przetwarzanie partiiowe. Rozproszona baza danych stanowi jeszcze formy przyszłościowe. W praktyce gospodarczej nie funkcjonują bowiem rozproszone bazy danych dysponujące Sieciowymi Systemami Zarządzania Bazą Danych, spełniającymi wszystkie wymagania odnośnie współzarządzania zasobami /dane, sprzęt, oprogramowanie/. W wielu informatycznych ośrodkach naukowo-badawczych prowadzi się badania nad rozwiązaniami w zakresie rozproszonej bazy danych. W większości badań stosowany jest mo-

del relacyjny bazy danych i związany z nim język manipulacji danymi [5, s. 19]. Obecnie oferowane projekty rozproszonych baz danych nie są w pełni, automatycznie zintegrowane. Charakteryzują się one węzłami posiadającymi bazy oprogramowane w określonych Systemach Zarządzania Bazą Danych. Węzły te są co prawda połączone łączem telekomunikacyjnym, ale brak tu automatycznej obsługi współdziałania baz między sobą. Czasami wymagają one od użytkowników specyfikowania miejsc przetwarzania i przechowywania danych. Nie spełniony jest tu warunek niezależności danych od rozproszenia geograficznego. Zwykle zaprojektowane są one na bazie sieci o jednorodnym środowisku komputerowym i programowym. Z reguły procesory nie współpracują w trybie dialogowym lecz w trybie partiiowym. Oczekuje się, że odpowiednie Sieciowe Systemy Zarządzania Bazą Danych będą dostępne w praktyce gospodarczej w drugiej połowie lat 80-tych [5, s. 14].

Systemy rozproszone stawiają bardzo wysokie wymagania sprzętowo-programowe. Prawdopodobnie z tej przyczyny większość opracowań z zakresu rozproszonego przetwarzania danych dotyczy problemów sprzętowo-programowych, nadając tym samym priorytet tym zagadnieniom.

System informacyjny użytkownika pozostaje "obok" lub "w tle". Wydaje się, że zagadnienia dotyczące systemu informacyjnego użytkownika powinny mieć podstawowe znaczenie, a przynajmniej równorzędne jak zagadnienia sprzętowo-programowe. Postęp w dziedzinie programowo-sprzętowej znacznie wyprzedził postęp w obszarze analizy systemu informacyjnego użytkownika. Baczniejszy wgląd w dotychczasową literaturę przedmiotu pozwala sądzić, że siła systemów rozproszonych będzie następstwem rygorystycznie przeprowadzonej analizy systemu informacyjnego użytkownika, a tym samym coraz głębszej orientacji środowiska informatyków na zagadnienia gospodarcze [1]. Pamiętając o tym, że każda jednostka posiada swojego rodzaju specyfikę, filozofię zarządzania, niemożliwością jest sformułowanie ścisłych powtarzalnych reguł analizy rozproszenia strumieni informacyjnych. Korzystać tu trzeba z najnowszych osiągnięć metodologii projektowania systemów informatycznych w warunkach bazy danych [6] oraz z tzw. selekcyjnej metodyki projektowania systemów [4].

Zwrócić należy też uwagę na kapitalne znaczenie koncepcji tzw. modelowania semantycznego [2 i 3]. Podejście to zmierza do ujmowania systemu informacyjnego jako modelu rzeczywistości gospodarczej tzn. silnego wiązania go ze sferą realną. Koncepcja ta uniezależnia w pewnym sensie system informacyjny od trudno definiowanych potrzeb informacyjnych użytkownika oraz od zmian w procedurach i metodach zarządzania. Modelowanie to wymaga od projektanta systemu znacznej wiedzy o podstawowych celach i funkcjach przedsiębiorstwa oraz ich wzajemnych warunkowaniach.

Literatura

- [1] Bratkovich R.N.: Distributed Computing for Engineering Drawing System "Data Management" 1981, nr 6
- [2] Brodie M.L. et al. "On Conceptual Modelling". Springer-Verlag 1984
- [3] Kent W.: Data and Reality. Basic Assumptions in Data Processing. North Holland 1978
- [4] Maćkowiak K.: Metodyka projektowania systemów rozproszonego przetwarzania danych w porównaniu z metodami tradycyjnymi. PN AE Wrocław 1982, nr 213
- [5] Nowości w dziedzinie baz danych. EPBD 1983, nr 133
- [6] Oleński J., Staniszkis W.: Projektowanie bazy danych. PWE, Warszawa 1984 r.

The Distributed System - an Attempt to Re-examine
the Information Resources of the enterprise.

In this article distributed data processing has been defined. The distributed data base has been indicated as a future solution. It has been assumed that it is essential for designers to have a new orientation in economic problems, and concept of so - called semantic modelling in order to realize a successful distributed system in economic applications.

ДИСПЕРСНАЯ СИСТЕМА - ПОПЫТКА ПОВТОРНОГО ПРОСМОТРА ИНФОРМАЦИОННЫХ РЕСУРСОВ ПРЕДПРИЯТИЯ

В статье дана дефиниция дисперсной системы обработки данных. Дисперсный базис данных указан как перспективное решение. Принято, что для успешной реализации дисперсной системы для применения в народном хозяйстве капитальное значение имеет само ориентирование проектантов на экономические вопросы, а также концепция т.н. семантического моделирования.

Stanisław Płaczek^x

sieci komputerowe
oprogramowanie

LOKALNE MONITORY EKRANOWE W SIECI MSK

Oprogramowanie systemowe procesora czołowego w sieci MSK umożliwia wykorzystanie lokalnych monitorów ekranowych w otwartej sieci komputerowej. Użytkownicy monitorów MERA 7911 są odcięci od zasobów programowych dostępnych w sieci. Zastosowanie w systemie G-3 Virtualnego Procesora Komunikacyjnego oraz programu Managera Komunikacyjnego daje możliwość rozwiązania problemu. Programy współpracujące z Managerem Komunikacyjnym realizują interfejs dostępu do usług transportowych komputera czołowego, a tym samym zapewniają dostęp do oddalonych komputerów obliczeniowych.

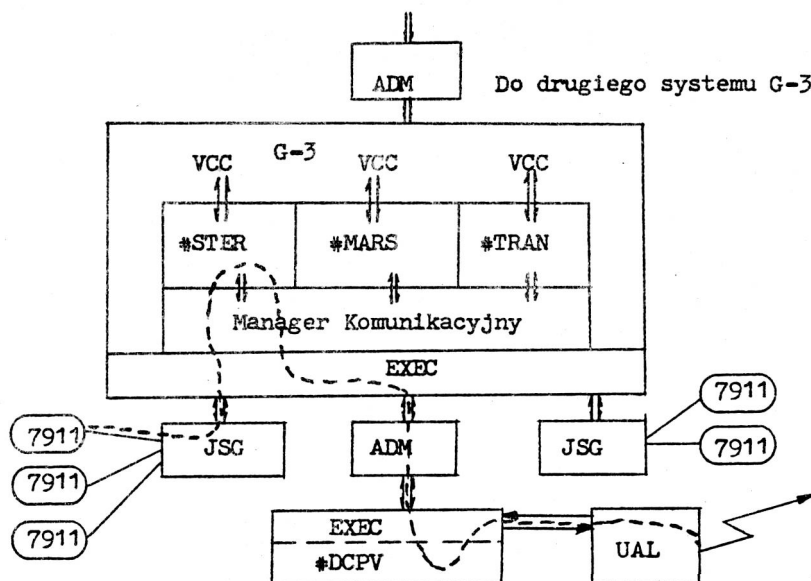
1. Wstęp

Istniejące oprogramowanie systemowe procesora czołowego obsługuje tylko monitory dalekopisowe (TTY) podłączone do Odry 1325 poprzez multiplexer MPX-325. Brak w programie #DCPV modułu obsługi grupowych jednostek sterowania monitorami ekranowymi (JSG), umożliwia wykorzystanie monitorów MERA 7911 w otwartej sieci komputerowej. Tak więc praktycznie biorąc tylko tzw. zdalni użytkownicy systemów komputerowych, wyposażeni w drukarki mozaikowe DZM180KSRE lub monitory MERA 7952N realizujące protokół w/w drukarek, mogą być użytkownikami sieci MSK. Użytkownicy podłączeni do systemu GEORGE 3 poprzez JSG i lokalne monitory ekranowe aktualnie nie mogą korzystać z usług sieci MSK. Jest to potencjalnie liczna grupa użytkowników. W niektórych ośrodkach na lokalnych monitorach ekranowych oparta jest cała dydaktyka. Z tego punktu widzenia, dopuszczenie użytkowników lokalnych monitorów ekranowych do sieci MSK jest bardzo ważne.

2. Struktura systemu.

W standardowych instalacjach grupowe jednostki sterujące monito-

^x Ośrodek Elektronicznej Techniki Obliczeniowej Politechniki Śl. Gliwice



Rys.1. Schemat komunikacji.

Fig.1. Communication diagram.

rami ekranowanymi (JSG 7802) obsługiwane są poprzez makro VIDIMOP z programem #XKL4. Dostęp do systemu G-3 realizuje się poprzez kanał wydawcy komend. Rozwiązanie to uniemożliwia wykorzystanie całoe ekranowego edytora. Zainstalowanie w sytemie G-3 Managera Komunikacyjnego obsługującego wszystkie jednostki komunikacyjne (JSG, MPX, ICL 7903) umożliwia zrealizowanie nast. celów (rys.1.):

- a) lokalne monitory ekranowe MERA 7911 realizują dostęp do systemu G-3 poprzez Virtualny Procesor Komunikacyjny (VCC) oraz pomocniczy program #MARS. Lokalni użytkownicy mogą korzystać z całostronicowego edytora (SCREEN EDIT),
- b) procesor czołowy podłączony do Odry 1305 poprzez ADM-325 jest również obsługiwany przez program Managera Komunikacyjnego, który realizuje interfejs międzymaszynowy ICL 7900. Program #STER, komunikujący się z Managerem Komunikacyjnym poprzez łącza programowe wewnątrzmaszynowe, umożliwia sterowanie identyfikatorami poprzez podawanie numerów logicznych dla poszczególnych urządzeń końcowych systemu teletransmisji.

Przydzielenie numerów poszczególnym urządzeniom sieci następuje w czasie kompilacji programu Managera Komunikacyjnego. Manager Komunikacyjny jak i wszystkie programy użytkowe używają identycznych iden-

tyfikatorów w stosunku do tych samych urządzeń.

Z programem Managera Komunikacyjnego może współpracować do 10 programów użytkowych. Pojedyncze terminale sieci mogą być przydzielone różnym programom, a także przełączane pomiędzy nimi. Standardowo monitory lokalne MERA 7911 przyłączone są do programu #MARS i współpracują z G-3 poprzez VCC. Użytkownik może przydzielić swój terminal do innego programu np. #STER.

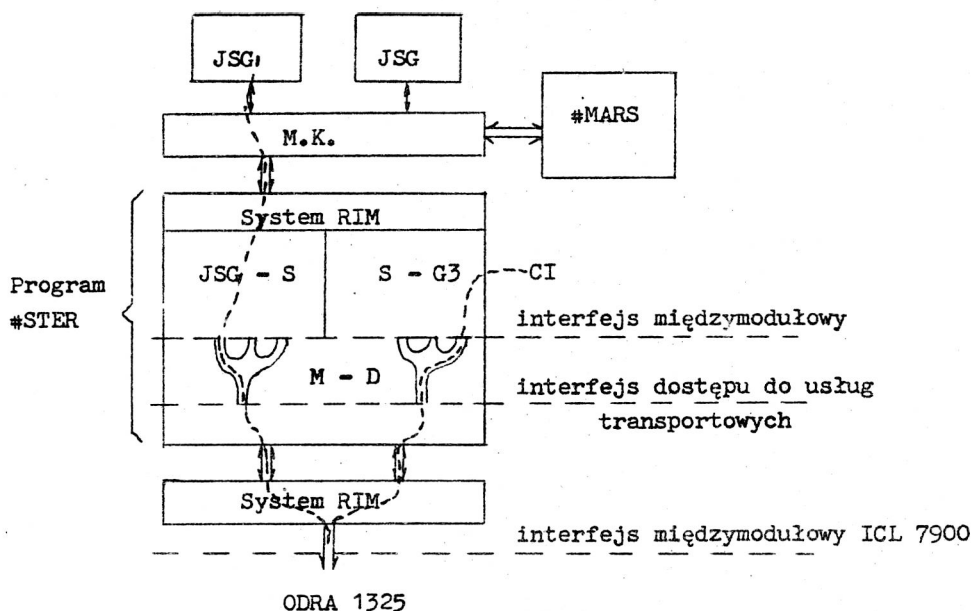
Trzeci wyszczególniony program #TRAN umożliwia logiczne przełączenie lokalnych monitorów pomiędzy dwoma systemami G-3 sprzężonymi drugim adapterem międzymaszynowym ADM-325. Realizowany jest również transfer zbiorów pomiędzy kartotekami w dwóch różnych systemach (zbiory graficzne, binarne, taśmy magnetyczne).

3. Oprogramowanie komunikacyjne

Interfejs dostępu do usług transportowych komputera czołowego (KC-Odra 1325) określa reguły współpracy pomiędzy użytkownikiem warstwy transportowej, rezydującym w systemie G-3, a punktem dostępu do usług transportowych stacji transportowej, implementowanej w KC. Interfejs ten realizuje program #STER, współpracujący poprzez łącza programowe wewnątrzmaszynowe z programem Managera Komunikacyjnego. Manager Komunikacyjny realizuje standardowy interfejs międzymaszynowy ICL 7900 w którym zanurzony jest w/w interfejs dostępu do usług transportowych (rys.2.).

Program #STER składa się z następujących modułów:

- a) sytem RIM, który poprzez łącza programowe wewnątrzmaszynowe realizuje wymianę informacji pomiędzy programem Managera Komunikacyjnego a modułami programu użytkowego,
- b) JSG-S (jednostka sterowania grupowego - sieć), który poprzez RIM przyjmuje/wysyła informację z/do lokalnego monitora adresowanego poprzez unikalny numer identyfikatora. Dzieli otrzymaną informację na ciąg segmentów i poprzez interfejs międzymodułowy przekazuje do modułu M-D. Organizuje synchronizację procesów realizowanych w oddalonych procesorach, poprzez właściwą obsługę przerw (break),
- c) S-G3 (sieć - system GEORGE 3), który poprzez kanał wydawcy komend przyjmuje/wysyła informację z/do systemu G-3. Dzieli otrzymaną informację na ciąg segmentów i poprzez interfejs międzymodułowy przekazuje do modułu M-D. Organizuje również synchronizację oddalonych procesów,
- d) M-D (multipleksowanie - demultipleksowanie). Jest to najważniejszy moduł programu. Realizuje inicjację całego programu, a przede



Rys.2. Struktura programu.

Fig.2. Program structure.

wszystkim odpowiada za inicjację interfejsu dostępu do usług transportowych w dwóch trybach [1],

- tryb pracy inicjatora, w którym inicjatorem żądającym usługi ustanowienia połączenia transportowego jest moduł JSG - S,
- tryb pracy odbiorcy, w którym moduł S - G3 oczekuje na zgłoszenie od partnera z komputera czołowego wskazujące żądanie ustanowienia połączenia transportowego.

Moduł po ustanowieniu połączenia transportowego, przechodzi do fazy transferu danych. W celu uzyskania całkowitej symetrii usług w komputerach obliczeniowych, wykorzystuje się dwa identyfikatory punktów dostępu do usług transportowych:

- praca inicjatora (identyfikator output),
- praca odbiorcy (identyfikator input).

Moduł M - D przeprowadza multipleksowanie w górę i w dół segmentów informacji (danych użytkowych).

4. Podsumowanie

Struktura systemu jest przystosowana do wykorzystania połączenia

transportowego pomiędzy dwoma komputerami obliczeniowymi Odra 1305 współpracującymi poprzez sieć MSK. System ten umożliwia wzajemne wykorzystanie lokalnych monitorów ekranowych w sieci. W przypadku większej ilości hostów należy zrealizować podwójne połączenia transportowe pomiędzy poszczególnymi stacjami transportowymi. Przebudować logicznie należy, przede wszystkim, moduł M - D, który odpowiada za multipleksowanie i demultipleksowanie informacji dla poszczególnych identyfikatorów. Moduł JSC - S musi dopuszczać adresację, poprzez interfejs międzymodułowy, większej ilości identyfikatorów. Aktualnie dla przedstawionego rozwiązania prowadzone są prace projektowo - programowe.

5. Literatura

- [1] Wojciech Żabniewski : Specyfika interfejsu dostępu do usług transportowych komputera czołowego Odra 1325 sieci komputerowej MSK.
- [2] Manager Komunikacyjny : Opracowanie Mera Elwro.

LOCAL MONITOR DISPLAYS IN MSK NETWORK

System software of Front End Processor (FEP - Odra 1325) does not permit to use of local monitor displays in Wide Area Networks (WAN). Users of MERA 7911 monitor displays don't have any access to resources available in network. Using a Virtual Communications Processor under GEORGE-3 and Communications Manager program resolves this problem. Programs co-operating with Communications Manager enable access to transport services of FEP and therefore make usage of remote computers (HOST-s) possible.

Локальные дисплеи в компьютерной сети MSK

Системное программное обеспечение сетевого процессора не даёт возможности использования локальных дисплеев в открытой компьютерной сети. Пользователи дисплеев МЕРА 7911 не могут использовать программные ресурсы доступные в сети. Применение в системе Г-3 Виртуального Телепроцессора, а также программы управления телепередачи даёт возможность решения этой проблемы. Программы, связанные с программой управления телепередачи, реализуют интерфейс доступа к транспортным услугам сетевого процессора, т.е. реализуют доступ к удалённым Главным Вычислительным Машинам (ГВМ).

Tomasz Walasek *

Krzysztof Zielinski *

lokalna sieć komputerowa,
protokół transportowy,
testowanie

KONCEPCJA OCENY POPRAWNOŚCI FUNKCJONALNEJ I EFEKTYWNOŚCI PODSYSTEMU TRANSMISJI SIECI UMMLAN-2

Przedstawiono koncepcję organizacji badań podsystemu transmisji lokalnej sieci komputerowej UMMLAN-2 skonstruowanej w Instytucie Informatyki AGH. Eksperymenty mają na celu przede wszystkim ocenę poprawności i efektywności programowej implementacji protokołu transportowego. Pomiarów wielkości związanych z komunikacją prowadzi się z wykorzystaniem dedykowanej stacji monitorującej i lokalnych programów obserwujących zdarzenia komunikacyjne.

1. WSTĘP

Opisywane w literaturze [1,2,3,6,7] nowe prace eksperymentalne mające na celu ocenę poprawności funkcjonalnej (czyli walidację) i efektywności lokalnych sieci komputerowych dotyczą zasadniczo trzech poziomów:

- warstwy fizycznej i sterowania dostępem do medium,
- oprogramowania warstwy transportu,
- komunikacji między procesami w systemach rozproszonych.

Niezależnie od poziomu, którego dotyczy badanie, stosuje się następujące środki dla organizacji eksperymentu:

- a) sprzęt i oprogramowanie dla generacji określonych warunków pracy sieci,
- b) środki dla obserwacji i rejestracji pracy sieci,
- c) programy do analizy danych zebranych w procesie monitorowania, pozwalające na uzyskanie syntetycznych wskaźników jakości badanego systemu.

Opracowując koncepcję organizacji badań podsystemu transmisji lokalnej sieci komputerowej UMMLAN-2, skonstruowanej w Instytucie Informatyki AGH, wykorzystano aktualne tendencje w zakresie metodyki prowadzenia eksperymentów w takich systemach.

* Instytut Informatyki AGH, Kraków

2. CEL BADAN EKSPERYMENTALNYCH PODSYSTEMU TRANSMISJI

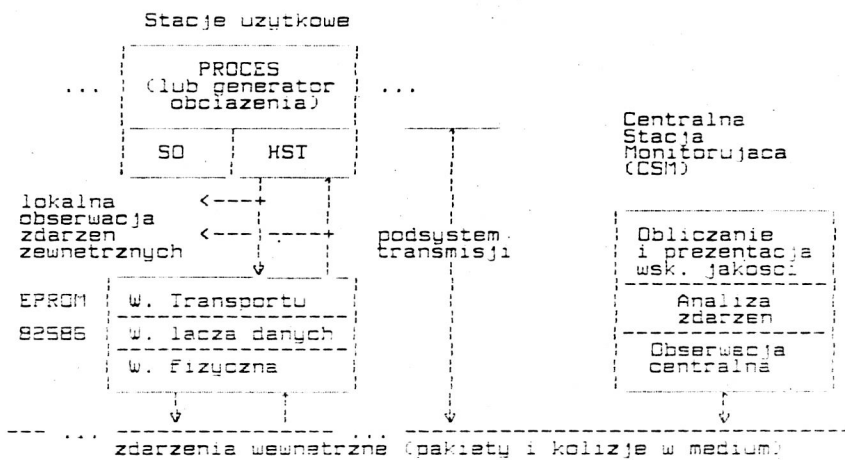
Podsystem transmisji sieci UMMLAN-2 [5] zapewnia przezroczyste, niezawodne przesyłanie wiadomości (z ewentualnym potwierdzeniem) między procesami. Jest to podsystem odpowiadający wymaganiom standardu IEEE-802.3. Logiczna struktura stacji w sieci UMMLAN-2 przedstawiono na rys.1. Podstawowe cechy przyjętego interfejsu między komputerem obliczeniowym a siecią są następujące:

- funkcje warstwy fizycznej i częściowo warstwy łącza danych realizuje układ VLSI (koprocessor Intel 82586),
- oprogramowanie transportowe umieszczono w pamięci EPROM dedykowanego w każdej stacji komputera komunikacyjnego,
- operacje transportowe są udostępniane procesom użytkowym przez handler stacji transportowej (HST) implementowany w systemie operacyjnym komputera obliczeniowego.

Eksperymentalne badania podsystemu transmisji mają na celu walidację podsystemu (w warunkach sztucznego obciążenia), ocenę efektywności pod kątem wymagań aplikacyjnych (przy obciążeniu naturalnym i sztucznym) oraz śledzenie operacji komunikacji międzyprocesowej (w warunkach eksploatacyjnych, przy obciążeniu naturalnym).

3. KONCEPCJA EKSPERYMENTU

Przyjęto zasadę, że pomiaru wielkości związanych z komunikacją należy dokonać w sposób nie wnoszący dodatkowego obciążenia medium transmisji i przy maksymalnej koncentracji danych pomiarowych. System pomiarowy (por. rys. 1) rejestruje wejściowe obciążenie podsystemu transmisji w każdej stacji oraz zdarzenia zachodzące w medium.



Rys. 1. Środowisko i organizacja eksperymentu
Fig. 1. Environment and organization of the experiment

Wykorzystują się przy tym specyfika topologii szynowej i rozgłoszeniowy charakter transmisji w sieci. Istnieje również możliwość wytworzenia sztucznego obciążenia za pomocą programów - generatorów o zadanych parametrach. Eksperyment obejmuje dwie fazy:

- I. Faza obserwacji i rejestracji zdarzeń wewnętrznych i zewnętrznych.
- II. Faza analizy zdarzeń i obliczania miar jakości.

Zdarzenia wewnętrzne (wszystkie transmisje w medium) są rejestrowane przez specjalizowaną stację sieci zwana Centralna Stacja Monitorująca CSM. Zbiór zdarzeń wewnętrznych (por. tablica 1) obejmuje:

DIG: transmisja pakietu typu Datagram,
 NPI: transmisja Numerowanego Pakietu Informacyjnego,
 ACK: transmisja pakietu potwierdzenia,
 CER: transmisja pakietu z błędna suma CRC,
 AER: kolizja.

Tablica 1. Atrybuty obserwowalnych zdarzeń wewnętrznych
 Table 1. Attributes of observable internal events

Zródło atrybutu:	Pakiet w medium								CSM
Atrybut:	S_zr	P_zr	S_dc	P_dc	Typ	Diug	NS	NSP	Czas
DIG	+	+	+	+			+		+
NPI									+
ACK	+	+	+	+		+		+	+
CER									+
AER									+

Objasnienia:

S_zr, P_zr: adres źródłowy (stacji i portu),
 S_dc, P_dc: adres docelowy (stacji i portu),
 Diug: długość pakietu,
 NS: numer sekwencyjny pakietu w ramach wiadomości,
 NSP: numer pakietu w ramach połączenia transportowego,
 Typ: typ potwierdzenia
 (pozytywne lub negatywne z podaniem przyczyny),
 Czas: czas zarejestrowania zdarzenia (czas lokalny CSM).

W poszczególnych stacjach sieci rejestrowane są zdarzenia zewnętrzne (wystąpienia zleceń dla warstwy transportu i odpowiedzi na zlecenia). Rejestracji dokonuje lokalny obserwator - procedura działająca w strukturze handlera HST. Obserwowane zlecenia (z) i odpowiedzi (o) (por. tablica 2) związane są z następującymi operacjami transportowymi:

ALLOCATE: przydziel port,
 DEALLOCATE: usun port,
 TEST_PORT: testuj porty,
 SEND_DIG: wyślij wiadomość w trybie bezpołączeniowym,
 SEND_NPI: wyślij wiadomość w trybie połączeniowym,
 TEST_SND: testuj stan operacji SEND_NPI,
 SIGNAL_SND: sygnalizuj zakończenie operacji SEND_NPI,
 WAIT_SND: zawies proces w oczekiwaniu na zakończenie SEND_NPI,
 RECEIVE: odbierz wiadomość,
 PREVIEW: sprawdź pakiet oczekujący na odebranie.

SIGNAL_RCU: sygnalizuj przybycie pakietu do portu,
 WAIT_RCU: zawies proces w oczekiwaniu na przybycie pakietu,
 REMOVE: usun pakiet oczekujacy na odebranie.

Tablica 2. Atrybuty obserwowalnych zdarzen zewnetrznych
 Table 2. Attributes of observable external events

Zrodlo atrybutu:	Pole operacji w obszarze procesu						lokalny obserwator	
	Atrybut:	P_Zr	S_do	P_do	Midg	NS	Stan	S_zr Czas
ALCT	(Z)	+	+	+				+
DLCT	(O)	+					+	+
TEST_P	(Z)	+					+	+
SEND_D	(Z)	+	+	+	+		+	+
SEND_N	(Z)	+	+	+	+		+	+
TEST_S	(Z)	+					+	+
SGNL_S	(Z)	+	+	+	+	+	+	+
WAIT_S	(Z)	+					+	+
RCU	(Z)	+	+	+	+	+	+	+
PRU	(Z)	+	+	+	+	+	+	+
SGNL_R	(Z)	+	+	+	+	+	+	+
WAIT_R	(Z)	+					+	+
REMOVE	(Z)	+	+	+	+	+	+	+
	(O)		+	+	+	+	+	+

Objasnienia:

Stan: kod statusu operacji (przyjeta, wykonana lub odrzucona).
 Czas: czas zarejestrowania zdarzenia (czas lokalny obserwatora).

Po zakonczeniu realizacji fazy I eksperymentu, wyniki rejestracji lokalnych zostaja przeslane do CSM a nastepnie poddane filtracji (redukcji) i analizie. Miary jakosci sa obliczane na podstawie wartosci atrybutow zdarzen oraz powiazan przyczynowo - skutkowych miedzy zdarzeniami. Zbiór zarejestrowanych danych obejmuje kompletne wartosci i wyjscie (zarowno od strony procesu jak i od strony medium) kazdego komputera komunikacyjnego w podsystemie transmisji. Daje to bardzo szczegolowy obraz komunikacji w sieci, wystarczajacy dla realizacji kazdego z celow postawionych w rozdziale 2.

Zamierzona konsekwencja przyjeciej metodyki pomiaru jest brak wspolnej skali czasu dla wszystkich zdarzen. Zdarzenia zewnetrzne maja przypisany czas lokalny odpowiedniego obserwatora a zdarzenia wewnetrzne - czas lokalny CSM. Takie rozwiazanie wprawdzie komplikuje analize przyczynowa (np. lancuchy zdarzen DTG/RECEIVE_z/RECEIVE_o i RECEIVE_z/DTG/RECEIVE_o nie beda rozrozniane na etapie analize przyczynowej), ale nie ogranicza to ani mozliwosci badan walidacyjnych ani badan efektywnosci. Unikniecie znacznego narzutu transmisji w medium, ktory bylby zwiazany z implementacja ktorejkolwiek z proponowanych metod synchronizacji czasu w systemie rozproszonym (np. [40]), jest bardzo wazne w przyjeciej koncepcji neutralnego pomiaru.

4. WNIOSKI

Przedstawiony sposób organizacji eksperymentu umożliwia kompleksowe badanie podsystemu transmisji przy zachowaniu następujących korzystnych cech systemu pomiarowego:

- a) Neutralność. Pomiar nie wnosi dodatkowego obciążenia medium.
- b) Przezroczystość. Pomiar nie narusza normalnych funkcji sieci i nie wymaga adaptacji programów użytkownika.
- c) Brak modyfikacji oprogramowania transportowego w komputerach komunikacyjnych.
- d) Centralizacja zbierania wyników obserwacji zdarzeń.
- e) Łatwość parametrycznego sterowania obciążeniem zewnętrznym.

Cena, jaka zapłacono za wymienione zalety była konieczność wydzielenia osobnej stacji sieci (CSM) dla potrzeb eksperymentu. CSM wymaga zmodyfikowanego oprogramowania w swoim komputerze komunikacyjnym oraz złożonych programów dla analizy zdarzeń.

5. LITERATURA

- [1] Cabrera L.F. et al., A User-Process Oriented Performance Study of Ethernet Networking Under Berkeley UNIX 4.2BSD. University of California, Berkeley, 1984.
- [2] Feather F., Siewiorek D., Segall Z., Validation of a Fault-Tolerant Multiprocessor: Synthetic Workload Implementation. Carnegie-Mellon University, Pittsburgh, 1985.
- [3] Franta W.R., Heath J.R., Measurement and Analysis of HYPERchannel Networks. IEEE Trans. on Computers, vol. C-33, no. 3, March 1984, pp. 249-260.
- [4] Lamport L., Time, clocks and the ordering of events in a distributed system. Commun. Ass. Comput. Mach., vol 21, July 1978, pp. 558-565.
- [5] Indulska J. i in., The impact of VLSI technology on communications software for local area networks - practical consideration of the UMMLAN-2 project. International Conference Systems Science IX, Wrocław, September 1986.
- [6] Miller B.P., DPM: a Measurement System for Distributed Programs. Computer Sciences Technical Report #592, University of Wisconsin-Madison, April 1985.
- [7] Watson W.B., Performance Monitoring a Large, High-Data-Rate Local Network. Lawrence Livermore National Laboratory, University of California, August 1985, UCRL-5366.

THE CONCEPT OF THE UMMLAN-2 VALIDATION AND PERFORMANCE MEASUREMENT

The report presents the concept of the UMMLAN-2 local area network validation and performance measurement. There has been described an experimental environment in which dedicated monitor station and local observatory software register communication events in the medium and in the process interface respectively. The collected data can later be analysed in order to calculate useful performance statistics and to check the conformity of the transport protocol to its specification.

КОНЦЕПЦИЯ ОЦЕНКИ ФУНКЦИОНАЛЬНОЙ ВЕРНОСТИ И ЭФФЕКТИВНОСТИ ПОДСИСТЕМЫ ТРАНСМИССИ СЕТИ UMMLAN-2

Здесь дается концепция организации исследований подсистемы передачи данных локальной сети UMMLAN-2. Эксперименты проводятся для оценки правильности и эффективности программной имплементации транспортного протокола. Измерения ведутся при использовании отдельной мониторирующей станции и местных программ, следящих за коммуникационными событиями.

Józef WOZNIAK*

radiowe sieci teleinformatyczne,
swobodny dostęp,
informacje kontrolne,
przepustowość kanałuWPŁYW INFORMACJI KONTROLNYCH NA RZECZYWISTE
WYKORZYSTANIE KANAŁU W RADIOWEJ SIECI TELEINFORMATYCZNEJ

W większości prac poświęconych analizie przepustowości kanałów w sieciach radiowych pomija się wpływ informacji kontrolnych, tj. pakietów potwierdzeń na rzeczywiste wykorzystanie kanału. W pracy przedstawiono zależności pomiędzy średnim rzeczywistym przepływem pakietów /przepustowością kanału/ a podstawowymi parametrami charakteryzującymi sieć jednokokową o pełnej lub gwiazdowej strukturze. Analizie poddano dwa podstawowe protokoły swobodnego dostępu - tzw. asynchroniczny i synchroniczny protokół typu ALOHA - z uwzględnieniem kilku ich wersji

1. WPROWADZENIE

Komutacja pakietów połączona z przypadkowym - swobodnym dostępem użytkowników do kanału radiowego została zastosowana po raz pierwszy przed 15 laty w sieci ALOHA [1], [4], [5], [6]. Od tego czasu datuje się stałe, ciągle rosnące zainteresowanie wykorzystaniem łączności radiowej do wymiany informacji pomiędzy cyfrowymi urządzeniami końcowymi. Należy przy tym zaznaczyć, że protokoły swobodnego dostępu znajdują również zastosowanie w sieciach z kablami współosiowymi czy też światłowodami. W literaturze poświęconej zagadnieniom radiowych sieci teleinformatycznych dominują aspekty związane z analizą efektywności protokołów komunikacyjnych swobodnego ew. częściowo- kontrolowanego dostępu do kanału. Głównymi parametrami charakteryzującymi efektywność są przy tym sprawność wykorzystania kanału oraz średnie opóźnienie przy przesyłaniu wiadomości.

W większości analiz poświęconych przepustowości kanałów w radiowych sieciach teleinformatycznych pomija się wpływ informacji o charakterze kontrolnym; tj. pakietów potwierdzeń na rzeczywiste wykorzystanie kanału. W sieciach tych protokół komunikacyjny narzuca potrzebę przesyłania pozytywnych potwierdzeń o pakietach odebranych poprawnie. Ten dodatkowy ruch generowany przez poszczególnych użytkowników sieci / ew. węzeł centralny/ odgrywa więc ważną rolę w jej prawidłowym funkcjonowaniu. Jednocześnie

* Politechnika Gdańska, Instytut Telekomunikacji, Gdańsk

ruch ten w sposób istotny obciąża kanał łączności. Przedmiotem pracy jest analiza wpływu strumienia informacji kontrolnych na przepustowość kanału radiowego w sieci jednokokowej. Zaprezentowane wyniki odnoszą się do kilku wersji dwóch podstawowych protokołów typu ALOHA, tzn. protokołu asynchronicznego /pure ALOHA [1],[4]/ oraz synchronicznego /slotted ALOHA [3],[4]/. Przedmiotem analizy są ponadto dwa typy konfiguracji. W pierwszym z nich : tzw. konfiguracji z kanałem wspólnym /KW/ , pakiety potwierdzeń /PACK/ są przesyłane przez kanał wykorzystywany jednocześnie do transmisji pakietów informacyjnych /PI/. W drugim typie przyjmuje się , że pakiety PACK są przesyłane kanałem pomocniczym /KP/ wydzielonym z paśmie KW . W tym przypadku pakiety PI są przesyłane niezależnym kanałem podstawowym /KPI/ .

2. ANALIZA ZMIAN ŚREDNIEGO PRZEPŁYWU PAKIETÓW PI - PAKIETY PACK PRZESYŁANE PRZEZ KW

W rozdziale tym zakładając będziemy, że pakiety PI o długości L_{PI} oraz pakiety PACK o długości L_{ACK} przesyłane są przez KW o paśmie B.

Protokół asynchroniczny bez priorytetu / ABP - KW /

Jeżeli przyjmiemy , że oba typy pakietów mogą być przesyłane , bez żadnego uprzywilejowania przez KW, prowadzić to będzie do wzajemnych ich kolizji. Udział w rzeczywistym przepływie S będą miały jedynie te udanie transmitowane pakiety PI, których pakiety PACK zostały również przesłane bez interferencji. Przeprowadzając analizę średniego przepływu pakietów PI przy założeniu, że z punktu widzenia wybranego użytkownika przesyłane przez niego pakiety PI oraz PACK pojawiają się w KW w przypadkowych chwilach czasu a odstępy pomiędzy kolejnymi pakietami są wykładnicze / patrz np. 8 / otrzymujemy :

$$S_{ABP-KW} = \frac{L_{PI}}{L} \text{Gexp}(-G \frac{L + L_{PI} + L_{ACK}}{L}) \quad (1)$$

gdzie L - średnia długość pakietów PI oraz PACK;

G - średnia zajętość KW .

Przyjmując ponadto $L_{PI} = L_{ACK}$ mamy :

$$S_{ABP-KW} = G e^{-3G} \quad (2)$$

Podobne wartości S otrzymujemy również w wyniku analizy zaprezentowanej w [2] / przy założeniu pojedynczych potwierdzeń / .

Protokół asynchroniczny z priorytetem / AZP - KW /

W tym przypadku użytkownicy mają możliwość identyfikacji pakietów

przesyłanych poprawnie. Po wykryciu udanej transmisji pakietu PI wszyscy potencjalni nadawcy wstrzymują swoje transmisje przez czas $t \geq T_{ACK}$, następujący bezpośrednio po tej transmisji, gdzie:

$$T_{ACK} = L_{ACK} \frac{1}{B} \quad (3)$$

Prowadzi to do następującej zależności:

$$S_{AZP-KW} = \frac{G e^{-2G}}{1 + a G e^{-2G}} \quad ; \quad a = \frac{L_{ACK}}{L_{PI}} \quad (4)$$

Podobną zależność można również znaleźć w [2].

Protokół synchroniczny bez priorytetu / SBP-KW /

Podobnie jak w przypadku protokołu asynchronicznego potwierdzenia przesyłane są natychmiast po zakończeniu udanej transmisji pakietu PI. Przyjmując, że rozkład liczby pakietów PI pojawiających się w dowolnej ramce jest opisany wzorem:

$$q_k = \frac{G^k}{k!} e^{-G} \quad ; \quad k = 0, 1, 2, \dots \quad (5)$$

Średni przepływ otrzymujemy w postaci [7]:

$$S_{SBP-KW} = \frac{G e^{-2G}}{1 + G e^{-G}} \quad (6)$$

Protokół synchroniczny z priorytetem / SZP-KW /

W przypadku tego protokołu wszyscy użytkownicy /z wyjątkiem jednego przesyłającego pakiet PACK/ wstrzymują swoje transmisje po stwierdzeniu udanie przesłanego pakietu PI. Przyjmując, że $L_{PI} = L_{ACK}$ mamy wówczas [7]:

$$S_{SZP-KW} = \frac{G^{-G}}{1 + 2G e^{-G} (1 - e^{-G})} \quad (7)$$

3. ANALIZA PRZEPUSTOWOŚCI - PAKIETY PACK PRZESYŁANE PRZEZ KP

Założymy obecnie, że pakiety PACK przesyłane są kanałem pomocniczym KP o paśmie B_a wydzielonym z pasma B kanału KW. Pasma kanału KPI jest natomiast równe B_i .

Protokół asynchroniczny bez buforowania pakietów PACK / ABB-KP /

W tym przypadku pakiety PACK są przesyłane przez KP natychmiast po udanej transmisji pakietu PI. Aby uniknąć interferencji pakietów PACK w KP musimy zapewnić warunek:

$$L_{ACK} \frac{1}{B_a} = L_{PI} \frac{1}{B_i} \quad (8)$$

Tym samym przepustowość systemu $C_{ABB-KP} = \max_G S_{ABB-KP}$ przyjmuje postać :

$$C_{ABB-KP} = \frac{1}{1 + a} \frac{1}{2e} \quad \left| \quad G = 1/2 \right. \quad (9)$$

Protokół asynchroniczny z buforowaniem pakietów PACK / AZB-KP /

Protokół ten może znaleźć zastosowanie w sieci z wydzielonym węzłem centralnym, w którym pakiety PACK przed ich wysłaniem mogą być buforowane. Pozwala to na efektywniejsze wykorzystanie pasma B_a kanału KP. Przepustowość systemu otrzymujemy w tym przypadku jako:

$$C_{AZB-KP} = \frac{1}{a + 2e} \quad \left| \quad G = 1/2 \right. \quad (10)$$

W przypadku protokołu synchronicznego odpowiednie zależności przyjmują postać:

Protokół synchroniczny bez buforowania / SBB-KP /

$$C_{SBB-KP} = \frac{1}{1 + a} \frac{1}{e} \quad \left| \quad G = 1 \right. \quad (11)$$

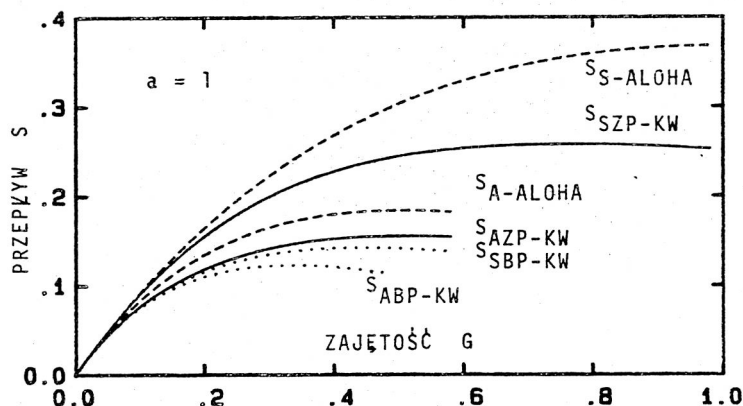
Do identycznej zależności prowadzi analiza systemu, w którym każda ramka czasowa podzielona jest na dwie części; informacyjną i kontrolną, służące do przesyłania PI oraz PACK, przy czym $L_{ACK}/L_{PI} = a$.

Protokół synchroniczny z buforowaniem / SZB-KP /

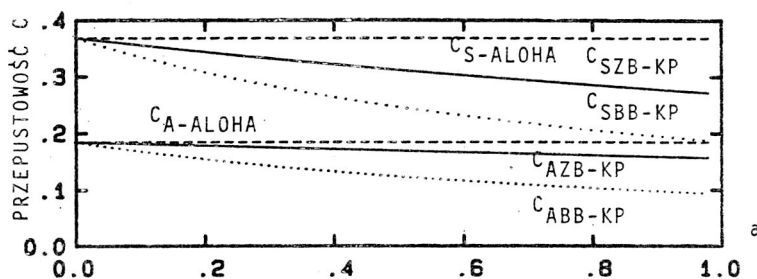
$$C_{SZB-KP} = \frac{1}{a + e} \quad \left| \quad G = 1 \right. \quad (12)$$

4. PORÓWNANIE PROTOKOŁÓW I WNIOSKI KOŃCOWE

Porównania protokołów omówionych w rozdz.2 i 3 dokonano na rys.1 i 2. Ilustrują one spadek wykorzystania kanału /przepustowości/ spowodowany przesyłaniem pozytywnych potwierdzeń o pakietach PI odebranych poprawnie.



Rys.1. Rzeczywisty przepływ S w funkcji zajętości kanału G.
Fig.1. Effective channel throughput S v. total traffic G.



Rys.2. Rzeczywista przepustowość C w funkcji $a = L_{ACK}/L_{PI}$.
Fig.2. Effective channel capacity C v. a .

Z zaprezentowanych wyników widać, że w przypadku konfiguracji z kanałem wspólnym /rys.1/ zdecydowaną przewagę posiada protokół z priorytetem. Protokół ten może być jednak stosowany jedynie wówczas, gdy wszyscy użytkownicy badają stan kanału KW. W przypadku protokołu bez priorytetu pakiety PACK powodują bardzo wyraźny spadek przepustowości KW. Przyjmując $a = 1$ maksymalne wykorzystanie kanału w systemie asynchronicznym CA-ALOHA spada z $1/2e$ do ok. 0.12. Jeszcze drastyczniejszy spadek z $1/e$ do ok. 0.14 obserwujemy w systemie synchronicznym. Z kolei wśród priorytetów odnoszących się do konfiguracji z kanałem wydzielonym najlepsze wyniki pozwala uzyskać protokół z buforowaniem pakietów PACK. Wiąże się to z lepszym wykorzystaniem przepustowości KP/patrz rys.2 /.

LITERATURA

- [1] Abramson N. : The throughput of packet broadcasting channels. IEEE Trans.Commun.1977, pp. 117-128.
- [2] Borgonovo F., Fratta L. : Analysis of random channel throughput degradation due to acknowledgment traffic. URSI Gen. Assembly, 1981.
- [3] Eslanadidi M., Chu W. : Study of acknowledgment schemes in a star multiaccess network. IEEE Trans.Commun.1982, pp. 1657-1667.
- [4] Kozłowski A., Woźniak J. : Radiowe sieci teleinformatyczne ; w: Problemy teleinformatyki. WKiŁ 1984.
- [5] Seidler J. : Analiza i synteza sieci łączności dla systemów teleinformatycznych. PWN 1979.
- [6] Tobagi F. : Multiaccess protocols in packet communication systems. IEEE Trans.Commun.1978, pp. 468-488.
- [7] Tobagi F., Kleinrock L. : The effect of acknowledgment traffic on the capacity of radio channels. IEEE Trans.commun.1978, pp. 815-826.
- [8] Woźniak J. : Variations of the channel throughput for geometrically distributed packets lengths. Arch.Aut.Telemech.1984, pp.377-388.

INFLUENCE OF ACKNOWLEDGMENT TRAFFIC ON THE CHANNEL THROUGHPUT IN PACKET-SWITCHING RADIO NETWORKS

Most analysis devoted to packet-switching radio networks neglects the influence of acknowledgment traffic on network performance parameters. In the paper we investigate the channel throughput variations in star and distributed single-hop radio networks taking into account ACK packets. We study the channel throughput degradation due to ACK traffic assuming that pure and slotted ALOHA protocols are used.

ВЛИЯНИЕ КОНТРОЛЬНЫХ ИНФОРМАЦИЙ НА ДЕЙСТВИТЕЛЬНОЕ ИСПОЛЬЗОВАНИЕ КАНАЛА В ТЕЛЕИНФОРМАЦИОННОЙ РАДИОСЕТИ

В большинстве работ посвященных анализу пропускной способности каналов в радиосетях опускается влияние контрольных информации, т. е. пакетов подтверждений на действительное использование канала. В работе представлены зависимости между средним действительным проходом пакетов (пропускаемостью канала) и основными параметрами характеризующими единичную сеть полной или звездообразной структурой. Проанализировано два основных протокола свободного доступа - так называемые асинхронический и синхронический протокол типа АЛОНА - с учётом нескольких их версии.

SPIS RZECZY

E. BIELENINIK, J. LEWOC, A. STANISZ, Podsić transmisji danych sieci MSK KASK	5
E. BILSKI, Usługi i standardy sieci transmisji danych	20
L. BÓRZEŃSKI, A. GRZECH, A. KASPRZAK, Projektowanie lokalnych sieci komputerowych	34
M. BUDKA, J. WOJTUSZEK, Współpraca lokalnych i rozległych sieci komputerowych - zasady i sposoby implementacji	53
Z. HUZAR, J. KWIATKOWSKI, Rozproszone przetwarzanie zadań w sieci komputerowej	65
Z. HUZAR, A. KALIŚ, P. KOCIATKIEWICZ, Projekty i realizacje sieci komputerowych	82
A. KALIŚ, R. KASZUBA, Transfer, dostęp i zarządzanie zbiorami w modelu systemu otwartego	100
W. MOLISZ, Teleinformatyczne systemy przekazywania wiadomości (MHS)	115
M. STROIŃSKI, Metody formalnej specyfikacji protokołów oraz ich zastosowania	133
J. WIETRZYCH, Usługi i protokół wirtualnego terminala w modelu połączeń systemów otwartych ISO OSI	150
W. ABRAMOWICZ, O nowym sposobie udostępniania informacji w sieciach komputerowych	169
J. BARCZYŃSKI, Opis i weryfikacja protokołów - podejście aksjomatyczne	174
L. BUDZIANOWSKI, A. FRYŚ, Własności algorytmu wyboru drogi w sieci SKJS 2	182
P. COFTA, S. KACZMAREK, Zastosowanie sieci lokalnych w dedykowanych systemach wielomikrokomputerowych	188
W.B. DASZCZUK, Pierścieniowa sieć mikrokomputerów	192
A. DUBIELEWICZ, J. RATAJCZAK, Ocena wykorzystania czasu pracy stacji transferu zbiorów w warunkach występowania uszkodzeń	193
I. DUBIELEWICZ, Z. FRYŻLEWICZ, K. KOŁEŚNIK, W. KOMOROWSKI, L. MISIURA, Aparatura pomiarowo-diagnostyczna dla sieci komputerowej	205
I. DUBIELEWICZ, Z. FRYŻLEWICZ, K. KOŁEŚNIK, W. KOMOROWSKI, L. MISIURA, Testowanie programowe w sieci transmisji danych	211
M. GUT-MOSTOWY, Zagadnienia synchronizacji sieci SLLKOM	215
E. HUDYMA, E. KOSMULSKA-BOCHENEK, Badanie przepływu pomiarowo-sterującego w systemie transmisji danych sieci komputerowej ...	220

A. KALIŚ, R. KASZUBA, J. KWIATKOWSKI, Implementacja transferu, dostępu i zarządzania zbiorami w modelu systemu otwartego	226
A. KALIŚ, R. KASZUBA, J. KWIATKOWSKI, Usługa transferu zbiorów w sieci komputerowej	232
G. KATANA, L. KOTULSKI, A. SOBALA, Miejsce alokacji w systemie generacji oprogramowania współbieżnego dla lokalnych sieci komputerowych	238
I. KLIMEK-GRZESIAK, Wymagania stawiane lokalnym sieciom komputerowym przez informatyczny system szpitalny	242
L. KOTULSKI, Możliwości zastosowania języków współbieżnych do implementacji oprogramowania w lokalnych sieciach komputerowych	249
L. KOTULSKI, O możliwościach zastosowania języka COLNET do implementacji rozproszonych baz danych	254
J. LEWOC, L. MISIURA, Analiza przepustowości w sieciach lokalnych z transmisją zbiorów	258
J. LEWOC, L. MISIURA, Oszacowanie opóźnienia po oczku dla sieci lokalnej Ethernet	264
T. LUBIŃSKA, System rozproszony - próba ponownego przyjrzenia się zasobom informacyjnym przedsiębiorstwa	269
S. PŁACZEK, Lokalne monitory ekranowe w sieci MSK	273
T. WALASEK, Z. ZIELIŃSKI, Koncepcja oceny poprawności funkcjonalnej i efektywności podsystemu transmisji sieci UMELAN-2	278
J. WOŹNIAK, Wpływ informacji kontrolnych na rzeczywiste wykorzystanie kanału w radiowej sieci teleinformatycznej	283

CONTENTS

E. BIELENINIK, J. LEWOC, A. STANISZ, Communication Subnetwork of the MSK KASK networks	5
E. BILSKI, Services and standards of data transmission networks	20
L. BORZEMSKI, A. GRZECH, A. KASPRZAK, Local area network design-access-control discipline adjustment problem	34
M. BUDKA, J. WOJTUSZEK, Interworking LAN and WAN - principles and implementations	53
Z. HUZAR, J. KWIATKOWSKI, Distributed job processing in computer network	65
Z. HUZAR, A. KALIŚ, P. KOCIATKIEWICZ, Computer network projects and implementations	82
A. KALIŚ, R. KASZUBA, File transfer, access and management for open systems interconnection	100



W. MOLISZ, Message handling systems	115
M. STROIŃSKI, Protocol specification methods and their application	133
J. WIETRZYCH, The virtual terminal service and protocol in the ISO OSI open systems interconnection model	150
W. ABRAMOWICZ, CODI-a new concept to the dissemination of information in computer networks	169
J. BARCZYŃSKI, Description and verification of protocols-an axiomatic approach	174
L. BUDZIANOWSKI, A. FRYS, Properties of routing algorithm in the computer network SKJS 2	182
P. COFTA, S. KACZMAREK, The use of the local area network for the dedicated multimicrocomputer systems	188
W. B. DASZCZUK, The ring network of microcomputers	192
A. DUBIELEWICZ, J. RATAJCZAK, The file transfer station time utilization under faults occurrence	199
I. DUBIELEWICZ, Z. FRYŻLEWICZ, K. KOLEŚNIK, W. KOMOROWSKI, L. MISIURA, SPS-diagnostics and measuring system in computer networks	205
I. DUBIELEWICZ, Z. FRYŻLEWICZ, K. KOLEŚNIK, W. KOMOROWSKI, L. MISIURA, Programmed testing in computer networks	211
H. GUT-MOSTOWY, The synchronization problems of SYNKOM network	215
E. HUDYMA, E. KOSMULSKA-BOCHENEK, Estimation of the control and diagnostic flow in computer network	220
A. KALIŚ, R. KASZUBA, J. KWIATKOWSKI, File transfer, access and management implementation for open systems interconnection	226
A. KALIŚ, R. KASZUBA, J. KWIATKOWSKI, File transfer service in computer network	232
G. KATANA, L. KOTULSKI, A. SOBALA, Some notes concerning distributed systems software construction	236
I. KLIMEK-GRZESIAK, The hospital information system requirements for local networks	242
L. KOTULSKI, Possibilities of using concurrent languages to implementation of software environment of LANS	249
L. KOTULSKI, Possibilities of using COLNET language to implementation Distributed Data Bases	254
J. LEWOC, L. MISIURA, Transport analysis in local area networks with file transfers	258
J. LEWOC, L. MISIURA, Round-trip delay evaluation for the local area network Ethernet	264

T. LUBINSKA, The distributed system-an attempt to Re-examine the information resources of the enterprise	269
S. PLACZEK, Local monitor displays in MSK network	273
T. WALASEK, K. ZIELINSKI, The concept of the UMILAN-2 validation and performance measurement	278
J. WOZNIAK, Influence of acknowledgement traffic on the channel throughput in packet-switching ratio network	283



PRACE NAUKOWE CENTRUM OBLICZENIOWEGO

(wydane w latach 1980—1987)

Nr 1, Studia i Materiały nr 1, <i>Zagadnienia oprogramowania systemów komputerowego wspomagania dydaktyki</i> , Wrocław 1980	20,—
Nr 2, Studia i Materiały nr 2, <i>Zagadnienia oprogramowania węzła pod-sieci komunikacyjnej sieci komputerowej MSK</i> , Wrocław 1983	180,—
Nr 3, Konferencje nr 1, <i>Sieci komputerowe. Usługi, protokoły, modele — część 1</i> , Wrocław 1985	160,—
Nr 4, Konferencje nr 2, <i>Sieci komputerowe. Usługi, protokoły, modele — część 2</i> , Wrocław 1986	250,—

Cena 335,—

BIBLIOTEKA
Politechniki Wrocławskiej

A 4947 I/3
1987 nr 3/5

Subscription should be sent (at any time of the year) to:

„Ars Polona”

Krakowskie Przedmieście 7, 00-068 Warszawa
or OR PAN, 00-901 Warszawa, PKiN, POLAND

Bank account number: NBP, VIII OM Nr 1550-6-81574 Warszawa/Poland

Wydawnictwa Politechniki Wrocławskiej
ma stałe na składzie Księgarnia Wr 49
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław
oraz Wojewódzka Księgarnia Techniczna
ul. Świdnicka 8, 50-067 Wrocław

ISSN 0860-1615