

Novell®

1. Konceptja systemu Novell NetWare

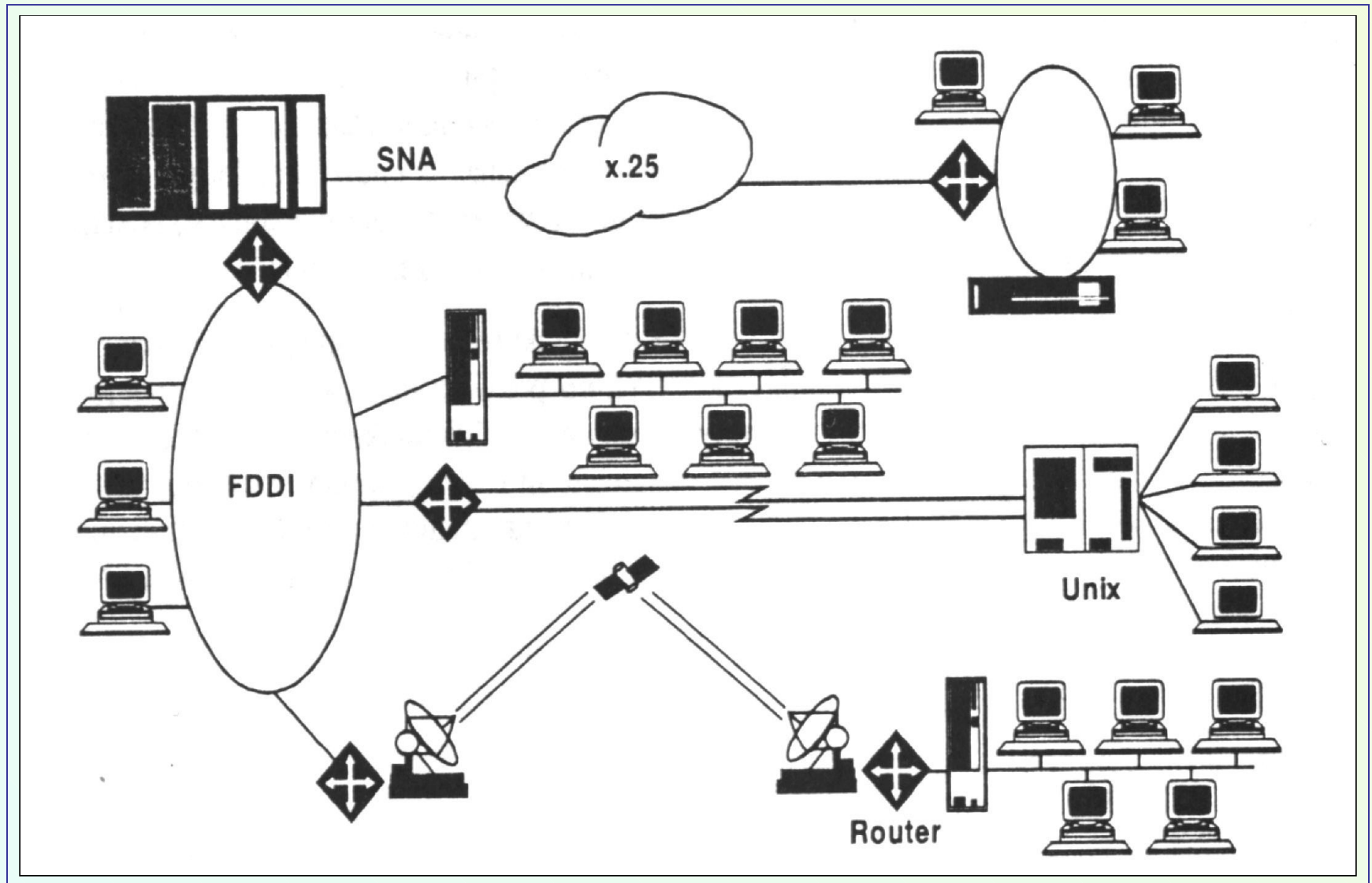
1. Koncepcja systemu

- System NetWare jest systemem operacyjnym dla sieci globalnych, rozumianych jako zespół, na ogół oddalonych od siebie, połączonych wzajemnie sieci lokalnych.
- Z punktu widzenia użytkownika najważniejszy jest łatwy dostęp do usług, którymi dysponuje sieć.
- System organizuje dostęp do nich w sposób, że nie wymaga od użytkownika znajomości struktury sieci. Kluczem do takiej organizacji jest baza informacyjna o wszystkich obiektach sieci, takich jak: użytkownicy, grupy użytkowników, drukarki, woluminy, komputery itp.

1. Koncepcja systemu

- System ten wychodzi znacznie poza obszar funkcjonowania serwera plikowego i drukowania, jest zorientowany na dostarczenie uniwersalnych usług w sieciach globalnych o rozproszonych zasobach.
- System zawiera elementy zorientowane na łączność w sieciach rozległych – dotyczy to funkcji komunikacyjnych, mechanizmów zarządzania i narzędzi diagnostycznych, ale także zoptymalizowania pewnych operacji, mających wpływ na szybkość i bezpieczeństwo przesyłania danych na duże odległości.

1. Koncepcja systemu



1.1. Usługi bazy obiektowej

- Najważniejszą cechą NetWare jest wyposażenie go we wszechstronne mechanizmy obsługi baz danych o wszelkich zasobach i obiektach sieci.
- System obsługi tej bazy nazywa się *NetWare Directory Services* (NDS), a jego zasadniczą funkcją jest zorganizowanie dostępu do informacji o wszelkich obiektach w sieci globalnej i kontrolowanie wzajemnej zależności tych obiektów.

1.1. Usługi bazy obiektowej

- Zgodność bazy obiektowej ze standardem X.500 narzuca pewne wymagania na jej strukturę i funkcjonalność:
 - wszelkie zasoby sieci definiowane są jako obiekty, które zarządzane są przez NDS w sposób hierarchiczny,
 - drzewiasta struktura bazy tych obiektów odzwierciedla schemat organizacyjny przedsiębiorstwa;

1.1. Usługi bazy obiektowej

- każdy obiekt (np. użytkownik, jednostka organizacyjna, komputer, drukarka) posiada zespół cech, którym przypisane są odpowiednie wartości (np. numery seryjne dla komputera czy drukarki, a adres dla użytkownika);
- użytkownik sieci może odwoływać się do obiektów za pomocą prostych nazw, kojarzonych z typem i cechami obiektów, natomiast NDS przeprowadza ich jednoznaczny identyfikację;

1.1. Usługi bazy obiektowej

- nazwy obiektów są dynamicznie przypisywane ich adresom lub fizycznym lokalizacjom;
- użytkownik może korzystać z funkcji skorowidzu typu *yellow pages*, czyli wyszukiwania obiektów w bazie na podstawie określonych przez siebie kryteriów
- baza **NDS** może być podzielona na logiczne fragmenty, zwane **partycjami**, niekoniecznie mającymi związek z fizyczną lokalizacją obiektów.

1.1. Usługi bazy obiektowej

- partycje te mogą być kopiowane (replikowane) na wielu serwerach, co zabezpiecza bazę przed utratą integralności w przypadku zniszczenia danych pierwotnej kopii partycji.
- baza **NDS** jest automatycznie synchronizowana, a globalny system synchronizacji czasowej kontroluje spójność danych w przypadku modyfikacji bazy z wielu odległych miejsc równocześnie.

1.2. Podstawowe funkcje NetWare

Kompresja danych

- System NetWare posiada wbudowane funkcje kompresji i dekompresji danych na dyskach sieciowych.
- Po odpowiednim ustaleniu parametrów tych funkcji, operacje na plikach wykonywane są automatycznie, bez udziału operatora.

1.2. Podstawowe funkcje NetWare

Kompresja danych

- Przyjęto następujące założenia:
 - kompresja danych jest zadaniem o niskim priorytecie i wykonywana jest tylko wtedy, gdy nie ma do obsłużenia innych zadań, związanych z zapisem lub odczytem dysku;
 - dekompresja danych ma wysoki priorytet i odbywa się w chwili odczytu danych przechowywanych na dysku w postaci skompresowanej.

1.2. Podstawowe funkcje NetWare

Migracja danych

- Mechanizm migracji danych polega na automatycznym przerzucaniu danych z dysków sieciowych na zewnętrzne nośniki pamięci (urządzenia o dużej pojemności takie jak pamięci taśmowe i optyczne).
- Administrator ustala jedynie parametry migracji, np. procentowy obszar dysku sieciowego, który musi pozostać wolny do bieżących zadań.

1.2. Podstawowe funkcje NetWare

Migracja danych

- W chwili gdy zostanie on naruszony przez nowo zapisywany plik, NetWare uruchamia migrację i najdłużej nie używane pliki zostają odesłane do pamięci zewnętrznej, zwalniając obszar na dysku sieciowym.
- Gdy użytkownik żąda dostępu do pliku, który podlegał migracji, plik ten zostaje natychmiast sprowadzony na dysk sieciowy.

1.2. Podstawowe funkcje NetWare

Alokacja ułamkowa bloków pamięci

- Zapis danych na dysku odbywa się blokami. Jeżeli zależy na dużej szybkości zapisu i odczytu danych, należy stosować duże bloki. Wiąże się to jednak z marnotrawstwem pamięci w przypadku, gdy rozmiar zapisywanego pliku nieznacznie przekracza wielokrotność bloku.
- W systemie NetWare zastosowano tzw. alokację ułamkową bloków pamięci, która automatycznie dokonuje podziału nie w pełni wykorzystanych bloków na mniejsze jednostki - po 512 bajtów

1.2. Podstawowe funkcje NetWare

Obsługa języków narodowych

- NetWare został zaadaptowany do równoczesnej obsługi wielu języków narodowych.
- Każda stacja robocza może zostać tak skonfigurowana, aby jej komunikacja z systemem NetWare odbywała się w wybranym języku (warunkiem jest istnienie danej wersji językowej).
- Również konsola serwera może być skonfigurowana do obsługi w języku innym niż angielski.

1.2. Podstawowe funkcje NetWare

Obsługa języków narodowych

- Wewnętrzna struktura systemu została tak zaprojektowana aby dostosowanie do wersji językowej nie wymagało ingerencji w kod programów.
- Wszystkie łańcuchy tekstowe (programów usługowych, programów typu NLM oraz systemu) zostały odseparowane od kodu.

1.2. Podstawowe funkcje NetWare

Sprawdzanie tożsamości użytkownika

- W systemie NetWare logowanie odbywa się do całej sieci a nie do pojedynczego serwera (system udostępnia zasoby całej sieci lecz tylko w zakresie posiadanych praw).
- W trakcie rejestrowania się użytkownika powstaje unikalny "podpis" (*signature*) będący wynikiem złożonych operacji na jego zaszyfrowanym haśle i kluczu prywatnym generowanym indywidualnie dla każdej sesji.

1.2. Podstawowe funkcje NetWare

Sprawdzanie tożsamości użytkownika

- Podpis ten jest następnie wykorzystywany do identyfikacji użytkownika (już bez jego udziału) w trakcie sięgania do poszczególnych zasobów.
- Hasło nigdy nie jest przesyłane w sieci, ani też przechowywane w pamięci lokalnej stacji roboczej.

1.2. Podstawowe funkcje NetWare

Kontrola i rejestracja zdarzeń

- W systemie NetWare są rejestrowane następujące zdarzenia:
 - operacje rejestrowania i wyrejestrowywania się z sieci,
 - operacje na obiektowej bazie zasobów sieci, takie jak tworzenie obiektów, ich usuwanie, zapisy i odczyty ich własności,
 - modyfikacje praw dysponenckich do obiektów,
 - operacje na plikach, takie jak ich tworzenie, usuwanie, zapisy i odczyty,

1.2. Podstawowe funkcje NetWare

Kontrola i rejestracja zdarzeń

- operacje na kolejkach do drukowania,
- działalność poszczególnych użytkowników i innych obiektów sieci.

Programy usługowe w środowisku graficznym

- Administratorzy i użytkownicy mają do wyboru programy usługowe z interfejsem znakowym, pracujące pod kontrolą DOS lub programy graficzne, przeznaczone dla środowiska MS-Windows i OS/2.

1.2. Podstawowe funkcje NetWare

Drukowanie

- Zarówno serwer drukowania, kolejki drukowania, jak i same drukarki są obiektami w bazie zasobów sieciowych NDS.
- Oznacza to, że administrator ma dużą swobodę w sterowaniu nimi i kontrolowaniu dostępu do nich, a zwykły użytkownik może je łatwo zlokalizować i poznać ich cechy wykorzystując graficzne lub tekstowe programy usługowe.
- Serwer drukowania zawarty w pakiecie NetWare potrafi obsłużyć do 256 drukarek.

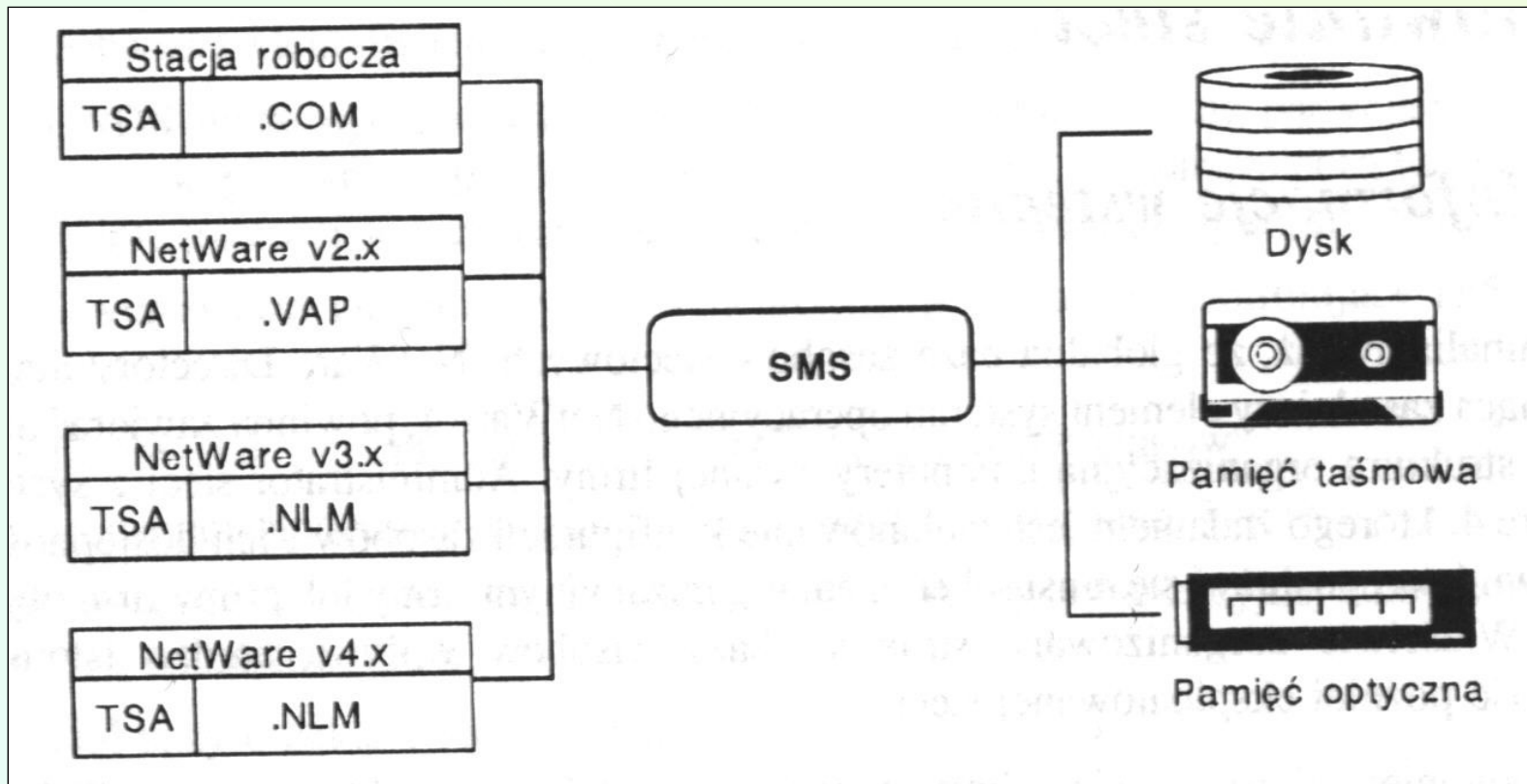
1.2. Podstawowe funkcje NetWare

Uniwersalne usługi archiwizowania

- NetWare zawiera usługi zarządzania pamięciami zewnętrznymi o nazwie **SMS** (*Storage Management Services*), które przeznaczone są do zorganizowanego przesyłania i odzyskiwania danych z zewnętrznych nośników pamięci.
- System ten charakteryzuje:
 - niezależność sprzętowa,
 - niezależność programowa,
 - wspólny format danych dla różnych urządzeń.

1.2. Podstawowe funkcje NetWare

Uniwersalne usługi archiwizowania



Novell®

2. Planowanie sieci

2. Planowanie sieci

Zasadniczym celem planowania jest:

- ułatwienie użytkownikom dostępu do sieci,
- zapewnienie ochrony zasobów przed zniszczeniem i niepowołanym dostępem,
- zminimalizowanie przesyłanej w sieci informacji,
- uproszczenie administrowania i utrzymania sieci.

2. Planowanie sieci

Proces planowania sieci można rozłożyć na kilka etapów:

- opracowanie standardu nazewnictw,
- opracowanie drzewa obiektów,
- podział bazy obiektów na partycje,
- określenie zasad synchronizacji czasowej w sieci,
- ustalenie zasad ochrony zasobów sieci.

2.1. Standard nazewnictw

- Nazewnictwo obiektów powinno być przejrzyste i jednoznaczne.
- Podobnie jak dla plików, pełna nazwa obiektu jest określana na podstawie jego położenia w drzewie obiektów, co oznacza, że w różnych gałęziach drzewa mogą znajdować się obiekty o tych samych nazwach.
- Dwie podstawowe operacje wykonywane na bazie obiektów to przeglądanie (*browsing*) i wyszukiwanie (*searching*).
- Precyzja tych operacji zależy w dużej mierze od właściwego przypisania obiektom nazw i unifikacji wartości przypisanych poszczególnym ich cechom.

2.2. Opracowanie drzewa obiektów

- Nie można tu mówić o jednym poprawnym rozwiązaniu problemu, lecz przy projektowaniu należy przestrzegać pewnych ogólnych zasad:
 - każde przedsiębiorstwo powinno posiadać tylko jedno drzewo obiektów (system tworzy oddzielne bazy danych o obiektach dla każdego drzewa, a informacje z jednej bazy nie są widziane przez obiekty innych baz),
 - struktura drzewa powinna być przejrzysta i odpowiadać logicznej organizacji przedsiębiorstwa,
 - jeżeli planowane jest centralne zarządzanie siecią, należy starać się planować strukturę "płaską" o małej liczbie poziomów.

2.2. Opracowanie drzewa obiektów

- Drzewo tworzone jest z trzech typów obiektów:
 - korzenia (*Root*),
 - kontenerów (*Container*),
 - liści (*Leaf*).



Obiekt typu **korzeń** tworzony jest w chwili instalacji systemu na szczycie drzewa i jest jedynym obiektem tego typu.

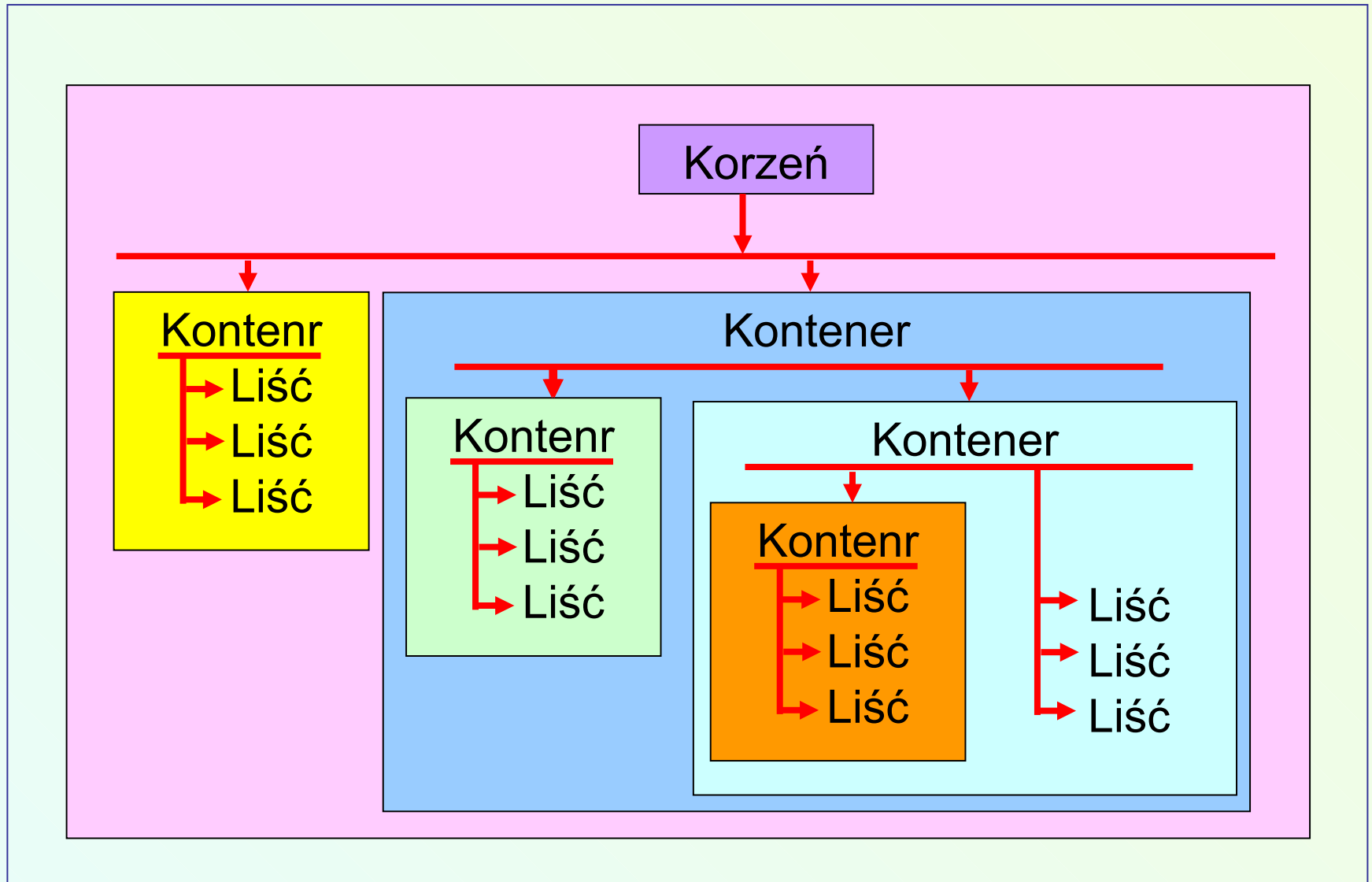


Rozgałęzienia struktury zawierają kontenery i liście, przy czym kontenery mogą zawierać inne obiekty tego typu.



Obiekty typu liść tworzą zakończenia gałęzi drzewa i nie mogą już zawierać innych obiektów.

2.2. Opracowanie drzewa obiektów



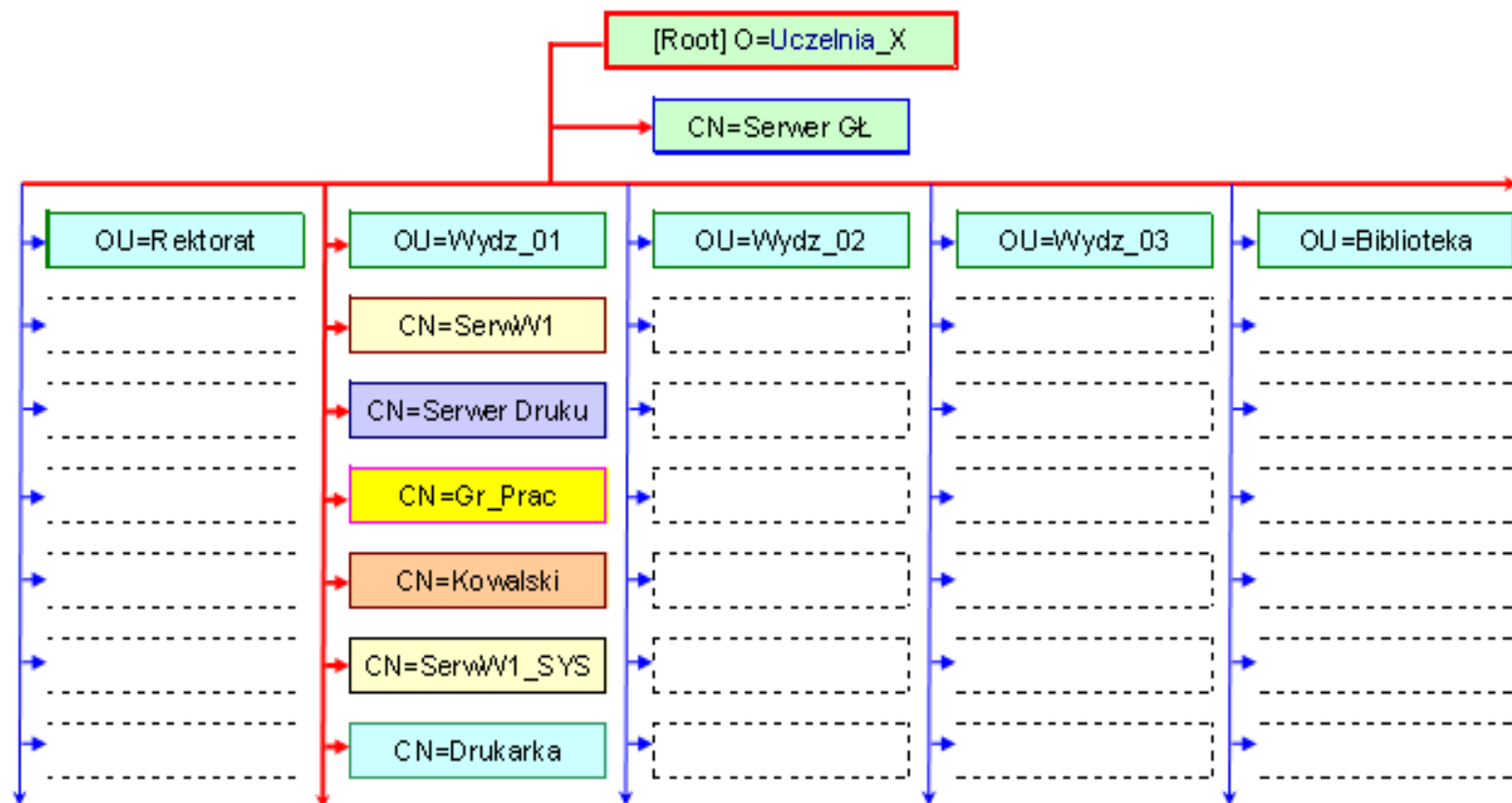
2.2. Opracowanie drzewa obiektów

- Istnieją trzy rodzaje obiektów typu kontener:
 - Kraj (*Country*) - oznaczany literą **C**,
 - Organizacja (*Organization*) - oznaczany literą **O**,
 - Jednostka Organizacyjna (*Organizational Unit*) - oznaczana symbolem **OU**.
- W każdej strukturze drzewa powinien znaleźć się przynajmniej jeden obiekt typu "**Organizacja**",
- Występowanie obiektów typu "**Jednostka Organizacyjna**" nie jest obowiązkowe.

2.2. Opracowanie drzewa obiektów

- Obiekt typu "Kraj" winien być stosowany tylko w wyjątkowych przypadkach.
- Liście to fizyczne lub logiczne obiekty sieci nie zawierające innych obiektów, np. użytkownicy, grupy użytkowników, serwery, woluminy, drukarki, kolejki drukowania, komputery itd. Posiadają one zazwyczaj "Ogólne Nazwy" (*Common Name*) oznaczane symbolem CN.

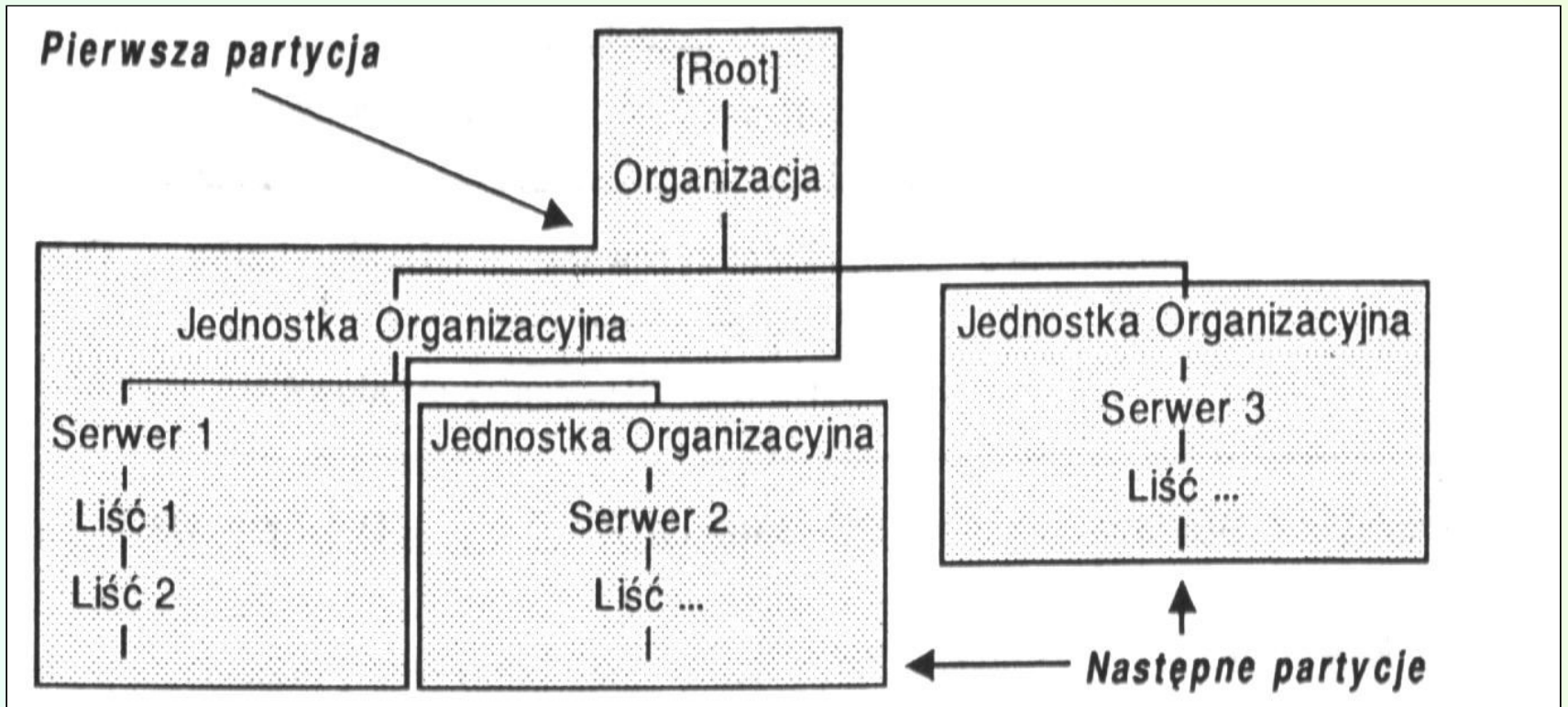
2.2. Opracowanie drzewa obiektów



2.3. Podział bazy obiektów na partycje

- Partycją nazywamy fragment bazy obiektowej NDS powstały z podzielenia jej według kryteriów logicznych.
- Pierwsza partycja, związana z obiektem typu "*Korzeń*", tworzona jest automatycznie w czasie instalacji systemu.
- Następne tworzone są przy instalacji kolejnych serwerów dla nowych *Jednostek Organizacyjnych*..
- Baza NDS jest więc bazą rozproszoną przechowywaną na wielu serwerach.

2.3. Podział bazy obiektów na partycje



2.3. Podział bazy obiektów na partycje

- W przypadku, gdyby któryś z serwerów uległ uszkodzeniu lub nie został włączony do sieci, informacje w bazie byłyby niekompletne.
- Aby temu zapobiec, informacje te są kopiowane i przechowywane na wielu serwerach.
- Zamiast jednak kopiować całą bazę, kopiuje się ją fragmentami, czyli tworzy się kopie (*repliki*) partycji.
- Repliki te są z kolei powielane przynajmniej na dwu serwerach.

2.3. Podział bazy obiektów na partycje

- Wyróżnia się trzy typy replik:
 - **Replika Master** - tworzona automatycznie w trakcie instalacji serwera dla każdej nowo tworzonej *Organizacji lub Jedn. Organizac..*
 - **Replika Read/Write** - w której dane o obiektach mogą być zarówno czytane jak i zmieniane. Standardowo replika ta jest tworzona na serwerze zawierającym strukturę wyższego poziomu.
 - **Replika Read Only** - służy tylko do odczytu i stanowi zabezpieczenie przed skutkami awarii serwera zawierającego replikę *Master*. Z reguły replikę umieszcza się na odległym serwerze.

2.4. Metody synchronizacji czasowej w sieci

- Ze względu na to, że serwery NetWare przechowują dane o obiektach w bazie rozproszonej, która może być modyfikowana z wielu miejsc, zachodzi konieczność synchronizacji wszystkich operacji związanych z bazą obiektową, a w konsekwencji - **zsynchronizowania pracy serwerów**.
- Novell wprowadził pojęcia *serwerów czasowych*, które odpowiedzialne są za synchronizację zdarzeń występujących w sieci.

2.4. Metody synchronizacji czasowej w sieci

- Dostępne są cztery typy serwerów czasowych:
 - **Pojedynczy Wzorcowy Serwer Czasu** (*Single Reference Time Server - SRTS*) - stanowi jedyne źródło czasu w niewielkich sieciach, korzystają z niego wszystkie Serwery Pomocnicze i klienci sieci;
 - **Wzorcowy Serwer Czasu** (*Reference Time Server - RTS*) - stosowany w dużych sieciach, winien być zsynchronizowany z zewnętrznym wzorcem czasu (np. radiowym). Stanowi on wzorzec czasu dla Serwerów Głównych i zawsze musi współpracować przynajmniej z jednym serwerem PTS;

2.4. Metody synchronizacji czasowej w sieci

- Dostępne są cztery typy serwerów czasowych:
 - **Główny Serwer Czasu** (*Primary Time Server - PTS*) - ich zadaniem jest uzgadnianie czasu w sieci na podstawie informacji z innych serwerów i przekazywanie do **Serwerów Pomocniczych**,
 - **Pomocniczy Serwer Czasu** (*Secondary Time Server - STS*) - jedynie uzyskuje informacje od któregoś z pozostałych i przekazuje ją klientom sieci.

Novell.[®]

3. Administrowanie systemem NetWare

3.1. Obiekty sieci i ich własności

Baza NDS została tak pomyślana przez twórców systemu, aby w sposób możliwie precyzyjny i zrozumiały opisać logikę działania sieciowego systemu operacyjnego, zaimplementowanego w rzeczywistych warunkach.

3.1. Obiekty sieci i ich własności

- Po pierwsze, nawet w przy komputeryzacji bardzo dużej instytucji, baza NDS jest w stanie odzwierciedlić jej schemat organizacyjny (strukturę firmy, miejsca pracy osób, a także rozmieszczenie ważnych obiektów takich jak serwery czy też drukarki.
- Po drugie, cały system podlega ścisłym regułom dostępu jednego obiektu do drugiego, co zapewnia ochronę nie tylko przechowywanych w systemie danych ale i zabezpieczenie obiektów NDS przed niepożądaną ingerencją ze strony innych obiektów.

3.1. Obiekty sieci i ich własności

- Zgodnie z zasadami konstrukcji bazy NDS obiektem nadrzędnym jest obiekt określany jako **Root**, jest to jak gdyby główny kontener w którym zawarte są wszelkie inne kontenery.

Obiekt	Kwantyfikator	Może być zawarty w:	Może zawierać:
Country	C	[Root]	Organization lub Alias
Organization	O	[Root] lub Country	Organizational Unit lub Common Name
Organizational Unit	OU	Organization lub Organizational Unit	Organizational Unit lub Common Name
Common Name	CN	Organization lub Organizational Unit	

3.1. Obiekty sieci i ich własności

- Obiekty typu liść są tak pomyślane, by obrazowały rozmieszczenie różnych fizycznych i logicznych elementów sieci w całej strukturze NDS.

Obiekt typu liść	Opis obiektu
AFP Serwer (Serwer typu AFP)	Serwer stosowany w sieciach typu Apple Macintosh, używa protokołu AFP (Apple Talk Filing Protocol).
Alias (Nazwa zastępcza)	Nazwa zastępcza (odsyłacz) do innego istniejącego obiektu NDS. Alias upraszcza odwoływanie się do obiektów w odległych kontenerach, gdyż zamiast podawać nazwę takiego obiektu wraz ze ścieżką dostępu, wykorzystuje się tylko jego nazwę zastępczą.

3.1. Obiekty sieci i ich własności

Obiekt typu liść	Opis obiektu
Computer (Komputer)	Obiekt reprezentujący fizyczny komputer. Stwierdza fakt jego istnienia w danym kontenerze i identyfikuje go podając jego cechy (np. kto na nim pracuje).
Directory Map (Przypisanie napędu)	Obiekt ułatwiający przypisanie liter alfabetu do katalogów woluminów serwerów zawartych w odległych kontenerach.
Group (Grupa użytkowników)	Obiekt reprezentujący pewną grupę użytkowników, pozwala na jednoczesne odwoływanie się do wszystkich członków takiej grupy w różnorodnych operacjach systemowych.

3.1. Obiekty sieci i ich własności

Obiekt typu liść	Opis obiektu
NetWare Server (Serwer typu NetWare)	Obiekt reprezentujący komputer, z system operacyjnym NetWare. Jest on częścią struktury NDS i jest magazynem plików, a ponadto są udostępniane informacje na jego temat np. rodzaj procesora, ilość pamięci operacyjnej, rodzaj i pojemność dysków, lista woluminów itp.
Organizational Role (Osoba pełniąca funkcję administracyjną)	Obiekt mający na celu ułatwienie administrowania systemem. Takiemu obiektowi nadaje się prawa do zarządzania innymi obiektami NDS i wpisuje do jego własności „ <i>occupant</i> ” nazwę konkretnego użytkownika. Użytkownik nabywa tym samym prawa posiadane przez obiekt Organizational Role .

3.1. Obiekty sieci i ich własności

Obiekt typu liść	Opis obiektu
Print Server (Serwer druku)	Moduł programowy obsługujący kolejki drukarskie. Obsługa ta polega na kierowaniu zadań umieszczonych w tych kolejkach na drukarki dostępne w strukturze NDS.
Printer (Drukarka)	Reprezentuje fizyczne urządzenie (drukarkę). Informuje użytkowników sieci o istnieniu takiego urządzenia w strukturze sieci, zawierając jego opis, parametry techniczne, numer itp.

3.1. Obiekty sieci i ich własności

Obiekt typu liść	Opis obiektu
Print Queue (Kolejka zadań drukarskich)	Magazyn zadań przygotowanych do druku. Obiekt typu kolejka oprócz samego spełniania zadania informowania o istnieniu kolejki drukarskiej w danym miejscu struktury NDS posiada również opis cech kolejki takie jak lista użytkowników uprawnionych do korzystania, lista osób zarządzających itp.
Profile (Profil)	W czasie logowania użytkownika do sieci mogą być kolejno wykonane trzy skrypty logowania: kontenerowy, profilowy i indywidualny. Obiekt tego typu jest tworzony po to by przy logowaniu użytkownika posiadającego określony profil był wykonywany profilowy skrypt logowania.

3.1. Obiekty sieci i ich własności

Obiekt typu liść	Opis obiektu
User (Użytkownik sieci)	Reprezentuje rzeczywistego użytkownika NDS, uprawnionego do pracy w sieci. Właściwości tego obiektu zawierają informacje identyfikujące (adres, pełna nazwa), restrykcje i uprawnienia.
Volume (Wolumin)	Obiekt identyfikujący magazyn danych (plików). Dane te są uporządkowane hierarchicznie w postaci struktury katalogów (struktura ta nie należy do NDS). Właściwości tego obiektu zawierają między innymi informacje o wielkości woluminu, rodzaju plików, restrykcje.
Bindery Object (Emulator bindery)	Obiekt umożliwiający starszym wersjom NetWare (v.3.11 i wcześniejsze) dostęp do danych zawartych na woluminach serwerów zdefiniowanych w strukturze NDS.

3.1. Obiekty sieci i ich własności

Obiekt typu liść	Opis obiektu
Bindery Queue	Obiekt emulujący kolejki drukarskie ze starszych wersji NetWare
USER_TEMPLATE (Wzorzec użytkownika)	Obiekt, który istnieje po to by nieść w swoich własnościach opis potencjalnego przyszłego użytkownika. Tworząc nowego użytkownika można wtedy posłużyć się wzorcem.
Unknown (obiekt nieznan)	Obiekt, który na skutek jakichś anormalnych wydarzeń w sieci (np. awaria) przestał być rozpoznawalny.

3.1. Obiekty sieci i ich własności

- Każdy rodzaj obiektu posiada charakterystyczną tylko dla niego listę własności, listy te mogą różnić się między sobą rodzajem występujących elementów jak ich liczbą.
- Z kolei na poziomie obiektu, obiekty tego samego rodzaju mogą posiadać różne wartości poszczególnych własności np.:
 - Obiekt drukarka (*Printer*)

Własność	Wartość własności
Name (nazwa drukarki)	EPSON LQ 1070
Description (opis)	Drukarka do zestawień zbiorczych
Location (likalizacja)	Księgowość (pok. 23)

3.1. Obiekty sieci i ich własności

- Obiekt użytkownik (*User*)

Własność	Wartość własności
Name (nazwa użytkownika)	Kowalski
Login script (program zgłoszenia)	Write „Witaj %LOGIN_NAME !” Map p:=FS_SYS:APPL/FPRO Drive p:
Login time restrictions (ograniczenia czasu pracy)	Można zdefiniować z dokł. do ½ godziny czas dostęp do systemu inny na każdy dzień tygodnia
Password restrctions (ograniczenia przy zmianie hasła przez użytkownika)	(ograniczenia na minimalną długość hasła, wymuszenie zmiany hasła co określoną ilość dni, wymaganie hasła unikalnego itp.)

3.2. Pojęcie kontekstu

- Pojęcie *kontekstu* jest nieco podobne do pojęcia ścieżki dostępu w hierarchicznej strukturze katalogów systemu DOS.
- Kontekst precyzuje w którym miejscu struktury NDS dany obiekt się znajduje.
- Odwołanie do obiektu jest poprawne, jeżeli wyspecyfikowano jego pełny kontekst, tj. ścieżkę zawierającą listę wszystkich kontenerów dzielących dany obiekt od [**Root**].

3.2. Pojęcie kontekstu

- Zgodnie z zastosowaną analogią można też zdefiniować pojęcie *kontekstu bieżącego* jako aktualnej pozycji w NDS, w której znajduje się umowny niewidzialny wskaźnik bieżącego kontenera

CX

- sprawdzenie bieżącego kontekstu;

CX /T /R

- wypisanie całej struktury NDS począwszy od szczytu poprzez wszystkie elementy organizacyjne;

CX /A /CONT

- jak wyżej, plus informacja o wszystkich obiektach;

CX <*nowy_kont.*>

- zmiana bieżącego kontekstu na wskazany.

3.3. Konwencje syntaktyczne w NDS

Wszystkie obiekty istniejące w strukturze NDS muszą mieć swoje nazwy (własność *Nane*) nadawane w chwili ich tworzenia.

- Nazwy obiektów nie mogą być dłuższe niż 64 znaki (z wyjątkiem dwuznakowej nazwy kontenera *Country*)
- W jednym kontenerze nie może wystąpić więcej niż jeden obiekt o tej samej nazwie.
- W nazwie mogą wystąpić znaki specjalne.
- Małe i duże litery są traktowane jak te same znaki.

3.3. Konwencje syntaktyczne w NDS

- Znak odstępu jest utożsamiany ze znakiem podkreślenia „_”
- W NDS specyfikacja kontekstu (odwrotnie niż w DOS-ie) zaczyna się od nazwy obiektu i dalej wymieniane są kolejno nazwy kontenerów w kierunku szczytu struktury (obiekту [Root] jako ostatniego nie wymienia się).
- Dodatkowo jeszcze przyjęto założenie, że każdy obiekt musi (z pewnymi wyjątkami) być poprzedzony właściwym dla niego kwantyfikatorem typu (C, O, OU lub CN).

3.3. Konwencje syntaktyczne w NDS

Przykład:

.CN=JAN.OU=TECH.O=ATC - nazwa pełna

.CN=JAN.OU=SERWIS.O=ATC - nazwa pełna

gdy bieżącym kontekstem jest:

.OU=SERWIS.O=ATC

CN=JAN

- wskazanie Jana z kontenera SERWIS

OU=TECH

CN=TSERWER

CN=TSERWER_SYS

CN=JAN

OU=SERWIS

CN=SSERWER

CN=TSERWER_SYS

CN=JAN

3.4. Pliki systemowe

- Każdy plik posiada swoją nazwę odróżniającą go od innych plików oraz dodatkowe parametry takie jak: data powstania, data modyfikacji, atrybuty i inne.
- W systemie NetWare magazynem plików są woluminy.
- W woluminie pliki te mogą być zorganizowane w hierarchiczną strukturę katalogów.
- Każdy serwer na zamontowanych w nim dyskach posiada przynajmniej jeden wolumin (a maksymalnie 64).

3.4. Pliki systemowe

- Nazwa woluminu może zawierać od dwu do piętnastu znaków alfabetu, znaków cyfr i znaków specjalnych: ~!\$%^&()-_{}.
- Nazwa woluminu musi być unikalną w ramach serwera i zawsze kończy się znakiem dwukropka.
- Jeden z woluminów każdego serwera NetWare musi nazywać się **SYS:**.

3.4. Pliki systemowe



- Pełne odwołanie do woluminu fizycznego powinno być poprzedzone nazwą serwera na którym został on zainstalowany.
- Separatorem dzielącym nazwę woluminu od nazwy serwera jest znak „/” lub „\”.

Np. **XSERWER/VOL1:**

- Z kolei odwołując się do woluminu jako logicznego obiektu w strukturze NDS rolę separatora spełnia znak podkreślenia „_”.

Np. **XSERWER_SYS:**

3.4. Pliki systemowe

- W woluminie SYS:, w czasie instalacji NetWare, zakładane są specjalne katalogi ze zbiorami obsługi

Katalog	Rola pełniona w systemie
LOGIN	Zawiera programy umożliwiające rejestrację użytkownika do systemu a także wykonanie niektórych operacji dopuszczalnych bez w rejestracji się do systemu.
PUBLIC	Przechowuje programy i informacje, które powinny być dostępne dla każdego użytkownika zarejestrowanego do sieci.

3.4. Pliki systemowe

Katalog	Rola pełniona w systemie
MAIL	Może zawierać podkatalogi – po jednym dla każdego użytkownika, które system wykorzystuje na wiele różnych sposobów, m.in. jako skrzynki poczty elektronicznej.
SYSTEM	Przechowuje informacje o systemie operacyjnym, programy modułów ładowalnych oraz niektóre programy dostępne tylko dla operatorów systemu.
ETC	Zawiera pliki przykładowe pomagające skonfigurować system, zwłaszcza gdy dany serwer ma służyć jako węzeł w sieci TCP/IP.

3.4. Pliki systemowe

Katalog	Rola pełniona w systemie
DELETED. SAV	Zawiera skasowane pliki, które nie zostały jeszcze definitywnie usunięte z systemu.
QUEUES	Zawiera podkatalogi reprezentujące obiekty kolejek drukarskich, w których oczekują zadania do wydruku.
DOC	Katalog elektronicznej dokumentacji systemu.
NLS	Katalogi z komunikatami systemowymi i pliki zawierające treść pomocnika użytkownika.

3.5. System zabezpieczeń

- Jest to zestaw mechanizmów pozwalających na skuteczną ochronę zasobów sieci.
- Ochrona obejmuje zarówno system plików (dane) oraz obiekty hierarchicznej struktury NDS.
- System zabezpieczeń posiada kilka niezależnych od siebie poziomów:
- Poziomy zabezpieczeń są na ogół stosowane równocześnie i niezależnie od siebie.

3.5. System zabezpieczeń

Poziomy zabezpieczeń:

- *Login restrictions* - zabezpieczenia na poziomie rejestracji użytkownika do sieci.
- *NDS security* - zabezpieczenia na poziomie struktury hierarchicznej NDS.
- *File system security* - zabezpieczenia ochrony danych i katalogów w woluminach.

3.5.1. Zabezpieczenia na poziomie użytkownika

- System dopuszcza do korzystania ze swoich zasobów tylko wtedy, gdy dany użytkownik przedstawi się, podając swoją nazwę.
- Jeżeli w chwili logowania bieżący kontekst jest inny niż wskazujący na kontener w którym umieszczono użytkownika nazwa powinna zawierać także pełny (lub częściowy) kontekst.

Przykład.

- logowanie z podaniem pełnego kontekstu

LOGIN .CN=JAN.OU=SERWIS.O=ATC

- logowanie przy bieżącym kontekście:

.OU=SERWIS.O=ATC

LOGIN JAN

3.5.1. Zabezpieczenia na poziomie użytkownika

Kolejnym krokiem w procesie logowania jest podanie hasła, po którym system przechodzi do sprawdzenia następujących warunków:

1. czy konto użytkownika nie zostało zablokowane (tj. czy użytkownik nie dostał trwale lub czasowo zakazu pracy w systemie),
2. czy data ważności konta użytkownika nie została przekroczona,
3. czy wprowadzone hasło nie jest zbyt krótkie,

3.5.1. Zabezpieczenia na poziomie użytkownika

4. czy wprowadzone hasło nie powinno być właśnie w tym dniu zmienione (system może żądać cyklicznej zmiany hasła co określoną liczbę dni),
5. czy użytkownik, któryś raz z rzędu, zignorował komunikat systemu o konieczności zmiany hasła (po przekroczeniu zadanego limitu użytkownik nie zostanie dopuszczony do systemu),
6. czy użytkownik zmienił hasło na inne niż każde z siedmiu poprzednich (system może żądać unikalności hasła),

3.5.1. Zabezpieczenia na poziomie użytkownika

7. czy użytkownik zarejestrował się z jednej z fizycznych stacji (komputerów), z których mógł to zrobić (system pozwala zadać listę adresów stacji, z których dany użytkownik może się rejestrować),
8. czy użytkownik zarejestrował się w dniu i godzinie dla niego dozwolonej (system może z dokładnością do pół godziny, w ramach tygodnia kalendarzowego, ustalać dozwolony czas rejestrowania się indywidualnie dla każdego z użytkowników),

3.5.1. Zabezpieczenia na poziomie użytkownika

9. czy użytkownik, rejestrując się do systemu, określoną ilość razy z rzędu (w określonym maksymalnym przedziale czasowym) wprowadził błędne hasło – informacja o takim zdarzeniu jest ogłaszana publicznie (np. w dzienniku systemowym), a konto użytkownika może zostać zablokowane na określony czas.

3.5.2. Zabezpieczenia na poziomie struktury bazy obiektowej

- Zabezpieczenia na poziomie bazy NDS regulują zasady dostępu jednego obiektu do drugiego obiektu i jego własności.
- Formalnie w strukturze NDS funkcjonują dwie kategorie uprawnień:
 - uprawnienia do obiektów (*object rights*),
 - uprawnienia do własności obiektów (*property rights*)

3.5.2. Zabezpieczenia na poziomie struktury bazy obiektowej

oraz pojęcia:

- **Dysponent (*trustee*)** – obiekt, który posiada uprawnienia do katalogu, pliku, innego obiektu lub jego własności,
- **Obiekt dysponowany (*trusteed object*)** – plik, katalog, obiekt lub jego własność, do której jakiś dysponent posiada prawa,
- **Prawo (*rights*)** – lista czynności, które dysponent może wykonać z obiektem dysponowanym lub cechą tego obiektu.

3.5.2. Zabezpieczenia na poziomie struktury bazy obiektowej

Uprawnienie	Znaczenie uprawnienia
Browse	dysponent może oglądać obiekt w drzewie NDS. Bez tego prawa użytkownik nie zobaczy danego obiektu. Prawo to nie gwarantuje jednak odczytu własności widzianego obiektu
Create	dysponent może tworzyć obiekty podrzędne, prawo to dotyczy tylko obiektów klasy Container
Delete	dysponent może usunąć obiekt z bazy NDS
Rename	dysponent może zmienić nazwę obiektu
Supervisor	dysponent ma wszystkie wyżej wymienione prawa

3.5.2. Zabezpieczenia na poziomie struktury bazy obiektowej

- Prawa do cech, są to prawa określające jakie czynności dysponent może wykonywać względem cech obiektu
- Prawa do cech można nadawać na dwa sposoby: do wszystkich cech jednocześnie (*All Properties*) lub tylko do wybranych cech (*Selected Properties*).

3.5.2. Zabezpieczenia na poziomie struktury bazy obiektowej

Uprawnienie	Znaczenie uprawnienia
Compare	dysponent może porównywać wartość cechy z daną wartością
Read	dysponent może odczytać wartość cechy
Write	dysponent może modyfikować wartość cechy
Add Self	dysponent może dodawać lub usuwać siebie jako wartość cechy
Supervisor	dysponent ma wszystkie wyżej wymienione prawa

3.5.2.1. Reguły dotyczące praw dysponenckich

- Każdy obiekt posiada własność o nazwie **ACL** (*Access Control List*). Własność ta zawiera listę wszystkich dysponentów tego obiektu i dysponentów jego własności z wyszczególnieniem posiadanych przez nich praw.
 - Jeżeli obiekt typu grupa użytkowników (*Group object*) jest dysponentem innego obiektu (lub jego własności), to każdy użytkownik należący do tej grupy uzyskuje w systemie takie same prawa.

3.5.2.1. Reguły dotyczące praw dysponenckich

- Istnieje specjalny obiekt umowny o nazwie [*Publc*]. Formalnie nie jest on umieszczony w żadnym miejscu struktury NDS, ale za to ma specjalną cechę polegającą na tym, że **każdy obiekt w NDS ma takie same prawa jak on do katalogów, plików i innych obiektów.**
- Prawa do zasobów oraz prawa dysponenckie obiektu [*Publc*] są dostępne dla każdego i nie wymagają logowania się do systemu.

3.5.2.1. Reguły dotyczące praw dysponenckich

- Każdy użytkownik posiada własność o nazwie *Security Eguivalence* co oznacza „*ma takie prawo jak ...*”. Jest to lista obiektów w strukturze NDS, których prawa dysponenckie posiada dany obiekt.
- Prawa dysponenckie podlegają dziedziczeniu.
- Są dwa rodzaje dziedziczenia:
 - dziedziczenie od przodków (*ancestory inheritance*)
 - i dziedziczenie polegające na spływaniu praw dysponenckich (*rights flow*).

3.5.2.1. Reguły dotyczące praw dysponenckich

- **Dziedziczenie od przodków** polega na tym, że jeżeli jakiś kontener zostaje dysponentem dowolnego obiektu, to uprawnienia tego kontenera przechodzą na zawarte w nim obiekty.
- **Dziedziczenie poprzez spływanie praw** dotyczy sytuacji gdy dany dysponent uzyskuje określone prawa do obiektu typu kontener to tym samym uzyskuje takie same prawa do obiektów zawartych w tym kontenerze.

3.5.2.1. Reguły dotyczące praw dysponenckich

- Prawa dysponenckie do obiektu (lub jego własności) odziedziczone z kontenerów nadrzędnych mogą zostać anulowane (zmienione na inne), przez ponowne jawne nadanie dysponenckie do tego obiektu (lub jego własności).
- Takie nadanie jest silniejsze od dziedziczenia praw i powoduje, że od tego obiektu począwszy w głąb struktury (o ile był to kontener) spływają tylko te prawa dysponenta, które pojawiły się w wyniku jawnego nadania.

3.5.2.2. Filtrowanie praw dysponenckich

- Ponieważ prawa dysponenckie spływają w głąb struktury, można na dowolny obiekt NDS (lub na jego wszystkie własności) założyć specjalny filtr (*Inherited Rights Filter*), który może zablokować spływanie tych praw.
- Domyślnie filtr **IRF** zawiera wszystkie możliwe prawa, a wykluczenie określonych praw z listy powoduje ich zablokowanie.

3.5.2.2. Filtrowanie praw dysponenckich

- W przypadku, gdy do jakiegokolwiek obiektu (lub wszystkich jego własności) pojawi się jawne nadanie dysponenckie, to te prawa dysponenta obowiązują go niezależnie od tego, jaki filtr IRF posiada ten obiekt lub wszystkie jego własności

3.5.2.3. Domyślne przypisania dysponenckie

- W przypadku tworzenia nowych obiektów w strukturze NDS system samoczynnie nadaje pewne prawa do nich innym obiektom:
 - dla utworzonego nowego obiektu typu *USER*
 - obiekt [*Root*] otrzymuje do niego prawo obiektowe *Browse*
 - obiekt [*Root*] otrzymuje do jego własności *Network Adress* i własności *Group Membership* prawo *Read*

3.5.2.3. Domyślne przypisania dysponenckie

- nowoutworzony obiekt otrzymuje do samego siebie prawo obiektowe *Browse*, prawo *Read* do wszystkich swoich własności oraz prawo *Write* do swoich własności *Login Script* i *Print Job Configuration*
- obiekt [*Public*] otrzymuje prawo *Read* do jego własności *Message Server*
- obiekt, który go utworzył, otrzymuje do niego prawo obiektowe *Supervisor*

3.5.2.3. Domyślne przypisania dysponenckie

- dla nowoutworzonego obiektu typu *SERVER*
 - obiekt, który go utworzył, otrzymuje do niego prawo obiektowe *Supervisor*
- dla nowoutworzonego obiektu typu *VOLUMEN*
 - obiekt [*Root*] otrzymuje do jego własności *Host Server Name* i *Host Resource Name* prawo *Read*

3.5.2.3. Domyślne przypisania dysponenckie

- w przypadku obiektu [**Root**] utworzonego automatycznie przy instalacji systemu
 - użytkownik **ADMIN** otrzymuje do niego prawo obiektowe *Supervisor*
 - obiekt [**Public**] otrzymuje do niego prawo obiektowe *Browse*

3.5.2.4. Prawa efektywne

- Odpowiedź na pytanie „*Co obiekt X może naprawdę zrobić z obiektem Y lub jego własnościami ?*” jest prosta tylko z pozoru, gdyż jego efektywne prawa wynikają z:
 - jawnych nadań dysponenckich,
 - przynależności do określonych grup,
 - bycia równoważnikiem innych obiektów (np. *Organization Role*)
 - dziedziczenia praw w strukturze,
 - ograniczeń wynikających z zastosowanych filtrów IRF.

3.5.3. Zabezpieczenia dostępu do plików

- Na tym poziomie ochronie podlegają pliki lub katalogi umieszczone na dyskach sieciowych, a chroni się je przed nieuprawnionym dostępem obiektów NDS.
- Logika zabezpieczeń systemu plikowego i logika zabezpieczeń struktury NDS są bardzo podobne, ale są to zupełnie niezależne systemy zabezpieczeń.
- Cechy, które istotnie różnią zabezpieczenia systemu plikowego od zabezpieczeń na poziomie bazy NDS to:

3.5.3. Zabezpieczenia dostępu do plików

- ❖ w przypadku posiadania praw obiektowych do voluminów lub ich własności, nie przenoszą się one przez dziedziczenie na zawarte w nich katalogi i pliki,
- ❖ inny, dostosowany do potrzeb systemu plikowego, zestaw praw dysponenckich,
- ❖ inne reguły rządzące działaniem prawa *Supervisory*, uniemożliwiające jawne odebranie go, gdy zostało odziedziczone z katalogu nadrzędnego,

3.5.3. Zabezpieczenia dostępu do plików

- ❖ inne działanie filtru IRF założonego na katalog lub plik uniemożliwiająca blokowanie specjalnego prawa *Supervisory*,
- ❖ możliwość zakładania na pliki i katalogi tzw. atrybutów, czyli własności precyzujących ich sposób traktowania przez system bądź zakresu dostępu do nich bez względu na uprawnienia dysponenckie,
- ❖ domyślne nadanie praw dysponenckich do systemu plikowego kontenerowi, w którym zawarty jest wolumin *SYS*::

3.5.3. Zabezpieczenia dostępu do plików

Zestaw praw dysponenckich do plików i katalogów

Uprawnienie	Znaczenie uprawnienia
Read (R)	dysponent może otworzyć zbiór (katalog) i odczytać dowolną jego część lub całość
Write (W)	dysponent może otworzyć zbiór i poddać go edycji (dopisywać, usuwać lub zmieniać jego zawartość)
File Scan (F)	dysponent może zobaczyć nazwę zbioru lub przeglądać zawartość katalogu.
Create (C)	dysponent może w katalogu, do którego ma takie prawo, zakładać nowe katalogi lub pliki

3.5.3. Zabezpieczenia dostępu do plików

Zestaw praw dysponenckich do plików i katalogów

Uprawnienie	Znaczenie uprawnienia
Erase (E)	dysponent może skasować plik lub katoalog do którego posiada to prawo
Modify (M)	dysponent może zmieniać nazwę pliku (katalogu) lub jego atrybuty
Access Control (A)	prawo to pozwala zmieniać prawa do plików lub katalogów dla innych użytkowników (nie dotyczy jednak prawa <i>Supervisory</i>)
Supervisor (S)	prawo to nadaje automatycznie wszystkie pozostałe prawa

3.5.3. Zabezpieczenia dostępu do plików

Atrybuty plików:

Nazwa atrybutu	Znaczenie
A (Archive Needed)	oznacza dla systemu konieczność archiwizacji pliku z takim atrybutem
ci (Copy Inhibit)	uniemożliwia skopiowanie pliku (dotyczy tylko plików systemu Macintosh)
Cc (Can't Compress)	nie pozwala na kompresja pliku (dotyczy kompresji wykonywanej przez system NetWare)
Dc (Don't Compress)	umożliwia kompresje pliku (patrz powyżej)

3.5.3. Zabezpieczenia dostępu do plików

Atrybuty plików:

Nazwa atrybutu	Znaczenie
Di (Delete Inhibit)	uniemożliwia usunięcie pliku
Dm (Don't Migrate)	zapobiega przenoszeniu pliku na urządzenie do archiwizacji
Ds (Don't Suballocate)	powoduje zapisywanie pliku w całych blokach, niezależnie od tego czy przydzielanie małych bloków jest włączone
Ic (Immediate Compress)	powoduje kompresję pliku w czasie jego zapisywania (kompresja Netware)

3.5.3. Zabezpieczenia dostępu do plików

Atrybuty plików:

Nazwa atrybutu	Znaczenie
M (Migrated)	oznacza pliki które zostały przeniesione na urządzenie do magazynowania o dużej pojemności
H (Hidden)	uniemożliwia wyświetlenie pliku, jeśli użytkownik ma prawo File Scan do program NDIR pokaże plik
I (Indexed)	umożliwia szybkie indexowanie pliku w tablicy FAT
N (Normal)	używany przez polecenia FLAG do przypisywania domyślnego zestawu atrybutów (S i Rw)

3.5.3. Zabezpieczenia dostępu do plików

Atrybuty plików:

Nazwa atrybutu	Znaczenie
P (Purge)	powoduje, że plik zostanie zlikwidowany natychmiast po usunięciu i nie będzie można go odzyskać przy pomocy narzędzia SALVAGE. Skasowane pliki bez tego atrybutu można najczęściej odzyskać
Ri (Rename Inhibit)	uniemożliwia zmianę nazwy pliku lub katalogu
Ro (Read Only)	uniemożliwia pisanie w pliku, zmianę nazwy lub usunięcie, automatycznie ustawia atrybuty Di oraz Ri

3.5.3. Zabezpieczenia dostępu do plików

Atrybuty plików:

Nazwa atrybutu	Znaczenie
Rw (Read/Write)	umożliwia odczytywanie i pisanie w pliku, jest ustawiany samoczynnie jak usuniemy atrybut Ro
S (Shareable)	umożliwia dostęp do pliku wielu użytkownikom jednocześnie
Sy (System)	oznacza plik systemowy - kombinacja atrybutów H i Ro
T (Transactional)	oznacza pliki TTS, chronione przez system śledzenia transakcji systemowych

3.5.3. Zabezpieczenia dostępu do plików

Atrybuty plików:

Nazwa atrybutu	Znaczenie
X (Execute Only)	uniemożliwia usuwanie, zmianę nazwy, kopiowanie pliku; atrybut ten nie może zostać usunięty

3.5.3. Zabezpieczenia dostępu do plików

Domyślne nadania dysponenckie dla kontenera zawierającego wolumin **SYS**

- W przypadku, gdy w kontenerze zostanie zainstalowany obiekt typu **CN** jakim jest wolumin **SYS**, kontener ten otrzymuje domyślne prawa *Read* i *File Scan* do katalogu **\PUBLIC** oraz prawo *Create* do katalogu **\MAIL**, zawartych w tym woluminie.
- Oznacza to, że na mocy dziedziczenia po przodkach każdy inny obiekt w tym kontenerze, a także w kontenerach podległych, posiada takie same prawa do tych katalogów.

3.6. Skrypt logowania

- Skrypty logowania są to polecenia wykonywane podczas podłączania się użytkownika do sieci.
- Zawierają one polecenie przyłączania sieciowych drukarek, mapowania dysków (czyli logiczne podłączanie napędów sieciowych itp.
- Wyróżniamy cztery rodzaje skryptów logowania:
 - Kontenerowy skrypt logowania dla kontenera w którym znajduje się użytkownik;
 - Profilowy skrypt logowania (jeśli jest przypisany)

3.6. Skrypt logowania

- Skrypt logowania użytkownika (jeśli został on zdefiniowany)
- Domyślny skrypt logowania (zostanie wykonany jeśli system NetWare nie znajdzie skryptu użytkownika)
- ❖ **Kontenerowy skrypt logowania** - można utworzyć dla każdego obiektu typu **Organization** lub **Organizational Unit**
- ❖ **Profilowy skrypt logowania** - obiekt typu profil jest obiektem NDS służącym do przypisywania tych samych ustawień wielu użytkownikom, czyli między innymi skryptów logowania,

3.6. Skrypt logowania

- ❖ **Skrypt logowania użytkownika** - skrypt ten może służyć np. do wykonywania dodatkowych mapowań dla konkretnych użytkowników, przekierowań dodatkowych drukarek itp.
- ❖ **Domyślny skrypt logowania** - jest częścią narzędzia login i nie można go edytować. Jest wykonywany gdy nie ma skryptu użytkownika. Skrypt domyślny można wyłączyć aby jego polecenia nie kolidowały np. z poleceniami skryptu kontenerowego.

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
#	Znak poprzedzający polecenie zewnętrzne (tj. program typu EXE, COM lub BAT. Pełna składnia to: <i>#[ścieżka]program [parametry]</i>	<i>#NLIST USER /A</i> - wyświetla zawartość katalogu;
ATTACH	Przyłączenie się do serwera. Składnia: <i>ATTACH [FSN[/UNAME[:HASŁO]]]</i> gdzie: <i>FSN</i> – nazwa serwera, <i>UNAME</i> – nazwa użytkownika; <i>HASŁO</i> – hasło użytkownika	<i>ATTACH UWM/ADMIN;SEKRET</i> - przyłączenie do serwera <i>UWM</i> jako <i>ADMIN</i> z hasłem <i>SEKRET</i>

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
BREAK	Przełącznik pozwalający (ON) lub zabraniający (OFF) przerwać program zgłoszenia. Składnia <i>BREAK [ON OFF]</i>	<i>BREAK OFF</i> - zablokowanie przerwania programu zgłoszenia przez ^C lub ^Break
COMSPEC	Wskazanie procesora poleceń. Składnia: <i>COMSPEC=[ścieżka]COMMAND.COM</i>	<i>COMSPEC=S2:COMMAND.COM</i> - program procesora umieszczony w katalogu na drugiej pozycji ścieżki przeszukiwania
CLS	Czyszczenie ekranu. Składnia: <i>CLS</i>	<i>CLS</i>

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
CONTEXT	Zmienia bieżący kontekst na podany. Składnia: <i>CONTEXT [kontekst]</i>	<i>CONTEXT .OU=KADRY.O=FIRMA</i>
DISPLAY	Wyświetla na ekranie zawartość zbioru. Składnia: <i>DISPLAY [ścieżka]zbiór</i>	<i>DISPLAY H:MOTTO.TXT</i>
FDISPLAY	Wyświetla na ekranie zawartość zbioru z pominięciem znaków niedrukowalnych. Składnia: <i>DISPLAY [ścieżka]zbiór</i>	<i>DISPLAY G:\WYDRUKI\RAP01.TXT</i>

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
<i>DOS BREAK</i>	Dopuszcza lub zabrania przerwanie wykonania programu. Składnia: <i>DOS BREAK [ON OFF]</i>	<i>DOS BREAK OFF</i> - zabrania przerywania programu DOS przez ^C lub ^Break
<i>DOS SET</i>	Przypisanie wartości zmiennej środowiskowej systemu DOS. Składnia: <i>DOS SET zmienna="ciąg_znaków"</i>	<i>DOS SET PROMPT="\$P\$G"</i> - ustawienie zmiennej prompt
<i>DOS VERIFY</i>	Wymusza weryfikację poprawności kopii przy kopiowaniu na napędy lokalne. Składnia: <i>DOS VERIFY [ON OFF]</i>	<i>DOS VERIFY OFF</i> - rezygnacja z weryfikacji poprawności kopii

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
<i>DRIVE</i>	Określa który z napędów będzie napędem bieżącym po wykonaniu. Składnia: <i>DRIVE [napęd:]</i>	<i>DRIVE K:</i> - bieżącym napędem jest katalog oznaczony przez K:
<i>EXIT</i>	Kończy wykonanie skryptu logowania i przechodzi do wykonania programu zewnętrznego. Składnia <i>EXIT „program”</i>	<i>EXIT NETADMIN</i> - przerywa skrypt logowania i uruchamia program NETADMIN
<i>FIRE PHASERS</i>	Wydawanie dźwięków przez stację. Składnia: <i>FIRE PHASERS n TIMES</i> ($n \leq 9$)	<i>FIRE PHASERS 5 TIMES</i> - wydanie pięciu sygnałów dźwiękowych

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
GOTO	Wykonanie skoku do linii skryptu rozpoczynającej się od etykiety. Składnia: <i>GOTO etykieta</i>	<i>GOTO WARIANT_2:</i> - skok do linii rozpoczynającej się od etykiety WARIANT_2
IF ... THEN	Instrukcja warunkowa. Składnia: <i>IF warunek [AND OR] warunek THEN</i> <i>Instrukcje</i> <i>[ELSE]</i> <i>Instrukcje</i> <i>[ENDIF]</i>	<i>IF DAY="01" AND MONTH=04 THEN</i> <i>WRITE "Życzenia imieninowe"</i> <i>ELSE</i> <i>WRITE „Zwykły dzień – do roboty”</i> <i>ENDIF</i>

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
<i>INCLUDE</i>	Wykonanie programu zawartego w wymienionym zbiorze lub pochodzącego od innego obiektu w bazie NDS. Składnia: <i>INCLUDE [ścieżka]zbiór obiekt</i>	<i>INCLUDE</i> <i>.CN=JAN.O=FIRMA</i> - wykonanie skryptu użytkownika JAN umieszczonego bezpośrednio w kontenerze FIRMA
<i>MACHINE</i>	Ustala zawartość zmiennej środowiskowej DOS o nazwie MACHINE. Składnia: <i>MACHINE = nazwa</i>	<i>MACHINE=IBM_PC</i> - ustala wartość zmiennej na IBM_PC

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
MAP	Definiuje tymczasowe oznaczenia literowe oraz ścieżki przeszukiwań w woluminach. Steruje trybem wykonania polecenia. Składnia: <i>MAP [opcja] [napęd:=ścieżka]</i>	<i>MAP</i> <i>H:=UWM_APL:\USERS\JAN</i> <i>MAP S16:=C:\NC</i> – katalog NC dysku lokalnego będzie ostatnią ścieżką
MAP ERRORS	Włączenie lub wyłączenie komunikatów o błędach w przypisaniu napędów. Składnia: <i>MAP ERRORS ON OFF</i>	

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
<i>MAP DISPLAY</i>	Włączenie lub wyłączenie komunikatów o przypisanych napędach. Składnia: <i>MAP DISPLAY ON OFF</i>	
<i>NO_DEFAULT</i>	Polecenie zabraniające wykonania domyślnego skryptu użytkownika - może być umieszczone w kontenerowym lub profilowym skrypcie zgłoszenia. Składnia: <i>NO_DEFAULT</i>	

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
SWAP	Polecenie przeniesienia kodu programu LOGIN do pamięci górnej albo na dysk, jeżeli w trakcie wykonania polecenia # zabrakło pamięci. Składnia: <i>SWAP [ścieżk]</i>	<i>SWAP</i> <i>FSN_SYS:\SWAPDIR</i> - przeniesienie kodu LOGIN do katalogu <i>FSN_SYS:\SWAPDIR</i>
NOSWAP	Polecenie zabraniające przeniesienia kodu programu LOGIN do pamięci górnej albo na dysk. Składnia: <i>NOSWAP</i>	

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
<i>PAUSE</i>	Zatrzymanie wykonania programu zgłoszeniowego do czasu naciśnięcia dowolnego klawisza. Składnia: <i>PAUSE</i>	
<i>REMARK [* :]</i>	Oznaczenie w skrypcie logowania linii komentarza. Składnia: <i>REMARK [tekst]</i>	<i>REMARK To jest komentarz</i> <i>* to też jest komentarz</i>
<i>SET</i>	Ustawienie zmiennej środowiskowej systemu DOS. Składnia: <i>SET zmienna="ciąg_znaków"</i>	<i>SET TEMP="C:\TEMP"</i>

3.6. Skrypt logowania

Polecenia stosowane w skryptach logowania:

Polecenie	Funkcja polecenia	Przykład zastosowania
<i>SET TIME</i>	Ustala czy stacja przy rejestracji do sieci ma ustawić swój zegar zgodnie z czasem obowiązującym w sieci, czy też nie. Składnia: <i>SET TIME ON OFF</i>	
<i>WRITE</i>	Polecenie wypisania na ekranie ciągu znaków. Składnia: <i>WRITE "[tekst] [%zmienna][; zmienna]</i>	<i>WRITE „Witaj ”;</i> <i>login_name</i> <i>Lub</i> <i>WRITE “Witaj</i> <i>%login_name”</i>

3.6. Skrypt logowania

Zmienne stosowane w skryptach

- Program zgłoszeniowy może posługiwać się zmiennymi do których można odwoływać się wyłącznie poprzez nazwę.
- Wszelkie możliwe do wykorzystania zmienne mają wartość tekstową i w przypadku użycia musi być podawana w cudzysłowie.
- W programach zgłoszeniowych dostępne są dwa rodzaje zmiennych:

3.6. Skrypt logowania

- **Zmienne predefiniowane**, a więc o wartości określonej przez system,
- **Zmienne środowiskowe** systemu DOS, które mogą być zmieniane poleceniem SET. Odwołanie do tych zmiennych wymaga jednak by ich nazwy były ujęte w nawiasy ostre (np. *<zmienna_środowiskowa>*)

3.6. Skrypt logowania

Zmienna	Opis
DAY	Numer dnia miesiąca („01”, „02”,”31”)
DAY_OF_WEEK	Nazwa dnia tygodnia (np. „Monday”)
MONTH	Numer miesiąca („01”, „02”,”12”)
MONTH_NAME	Nazwa miesiąca (np. „February”)
NDAY_OF_WEEK	Numer dnia tygodnia (“1”=”niedziela”, “2”=”pon.” itd.)
SHORT_YEAR	Dwie ostatnie cyfry roku (np. „99”)
YEAR	Cztery cyfry roku (np. „2001”)
AM_PM	Wartość „am” gdy przed południem, „pm” po godz. 12-stej
GREETING_TIME	Pora dnia (“morning”, „afternoon”, „evening”)
HOUR	Godzina doby od 1 do 12 („1”, „2”,”12”)

3.6. Skrypt logowania

Zmienna	Opis
HOUR24	Godzina doby od 1 do 23 („00”, „01”, „02”,”23”)
MINUTE	Minuta bieżącej godziny („00”, „01”, „02”,”59”)
SECOND	Sekunda bieżącej minuty („00”, „01”, „02”,”59”)
FULL_NAME	Pełna nazwa użytkownika (wartość własności Full Name)
LAST_NAME	Zawartość własności „nazwisko” użytkownika w NDS
LOGIN_NAME	Pierwsze 8 znaków używanych przy rejestracji do systemu
MEMBER OF „grupa”	Prawda gdy rejestrujący się użytkownik należy do podanej grupy, w przeciwnym razie fałsz

3.6. Skrypt logowania

Zmienna	Opis
NOT_MEMBER OFR „grupa”	Fałsz gdy rejestrujący się użytkownik należy do podanej grupy, w przeciwnym razie prawda
PASSWORD_EXPIRES	Ilość dni pozostałych do wygaśnięcia hasła (np. „24”)
USER_ID	Numer wewnętrzny nadawany każdemu użytkownikowi przez system
FILE_SERVER	Nazwa domyślnego serwera
NETWORK_ADDRESS	Ośmioznakowy numer logiczny okablowania, w którym pracuje stacja
MACHINE	Nazwa określająca typ komputera
OS	Typ systemu operacyjnego DOS używanego na stacji
OS_VERSION	Wersja systemu DOS używanego na stacji

3.6. Skrypt logowania

Zmienna	Opis
P_STATION	Dwunastoznakowy numer karty sieciowej stacji
STATION	Numer logiczny sesji nawiązanej przez stację roboczą
ERROR_LEVEL	Numer błędu powstałego w trakcie wykonywania programu zgłoszenia („0” – oznacza brak błędu)
%n	Kolejny n-ty parametr polecenia LOGIN

3.7. Drukowanie w sieci

Dla uzyskania sprawnie działającego systemu drukarskiego należy wykonać następujące czynności:

1. utworzyć przynajmniej jeden obiekt typu kolejka i wypełnić lub uzupełnić następujące własności:
 - *nazwa* - (*name*) nazwa kolejki,
 - *wolumin* - miejsce fizycznego przechowywania kolejki

3.7. Drukowanie w sieci

2. utworzyć przynajmniej jeden obiekt typu drukarka i wypełnić lub uzupełnić następujące własności:
 - *nazwa* - (*name*) nazwa drukarki,
 - *typ* - (*type*) typ połączenia drukarki (równoległe, szeregowe, inne),
 - *port* - port komputera do którego będzie przyłączona (najczęściej LPTx lub COMx),
 - *przerwanie* - (*interrupt*) numer przerwania używany w komunikacji z wybranym portem,

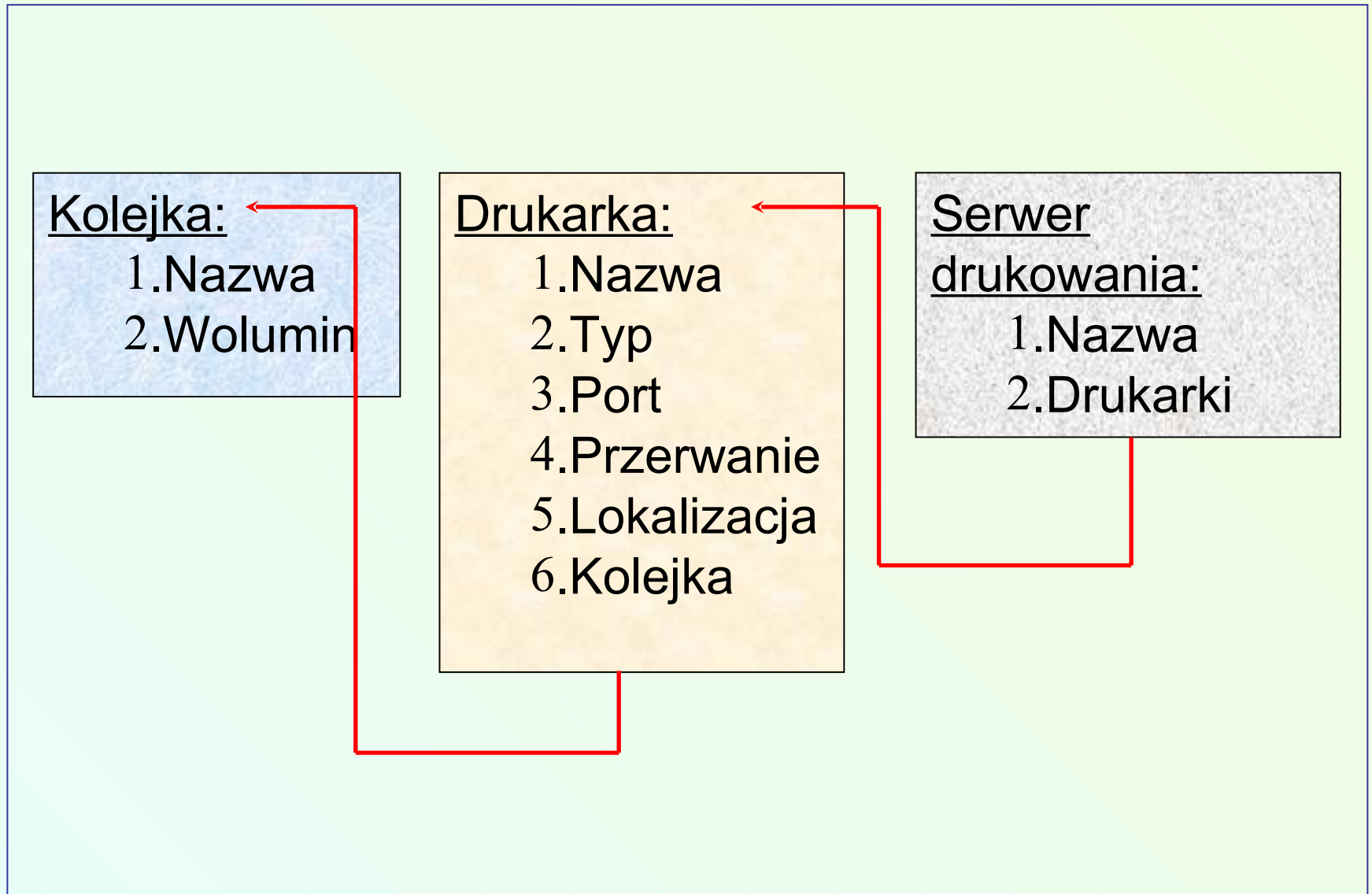
3.7. Drukowanie w sieci

- *lokalizacja* - (*location*) określenie czy drukarka będzie przypięta do tego samego komputera, na którym uruchomiono serwer drukowania (*Auto Load*), czy do jakiegoś innego komputera (*Manual Load*),
- *kolejki* - (*queues*) lista istniejących kolejek, które będą obsługiwane przez drukarkę

3.7. Drukowanie w sieci

3. utworzyć przynajmniej jeden obiekt typu **Print Server** i wypełnić następujące własności:
 - *nazwa* - (*name*) nazwa serwera drukowania,
 - *drukarki* - (*printers*) lista istniejących drukarek, które serwer ma obsłużyć.

3.7. Drukowanie w sieci



3.7. Drukowanie w sieci

4. Kolejnym krokiem jest uaktywnienie systemu druku, które polega na :
- *uruchomieniu procesu serwera drukowania jako modułu ładowalnego na wybranym serwerze NetWare poleceniem:*
***LOAD PSERVER** nazwa_serwera_druku*
 - *uruchomienie programów obsługujących pracę drukarek poleceniem:*
***NPRINT** nazwa_drukarki*

3.7. Drukowanie w sieci

lub w trybie konwersacyjnym:

- *wybrać serwer drukowania z przedstawionej listy aktywnych serwerów,*
- *wybrać z listy drukarek (**Manual Load**) dotychczas nie podłączoną*
- w przypadku, gdy drukarka podłączona jest do jednego z serwerów, należy z jego konsoli wydać polecenie:

LOAD NPrinter nazwa_drukarki

lub w trybie konwersacyjnym

LOAD NPrinter

3.7. Drukowanie w sieci

Nadzorowanie systemu druku:

- Możliwe jest usuwanie zadań z kolejki, „ręczne” dodawanie nowych zadań lub zmienianie kolejności drukowania, czasowe lub trwałe wstrzymanie wydruku zadań w kolejce, manipulowanie parametrami polecenia CAPTURE wydanego jeszcze przed utworzeniem zadania.
- Czynności te umożliwia opcja **Print Queues/Print Jobs** programu **PCONSOLE**.

3.7. Drukowanie w sieci

- Możliwe jest również nadzorowanie stanu drukarek sieciowych.
- Można usuwać właśnie drukujące się zadania, chwilowo zatrzymywać pracującą drukarkę, cofnąć zadanie wstecz do określonej pozycji (strony, kopii lub numeru bajtu).
- Możliwe jest też poinformowanie systemu, że właśnie została zmieniona formatka papieru na drukarce na inną i określenie sposobu obsługi zadań z różnymi formatkami.

3.7. Drukowanie w sieci

- Ostatnia możliwość to ustalenie kogo i co jaki czas informować o stanie drukarki (komunikaty typu „brak papieru”, „drukarka wyłączona”, „zadanie zakończone”).
- Czynności te mogą być wykonane opcją **Printers/Status** modułu **PSERVER.NLM** na konsoli serwera lub (pod warunkiem, że serwer druku jest aktywny) opcją **Print Servers/Printer/Printer Status** programu **PCONSOLE**.

3.7. Drukowanie w sieci

- Możliwe jest zatrzymanie serwera drukowania, czyli jego natychmiastowy **DOWN** lub po wydruku aktywnych zadań z kolejki.
- Czynność tą można wykonać w opcji **Print Servers/ Information and Status** programu **PCONSOLE** lub opcją **Print Server Information/Current Status** modułu **PSERVER.NLM**.

3.7. Drukowanie w sieci

- Możliwy jest podgląd stanu aktualnie drukowanego zadania (np. która kopia się drukuje, ile procent zadania jest już gotowe, z jakiej kolejki drukujące się zadanie pochodzi, jak się nazywa, jakiej używa formatki itp.).
- Możliwości te zapewnia opcja **Printers/Printer Status** modułu **PSERVER.NLM**.