

IBM

@server

iSeries

Administrowanie

Wersja 5 wydanie 3





@server

iSeries

Administrowanie

Wersja 5 wydanie 3

Uwaga

Przed rozpoczęciem korzystania z tych informacji i opisanego w nich produktu należy przeczytać “Uwagi”, na stronie 101.

Wydanie szóste (sierpień 2005)

Niniejsze wydanie ma zastosowanie do wersji 5, wydania 3, modyfikacji 0 programu iSeries Access for Windows (numer produktu 5722-XE1) oraz do wszystkich kolejnych wydań i modyfikacji, o ile nie zostanie wskazane inaczej. Wersja ta może nie działać prawidłowo na niektórych modelach komputerów o zredukowanej liczbie instrukcji (RISC), jak również na modelach CISC.

© Copyright International Business Machines Corporation 1998, 2005. Wszelkie prawa zastrzeżone.

Spis treści

Administrowanie iSeries Access for Windows 1

Co nowego w wersji V5R3	2
Drukowanie tego dokumentu	3
Środowiska sieciowe iSeries Access for Windows.	4
Microsoft Windows Terminal Server.	4
Korzystanie z iSeries Access for Windows w środowisku trójwarstwowym	5
Używanie serwera Microsoft Transaction Server (MTS).	5
Dostęp do usług Access iSeries z warstwy pośredniczącej	6
Dodawanie konfiguracji TCP/IP dla wszystkich użytkowników	7
Określenie położenia plików PC5250 dla wszystkich użytkowników	8
Profile użytkowników dla komputerów PC z wieloma użytkownikami	8
Administrowanie ODBC	9
Ogólne informacje o sterowniku ODBC iSeries Access	9
Konfigurowanie systemu iSeries Access do obsługi sterownika ODBC	10
Dodawanie systemu lokalnego do katalogu RDB.	11
Podanie źródła danych ODBC	11
Ochrona ODBC w iSeries Access for Windows	12
Ryzykowne strategie ochrony ODBC	12
Strategie ochrony programu ODBC.	13
Informacje pokrewne o ochronie ODBC	14
Rozwiązywanie problemów z ODBC	14
Diagnostyka ODBC i narzędzia wydajności	15
Komunikaty o błędach ODBC iSeries Access.	16
Rozwiązywanie problemów dotyczących połączenia z serwerem iSeries.	18
Typowe błędy ODBC.	20
Zbieranie informacji dla działu wsparcia IBM	21
Administrowanie serwerem hosta	22
Serwery OS/400	23
Lista serwerów hosta według funkcji iSeries Access for Windows	24
Serwer plików	25
Serwer baz danych	26

Serwer kolejek danych	30
Wydruk sieciowy	30
Serwer centralny	31
Serwer komend zdalnych i rozproszonych wywołań programów	32
Serwer wpisywania się do systemu	32
Program odwzorowujący porty serwera	32
Rozszerzony dynamiczny zdalny serwer SQL (QXDAEDRSQL).	32
Serwer DRDA/DDM	33
Korzystanie z serwerów hosta OS/400	33
Nawiązywanie komunikacji klient/serwer	34
Podsystemy w serwerze iSeries	38
Wartości systemowe serwera iSeries	49
Identyfikowanie zadań serwera w serwerze iSeries	52
Używanie programów EZ-Setup i iSeries Navigator z serwerami hosta	54
Używanie programów obsługi wyjścia serwera	55
Rejestrowanie programów obsługi wyjścia	55
Pisanie programów obsługi wyjścia.	58
Parametry programów obsługi wyjścia.	59
Przykłady: Programy obsługi wyjścia	75
Administrowanie iSeries NetServer.	90
Konfigurowanie ograniczeń użytkowników za pomocą strategii i funkcji Administrowania aplikacjami	90
Przegląd strategii iSeries Access for Windows	91
Typy i zasięgi strategii	92
Konfigurowanie systemu w celu korzystania ze strategii	93
Konfigurowanie serwera iSeries ^(TM) pod kątem strategii	93
Konfigurowanie klientów PC pod kątem strategii	93
Tworzenie plików strategii	93
Lista strategii iSeries Access for Windows	95
Strategie dla poszczególnych funkcji	95
Strategie według szablonu	98
Administrowanie Secure Sockets Layer (SSL)	99

Dodatek. Uwagi 101

Znaki handlowe	102
Warunki pobierania i drukowania publikacji.	103
Informacje dotyczące kodu	103

Administrowanie iSeries Access for Windows

W tej sekcji zakłada się, że użytkownik zna program iSeries^(TM) Access for Windows^(R) i zainstalował go w swoim systemie. Poniższe dokumenty zawierają informacje pokrewne dotyczące administrowania iSeries Access for Windows w środowisku klient/serwer:

“Co nowego w wersji V5R3” na stronie 2

Podsumowanie nowych funkcji opisanych w dokumencie dotyczącym administrowania dla tej wersji.

“Drukowanie tego dokumentu” na stronie 3

Informacje dotyczące przeglądania i drukowania dokumentu PDF iSeries Access for Windows.

“Środowiska sieciowe iSeries Access for Windows” na stronie 4

Opis środowisk sieciowych, w których może działać aplikacja iSeries Access for Windows. Sekcja ta zawiera również informacje dotyczące udostępniania klientom usług OS/400^(R) za pomocą iSeries Access for Windows w środowisku trójwarstwowym lub instalacji w wersji systemu operacyjnego Windows, która zapewnia obsługę zdalnego logowania się do systemu za pomocą Terminal Services. Opisane są tu także sposoby administrowania komputerami PC z wieloma użytkownikami.

“Administrowanie ODBC” na stronie 9

iSeries Access for Windows zawiera sterownik ODBC, który umożliwia aplikacjom wygodny dostęp do baz danych DB2^(R) UDB for iSeries w sieci. Temat ten zawiera ogólne informacje o ODBC, instrukcje konfigurowania sterownika i przewodnik rozwiązywania problemów.

Informacje i uwagi dotyczące pracy z interfejsami API ODBC zawiera sekcja Programowanie ODBC.

“Administrowanie serwerem hosta” na stronie 22

Temat ten opisuje serwery hostów najczęściej używane z programem iSeries Access for Windows i sposoby skutecznego zarządzania nimi i ich używania.

“Konfigurowanie ograniczeń użytkowników za pomocą strategii i funkcji Administrowania aplikacjami” na stronie 90

iSeries Access for Windows dostarcza wielu metod konfigurowania ograniczeń i profili. Obejmują one strategie utworzone za pomocą edytora strategii firmy Microsoft^(R) i funkcji Administrowanie aplikacjami programu iSeries Navigator.

» Aby zapoznać się z programem iSeries Access for Windows i opisem jego używania w sieci, należy przejrzeć dokument Wprowadzenie do iSeries Access for Windows. Informacje na temat instalowania i konfigurowania aplikacji iSeries Access for Windows zawiera sekcja Instalowanie i konfigurowanie. «

Dodatkowe informacje potrzebne do administrowania iSeries Access for Windows zawierają sekcje:

- “Administrowanie Secure Sockets Layer (SSL)” na stronie 99
- “Administrowanie iSeries NetServer” na stronie 90
- Programowanie w iSeries Access for Windows

Uwaga: Należy przeczytać sekcję Informacje dotyczące kodu, która zawiera istotne informacje prawne.

Co nowego w wersji V5R3

» Instalując wersję V5R3 iSeries^(TM) Access for Windows^(R), uzyskuje się możliwość zarządzania środowiskiem dzięki nowym funkcjom dodanym do dostawcy OLE DB lub nowemu dostawcy bazy danych .NET. Tak więc, dzięki wielu udoskonaleniom bazy danych, takim jak obsługa nowych typów danych i zwiększona dokładność obsługi liczb dziesiętnych można uzyskać dużo większą elastyczność.

Wszystkie funkcje bazy danych i dostępu do danych obsługują kod Unicode umożliwiający elektroniczne przesyłanie danych przez bariery utworzone przez różne schematy kodowania i zestawy znaków. Dodatkowo, funkcje te spełniają najnowsze standardy ISO i ANSI SQL.

Wiele udoskonaleń wersji V5R3 sprawia, że aplikacja iSeries Access for Windows pozostaje najlepszym z możliwych sposobem administrowania serwerem iSeries i jego bazami danych.

Nowe opcje dla administratorów iSeries Access for Windows to między innymi:

- **Opcje przesyłania danych**

Wersja V5R3 umożliwia kompresję danych, dzięki czemu prędkość ich przesyłania jest dużo większa, a aplikacje wykorzystują kod Unicode i jego nowy typ pliku tekstowego. Danymi liczbowymi można zarządzać z większą dokładnością, możliwe jest również zastosowanie nowych typów danych SQL: BINARY i VARBINARY. Tabele bazy danych DB2^(R) obsługują teraz z większą elastycznością dane UTF-8 i UTF-16.

Dzięki programowi Microsoft^(R) Excel, istnieje nowa obsługa standardu *komórek daty i godziny* oraz *konwersji danych liczbowych na znaki*, co czyni znacznie łatwiejszym zarządzanie przesyłaniem danych do i z serwerów w żądanym formacie. Dodatkowo, dużym ułatwieniem podczas administrowania są dodatki programu Excel, takie jak *lista ostatnio używanych żądań* oraz *ostatni katalog*.

- **Drukowanie i emulacja PC5250**

Wersja V5R3 aplikacji iSeries Access for Windows PC5250 jest integralnym wsparciem wersji 5.7 PC 5250. Jednym z kluczowych udogodnień wersji 5.7 jest to, że pozwala na zarządzanie dodatkowymi funkcjami dostępności, takimi jak numeryczna klawiatura popup, odwzorowanie kolorów oraz wizualne wskazywanie dźwięków. Jeśli istnieją określone potrzeby, takie jak obsługa dwukierunkowa LamAlef oraz odwzorowanie klawiatury Japanese USB 106, obsługa dodana w tej wersji będzie niezwykle pomocna. Do innych, bardziej ogólnych udoskonaleń należą znaczniki myszy oraz obsługa wydruków basic_ascii w formacie PDF i PDT, co daje dostęp do większej liczby opcji dla funkcji drukowania i wyświetlania.

- **iSeries Navigator**

W iSeries Navigator pojawiło się również wiele nowych opcji. Z ich opisem można zapoznać się w sekcji Co nowego w wersji V5R3 iSeries Navigator.

- **Przychodząca komenda zdalna**

Wersja V5R3 umożliwia pobranie informacji o profilu użytkownika dla komendy zdalnej uruchamianej w kontekście ochrony ID znanego użytkownika. Niektóre komendy mają teraz możliwości, których wcześniej nie miały ze względu na brak wymaganej autoryzacji do rejestracji użytkowników i zmiennych środowiskowych. Można ustawić i tę opcję, tak aby nie trzeba było ustawiać jej podczas każdego uruchomienia komendy. Więcej informacji i przykładów zawiera Podręcznik użytkownika (strona 3).

- **Interfejs ODBC**

W wersji V5R3, interfejs ODBC obsługuje typy danych BINARY i VARBINARY, dane UTF-8 i UTF-16 umożliwiające globalizację aplikacji, daje zwiększoną dokładność liczb dziesiętnych oraz oferuje zaawansowaną obsługę MTS.

- **Dostawcy baz danych**

- **Dostawca .NET** - Nowy dostawca IBM.Data.DB2.iSeries Data za pomocą pełnego zbioru klas .NET i typów danych umożliwia aplikacjom używającym struktury .NET dostęp do baz danych DB2^(R) UDB for iSeries^(TM). Uzupełnia ofertę istniejących dostawców OLE DB oraz umożliwia odnoszenie korzyści z nowszych technologii .NET przy odczycie i pobieraniu danych, dokonywaniu zmian oraz uruchamianiu komend SQL serwera w porównaniu do obiektów danych w środowisku ochrony serwera iSeries. Więcej informacji na ten temat sekcja Dostawca .NET. Więcej informacji zawiera sekcja Programowanie .NET.
- **Dostawca OLE DB** - Poza udoskonaleniem pełnozakresowej, elastycznej obsługi IBMDA400 podczas pracy z istniejącymi aplikacjami, użytkownik używając nowego dostawcy danych IBMDASQL ma teraz możliwość

rozwijania i zarządzania aplikacjami SQL z kontrolą transakcji i MTS. Jeśli nowe aplikacje wymagają dostępu na poziomie rekordu do kursorów typu forward-only i pobierania bloków (block fetches), jest to możliwe dzięki nowemu dostawcy danych IBM DARA. Dodano również obsługę kompresji danych SQL, obsługę pakietów, typów danych BINARY i VARBINARY, większą dokładność liczbową, kolejność sortowania NLSS oraz danych UTF-8 i UTF-16. Więcej informacji na ten temat zawiera artykuł Programowanie OLE DB.

- Szczegóły techniczne dotyczące dostawcy IBM.Data.DB2.iSeries zawiera publikacja *IBM^(R) DB2 UDB for iSeries .NET Provider Technical Reference*. Szczegóły dotyczące innych dostawców można znaleźć w publikacji **OLE DB Technical Reference**. Dostęp do tych dokumentów jest możliwy z *Programmer's Toolkit*, po przejściu poniższej ścieżki:
Start-> Programy-> IBM iSeries Access for Windows-> Programmer's Toolkit -> Programmer's Toolkit -> Common Interfaces

- **Konfigurowanie**

Począwszy od wersji V5R3, komendy cwback i cwbnv zapisują informacje kodując je za pomocą kodu Unicode, w wyniku czego pliki utworzone przez te programy użytkowe nie mogą być odtwarzane za pomocą wcześniejszych wersji komend cwrest lub cwbnv. Z tego powodu, dla komend cwback i cwbnv uruchamianych w wersji V5R3 udostępniono nowy parametr /c (Compatible - kompatybilny). Podanie tego parametru powoduje zapisanie informacji przy użyciu strony kodowej ANSI. Mogą być one wówczas odczytane przez starsze wersje komend cwrest i cwbnv. Wersja V5R3 komend cwrest i cwbnv może odtwarzać pliki zapisane zarówno w kodzie Unicode, jak i ANSI.

Uwaga: Ponieważ strona kodowa ANSI używana jest tylko po określeniu parametru /c, wszystkie znaki nie należące do niej mogą zostać utracone.

Inne informacje

Po zainstalowaniu iSeries Access for Windows, aby skorzystać z Podręcznika użytkownika, należy użyć następującej ścieżki z folderu iSeries Access for Windows: Start -> Programy -> IBM iSeries Access for Windows -> Podręcznik użytkownika.

Brak jest dalszego rozszerzenia interfejsu API C/C++ (Optimized SQL APIs) bazy danych. Być może w przyszłości jego obsługa zostanie całkowicie wyeliminowana. Zaleca się używanie jednej z innych technologii dostępu do baz danych.

Systemy operacyjne Windows 98 (wszystkie wydania) oraz Windows ME nie obsługują wersji V5R3 aplikacji iSeries Access for Windows. <<

Oznaczenie zmian i nowych informacji

W celu ułatwienia dostrzeżenia wprowadzonych zmian technicznych wprowadzono oznaczenia:

- Symbol >> oznacza początek nowych lub zmienionych informacji.
- Symbol << oznacza koniec tych informacji.

>> Aby znaleźć inne informacje dotyczące nowości lub wprowadzonych w tej wersji zmian, należy zapoznać się z dokumentem Informacje dla użytkowników. <<

Drukowanie tego dokumentu

Aby wyświetlić lub pobrać wersję PDF tego dokumentu, należy wybrać Administrowanie iSeries^(TM) Access for Windows^(R) (około 350 kB).



Aby zapisać plik PDF na stacji roboczej w celu jego dalszego wykorzystania:

1. W przeglądarce kliknij prawym przyciskiem myszy plik PDF (powyższy odsyłacz).
2. Kliknij **Zapisz jako...**, jeśli używana jest przeglądarka Internet Explorer. Kliknij **Zapisz odsyłacz jako...**, jeśli używana jest przeglądarka Netscape Communicator. <<

3. Przejdź do katalogu, w którym chcesz zapisać plik PDF.
4. Kliknij **Zapisz**.



Pobieranie programu Adobe Acrobat Reader

Do przeglądu lub drukowania dokumentów PDF konieczny jest program Adobe Acrobat Reader. Kopię programu można pobrać z serwisu WWWY firmy Adobe *(www.adobe.com/products/acrobat/readstep.html)  . 

Środowiska sieciowe iSeries Access for Windows

iSeries^(TM) Access for Windows^(R) oferuje kilka metod udostępniania użytkownikom usług iSeries. Zazwyczaj wymagają one bezpośredniego połączenia pomiędzy komputerem PC, na którym działa iSeries Access for Windows, a serwerem iSeries. Jednakże opisane poniżej metody umożliwiają wykorzystanie innych środowisk sieciowych.

- **“Microsoft Windows Terminal Server”**
Microsoft^(R) Windows Terminal Server jest funkcją umożliwiającą uruchomienie wielu jednoczesnych sesji klientów na jednym serwerze Windows. Umożliwia połączenie z wielu platform klienta, nie tylko Windows, lecz również stacji sieciowych, UNIX^(R), Linux, DOS, OS/2^(R) i innych. Instalując iSeries Access for Windows w serwerze Windows można zapewnić dostęp do usług iSeries ze stacji roboczych, w których nie zainstalowano produktu iSeries Access for Windows.
- **“Korzystanie z iSeries Access for Windows w środowisku trójwarstwowym” na stronie 5**
Instalując iSeries Access for Windows w warstwie pośredniczącej środowiska trójwarstwowego można zapewnić całej gamie stacji roboczych dostęp do usług serwera iSeries. Ponadto środowiska trójwarstwowe udostępniają dodatkowe usługi, takie jak rozszerzone zarządzanie transakcjami.


iSeries Access for Windows umożliwia również administrowanie komputerami PC z wieloma użytkownikami:


- **“Dodawanie konfiguracji TCP/IP dla wszystkich użytkowników” na stronie 7**
Aby skonfigurować połączenia z serwerem wszystkich użytkowników komputerów PC, należy użyć komendy CWBCFG.
- **“Profile użytkowników dla komputerów PC z wieloma użytkownikami” na stronie 8**
Systemy operacyjne Windows umożliwiają używanie profili użytkownika typu roaming i mandatory do zarządzania komputerami PC, na których pracuje więcej niż jeden użytkownik.

Microsoft Windows Terminal Server

Microsoft^(R) Windows^(R) Terminal Server jest funkcją umożliwiającą uruchomienie wielu jednoczesnych sesji klientów na jednym serwerze Windows. Umożliwia połączenie z wielu platform klienta, nie tylko Windows, lecz również stacji sieciowych, UNIX^(R), Linux, DOS, OS/2^(R) i innych. Instalując iSeries^(TM) Access for Windows^(R) na serwerze Windows można zapewnić dostęp do usług iSeries ze stacji roboczych, w których nie zainstalowano produktu iSeries Access for Windows.

Uwaga Gdy używa się Windows 2000 i nowszych wersji tego systemu operacyjnego z Terminal Services, należy ustawić opcję **Kiedy sprawdzać poziom usługi** w zakładce **Usługi** Właściwości iSeries Access for Windows na wartość **Nigdy**.

Informacje o instalowaniu, obsłudze, znanych problemach i rozwiązaniach dla iSeries Access for Windows z Microsoft Windows Terminal Server zawiera dokument APAR III1373. 

Więcej informacji o usługach Terminal Services na serwerze Windows NT^(R) można znaleźć na stronie WWW firmy Microsoft Windows NT Server 4.0 Terminal Server Edition. 

Korzystanie z iSeries Access for Windows w środowisku trójwarstwowym

Instalując iSeries^(TM) Access for Windows^(R) w środowisku pośredniczącym środowiska trójwarstwowego, udostępnia się usługi iSeries całej gamie klienckich stacji roboczych. Ponadto środowiska trójwarstwowe mają kilka innych zalet:

- **Ulepszona integracja pomiędzy różnymi aplikacjami klienta i serwera:** Wiele aplikacji klienckich działających na różnych klientach może komunikować się z wieloma aplikacjami serwera Windows jednocześnie. Każda z aplikacji serwera Windows może także komunikować się z wieloma bazami danych.
- **Ulepszone zarządzanie transakcjami za pomocą usługi Microsoft^(R) Transaction Server (MTS):** Środowiska trójwarstwowe umożliwiają przeprowadzanie bardziej złożonych transakcji, także transakcji powiązanych zależnościami. (Aby transakcje zostały zakończone, wszystkie muszą zakończyć się pomyślnie).
- **Importowanie danych z serwera iSeries do stron WWW za pomocą narzędzia Microsoft Internet Information Server (IIS):** IIS może wykorzystywać Active Server Pages w celu dynamicznego aktualizowania stron WWW danych z DB2 Universal Database^(TM) for iSeries.

Wszystkie środowiska trójwarstwowe dzielą komponenty i aplikacje na trzy warstwy. Mogą one znajdować się na oddzielnych komputerach PC i terminalach i komunikować się przez sieć. Będą one miały następujące cechy:

Warstwa klienta

Warstwa ta zawiera interfejs i aplikacje umożliwiające użytkownikom końcowym pracę z danymi. Na przykład, może to być przeglądarka działająca na stacji sieciowej lub aplikacja użytkownika używająca zdalnego komponentu. Warstwa ta nie używa klienta iSeries Access for Windows.

Warstwa pośrednicząca

Warstwa ta zawiera oprogramowanie firmowe lub aplikacje. W środowiskach wykorzystujących iSeries Access for Windows warstwa ta powinna składać się z serwera Windows z działającym skrypcem Microsoft Active Server Pages lub komponentem zdalnym. Dodatkowo, warstwa ta używa serwera Microsoft Internet Information Server (IIS) i Microsoft Transaction Server (MTS) do zarządzania transakcjami z warstwą klienta. iSeries Access for Windows używa sterownika ODBC lub dostawcy IBMDASQL OLE DB do obsługi MTS klientów i obsługuje komunikację z warstwą bazy danych. Zalecane jest używanie .NET, OLE DB, ActiveX Data Objects (ADO) oraz Remote Data Service do dostępu danych z komponentu w warstwie pośredniczącej.

Informacje o warstwie pośredniczącej znajdują się w następujących dokumentach:

- “Używanie serwera Microsoft Transaction Server (MTS)”
- “Dostęp do usług Access iSeries z warstwy pośredniczącej” na stronie 6

Warstwa bazy danych

Warstwa ta składa się zazwyczaj z bazy danych DB2 Universal Database for iSeries. Aplikacje mają dostęp do tej i innych usług iSeries poprzez programy serwerów hosta lub tworzone przez użytkownika programy dla iSeries.

Używanie serwera Microsoft Transaction Server (MTS)

Klient iSeries^(TM) Access for Windows^(R) dla serwera w wersji V5R1 i nowszych obsługuje serwer MTS w wersji 2.x i nowszych, ze sterownikiem ODBC iSeries Access oraz dostawcą OLE DB IBMDASQL.

MTS

MTS jest środowiskiem modelowania i uruchamiania oprogramowania, opartym na komponentach firmy Microsoft^(R), służącym do tworzenia, uruchamiania i zarządzania aplikacjami serwera internetowego. W wielu środowiskach trójwarstwowych Active Server Pages (ASP) wywołuje komponenty MTS, aby uzyskać dostęp do bazy danych, aplikacji mainframe i kolejek komunikatów. We współpracy z aplikacją iSeries Access for Windows działającą na serwerze pośredniczącym w środowisku trójwarstwowym, komponenty MTS zarządzają transakcjami pomiędzy aplikacjami klientów, komponentami iSeries Access for Windows i bazami danych uczestniczącymi w transakcjach.

MTS używa narzędzia Microsoft Distributed Transaction Coordinator (MSDTC) do zarządzania transakcjami, które obejmują wiele systemów Database Management Systems (DBMS), oraz do zapewnienia dwufazowej kontroli transakcji, których implementacje zależą od potwierdzenia powodzenia z obu systemów.

Uwagi do implementacji

- Jeśli MSDTC nie może załadować sterownika ODBC iSeries Access, funkcja SQLSetConnectAttr (SQL_ATTR_ENLIST_IN_DTC) zakończy się niepowodzeniem z kodem przyczyny 2 (niepowodzenie XaRmCreate). Jeśli zainstalowano emulator PC5250 iSeries Access for Windows, ścieżka środowiska systemowego MSDTC jest konfigurowana automatycznie. Aby uniknąć tej sytuacji, ścieżka środowiska systemowego w komputerze PC, w którym działa MSDTC, musi zawierać ścieżkę do katalogu Shared w katalogu, w którym zainstalowano aplikację iSeries Access for Windows. Na przykład: C:\Program Files\IBM\Client Access\Shared.
- Jeśli używana jest warstwa SSL lub dowolna inna wartość okna **Moje połączenia** → **Właściwości** w programie iSeries Navigator, nazwa połączenia iSeries w programie iSeries Navigator musi odpowiadać nazwie połączenia podanej w kliencie PC zarządzanym przez MTS. W celu nawiązywania połączeń z bazą danych DB2^(R) UDB for iSeries(TM)^(TM), MSDTC używa tych samych nazw połączeń, co komputery PC z klientami ODBC iSeries for Windows zarządzane przez MTS. Aby zmienić właściwości połączeń MSDTC, należy zmienić rejestr konta systemowego.

W tym celu można użyć komendy Incoming Remote Command (IRC) w połączeniu z programem narzędziowym CWBENV:

1. Uruchom w kliencie PC program CWBENV, aby pobrać informacje o konfiguracji środowiska.
2. Skopiuj plik wynikowy do komputera PC z MSDTC.
3. Uruchom usługę Komenda zdalna iSeries Access for Windows i upewnij się, że jest ona skonfigurowana do pracy w kontekście systemu lokalnego.
4. Aby zaimportować środowisko, za pomocą komendy RUNRMTCMD z sesji PC5250 wyślij do komputera PC z MSDTC komendę CWBENV.

Więcej informacji o tych funkcjach zawiera Podręcznik użytkownika (strona 3) w grupie programów iSeries Access for Windows.

Więcej informacji o systemie MTS zawiera serwis WWW firmy Microsoft .

Dostęp do usług Access iSeries z warstwy pośredniczącej

Istnieje kilka sposobów udostępniania komponentów warstwy pośredniczącej przy dostępie do serwera iSeries^(TM).

Uwaga: Komponenty warstwy pośredniczącej nie mogą mieć interfejsu użytkownika; dlatego aplikacje warstwy trzeciej mogą się zawiesić, jeśli iSeries Access wyświetli okno wpisywania się. Aby temu zapobiec, programiści powinni korzystać z nowego obiektu systemowego, aby określić wymagane informacje (identyfikator użytkownika i hasło), potrzebne przy łączeniu się z serwerem iSeries. Wartość trybu wiersza komend dla takiego obiektu musi być ustawiona na **prompt never**.

iSeries^(TM) Access for Windows^(R) .NET Data Provider

» Najnowsza wersja V5R3 bazy danych programu iSeries Access for Windows kontynuuje pokaz możliwości serwera iSeries w sposób pozwalający na jak najlepsze wykorzystanie pojawiających się technologii przemysłowych. «
IBM(R) DB2(R) UDB for iSeries .NET Provider oferuje znakomity dostęp do bazy danych iSeries programistom piszącym aplikacje przy użyciu struktury Microsoft^(R) .NET Data Access. W niniejszej dokumentacji termin **Managed Provider** jest używany wymiennie z **IBM DB2^(R) UDB for iSeries(TM)^(TM) .NET Provider** i **IBM.Data.DB2.iSeries data provider**. Niezależnie od nazwy, można wykorzystać pełny zestaw typów danych .NET i funkcjonalności instrukcji SQL, aby ułatwić pracę aplikacji z danymi bezpiecznie przechowywanymi w bazie danych serwera iSeries.

Więcej informacji zawiera sekcja Programowanie .NET.

Dostawca OLE DB w iSeries Access for Windows

Większość aplikacji i komponentów korzysta z dostawcy OLE DB iSeries Access for Windows, używając Obiektów Danych ActiveX (ActiveX Data Objects - ADO). Implementacja tej techniki daje cztery podstawowe korzyści:

- aby uzyskać dostęp do programów iSeries, komend, zapytań SQL, procedur osadzonych oraz zbiorów fizycznych i logicznych, programiści muszą wprowadzić jedynie niewielkie zmiany w interfejsie i technice programowania,
- wspomaga automatyczne konwersje danych pomiędzy typami danych iSeries i komputerów PC,
- pozwala uniknąć nakładu pracy związanego z SQL, poprzez udostępnianie obsługi zbiorów na poziomie rekordów,
- jest to technika stosunkowo łatwa do implementacji i programowania aplikacji; jako taka jest generalnie najprostszą technologią podczas programowania aplikacji warstwy trzeciej.

Więcej informacji na ten temat zawiera artykuł Programowanie OLE DB.

Sterownik ODBC w iSeries Access for Windows

Ponadto dostęp do sterownika ODBC iSeries Access można uzyskać poprzez ADO lub RDS (Remote Data Services), korzystając z Microsoft OLE DB dla ODBC (MSDASQL).

Więcej informacji dotyczących korzystania z ODBC poprzez obiekty ADO zawiera sekcja opisująca wybór interfejsu umożliwiającego dostęp do sterownika ODBC.

Inne informacje dotyczące sterownika ODBC w iSeries Access zawiera sekcja Programowanie ODBC.

Uwaga: Dostawca OLE DB w iSeries Access oraz kilka funkcji w sterowniku ODBC iSeries Access wymagają MDAC w wersji 2.5.

Obiekty automatyzacji ActiveX

Klient iSeries Access udostępnia bibliotekę nowych, rozszerzonych obiektów automatyzacji ActiveX, które programiści mogą wykorzystywać w projektowaniu warstwy pośredniczącej. Te obiekty dają dostęp do:

- kolejek danych iSeries,
- komend zdalnych i rozproszonych wywołań programów,
- obiektów administrowanych,
- obiektów systemowych iSeries,
- przesyłania danych do tabel baz danych iSeries.

W niektórych przypadkach obiekty ActiveX zapewniają większą uniwersalność i funkcjonalność niż obiekty ADO, ale wymagają stosowania bardziej złożonych metod programowania.

Uwaga: Klient iSeries Access for Windows zawiera bibliotekę automatyzacji z klienta Windows 95/NT (produkt XD1). Te obiekty automatyzacji, także baza danych, nie obsługują środowiska trójwarstwowego.

Funkcje API Express C/C++

Funkcje API iSeries Access for Windows umożliwiają szybki dostęp do serwerów OS/400(R) niskiego poziomu. Jednak korzystanie z nich wymaga od programistów C/C++ dużego doświadczenia. Dotyczy to zwłaszcza używania funkcji API, języka C, typów danych oraz ochrony wątków podczas tworzenia oprogramowania.

Dodawanie konfiguracji TCP/IP dla wszystkich użytkowników

Aby skonfigurować połączenia z serwerem iSeries^(TM) dla wszystkich użytkowników zdefiniowanych w komputerze PC, użyj komendy CWBCFG z wiersza komend lub z menu **Start** → **Uruchom**. Użycie tej komendy dodaje informacje o konfiguracji do domyślnego profilu użytkownika systemu Windows^(R), który jest profilem używanym podczas tworzenia dodatkowych profili użytkownika.

Aby dodać lub zmienić lokalizację, której używa emulator PC5250 podczas otwierania lub tworzenia plików, można również użyć komendy CWBCFG. Komenda CWBCFG może zmienić ustawienia lokalizacji dla wszystkich użytkowników komputerów PC.

Informacje online dotyczące komendy CWBCFG można znaleźć w publikacji iSeries^(TM) Access for Windows Podręcznik użytkownika (strona 3).

Określenie położenia plików PC5250 dla wszystkich użytkowników




Domyślna lokalizacja, w której emulator PC5250 wyszukuje i składa wszystkie pliki dla zdefiniowanych użytkowników, jest współużytkowana przez wszystkich użytkowników PC, chociaż niektórzy z nich mogą nie mieć odpowiednich uprawnień. Domyślną lokalizacją jest:

(folder instalacyjny iSeries^(TM) Access for Windows^(R))\emulator\private

Domyślna lokalizacja może być zmieniona przez każdego autoryzowanego użytkownika na pulpicie PC5250 w oknie Właściwości iSeries^(TM) Access for Windows. Aby ją zmienić dla wszystkich użytkowników jednocześnie, w wierszu komend administrator może wpisać komendę CWBCFG podając opcję /pc5250path.

Uwagi:

- Dowolne konto użytkownika utworzone po uruchomieniu komendy CWBCFG używa położenia domyślnego ustawionego przez komendę CWBCFG.
- Komenda CWBCFG może być używana wyłącznie przez administratorów.
- Komenda CWBCFG nie powoduje zmiany położenia zbiorów. W razie potrzeby zbiory muszą być przemieszczane ręcznie. 

Więcej informacji online na temat komendy CWBCFG zawiera Podręcznik użytkownika (strona 3) iSeries Access for Windows.

Profile użytkowników dla komputerów PC z wieloma użytkownikami

Za pomocą programu iSeries^(TM) Access for Windows^(R) można administrować wieloma użytkownikami. Ten typ administracji jest dostępny jako funkcja systemu operacyjnego Windows za pomocą profili: roving, roaming i mandatory.

Uwaga: Dokumentacja dotycząca sposobów implementacji metod administrowania wieloma użytkownikami w sieci znajduje się w pakiecie Microsoft^(R) Resource Kit for Windows dla używanego systemu operacyjnego Windows. Dokumenty Resource kit udostępniane są przez firmę Microsoft i dołączane do produktu Microsoft Developers Kit.

Profil roaming

Profile użytkownika roaming są profilami Windows, które mogą być przenoszone pomiędzy komputerami PC. Zmiana konfiguracji związana jest z użytkownikiem. Profile te są zwykle umiejscowione na serwerze Windows. Każdy użytkownik z profilem roaming ma na serwerze Windows katalog określony przez ścieżkę profilu użytkownika w ustawieniach tego profilu. Katalog ten zawiera zarówno informacje o rejestrach, jak i informacje o menu Start i pulpicie każdego użytkownika. Profile roaming mogą być przenoszone tylko między komputerami PC z systemami operacyjnymi z rodziny Windows NT^(R).

Profil mandatory

Profile mandatory są profilami użytkowników, które są konfigurowane przez administratora systemu, tak aby umożliwić użytkownikom korzystanie z komputerów PC. Dokumentacja dotycząca sposobów implementacji metod

administrowania wieloma użytkownikami w sieci znajduje się w pakiecie Microsoft Resource Kit dla używanego systemu operacyjnego Windows. Profile mandatory mogą znajdować się na jednym komputerze PC lub być przenoszone pomiędzy kilkoma komputerami PC.

Administrowanie ODBC

Open Database Connectivity (ODBC) jest standardem firmy Microsoft^(R) zapewniającym dostęp do baz danych. Zawiera on dobrze zdefiniowany zestaw interfejsów API wykorzystujących język SQL przy dostępie do baz danych.

“Ogólne informacje o sterowniku ODBC iSeries Access”

Temat ten zawiera ogólny opis ODBC i instrukcje używania go z iSeries^(TM) Access for Windows^(R).

“Konfigurowanie systemu iSeries Access do obsługi sterownika ODBC” na stronie 10

Temat ten prezentuje procedury konfigurowania środowiska do obsługi sterownika ODBC. Aby uzyskać pomoc przy konfigurowaniu sterownika ODBC, należy uruchomić program administrowania ODBC w grupie iSeries Access for Windows i przejrzeć pomoc online.

“Ochrona ODBC w iSeries Access for Windows” na stronie 12

Temat ten omawia zagadnienia ochrony przy pracy z ODBC i zawiera odniesienia do bardziej szczegółowych instrukcji ochrony.

Linux

Temat ten omawia instalowanie i korzystanie ze sterownika iSeries ODBC dla systemu Linux w celu uzyskania dostępu do bazy danych w systemie iSeries.

Uwaga: Sterownik iSeries ODBC dla systemu Linux nie jest częścią aplikacji iSeries Access for Windows. Jest to oddzielny produkt przeznaczony wyłącznie dla systemu operacyjnego Linux.

“Rozwiązywanie problemów z ODBC” na stronie 14

Temat ten może być pomocny przy rozwiązywaniu najczęstszych problemów z iSeries Access for Windows i ODBC. Podaje także kilka narzędzi służących do usuwania wąskich gardeł wydajności. Dokumentację tę należy przejrzeć przed skontaktowaniem się z obsługą techniczną.

Szczegółowe informacje o integrowaniu obsługi ODBC z aplikacjami zawiera dokument iSeries Access for Windows Programowanie ODBC, w którym są opisane następujące zagadnienia:

- Lista interfejsów API ODBC
- Implementacja interfejsów API ODBC
- Przykłady programów
- Wydajność ODBC

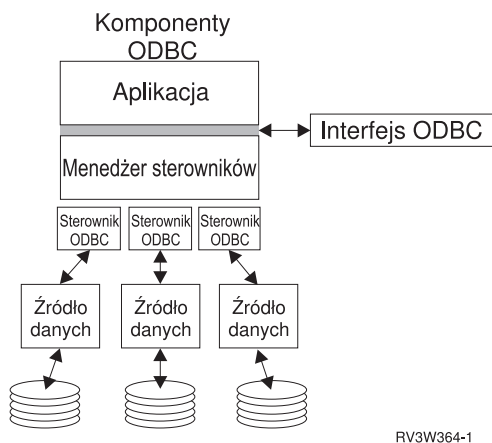
Ogólne informacje o sterowniku ODBC iSeries Access

Sterownik ODBC iSeries^(TM) Access jest zestawem interfejsów API umożliwiającym dostęp do informacji znajdujących się w bazach danych za pomocą języka SQL. Korzystanie ze sterownika ODBC iSeries Access umożliwia aplikacjom dostęp do różnych baz danych w serwerze iSeries za pomocą tego samego kodu źródłowego i obsługę danych w formatach najdogodniejszych dla tych aplikacji. ODBC zapewnia twórcom aplikacji relatywnie prosty model tworzenia przenośnych aplikacji lub komponentów, które muszą pracować z wieloma serwerami DBMS.

Architektura ODBC dotyczy aplikacji, menedżera sterowników i źródła danych. Z aplikacją iSeries Access dostarczany jest sterownik ODBC w wersji 32- i 64-bitowej. Sterownik 64-bitowy jest instalowany razem ze sterownikiem 32-bitowym automatycznie w przypadku 64-bitowej wersji systemu Windows^(R). Aplikacje ODBC w 64-bitowej wersji

Windows będą automatycznie korzystać z odpowiedniego sterownika ODBC w zależności od tego, czy zostały skompilowane w wersji 32- czy 64-bitowej. Sterownika 64-bitowego można używać tylko razem z aplikacjami 64-bitowymi.

Aby aplikacja mogła korzystać z ODBC, należy skonfigurować źródło danych. W tym celu można użyć Administratora ODBC. Istnieją dwie wersje tej aplikacji: 32-bitowa i 64-bitowa, które są dostępne z folderu iSeries^(TM) Access for Windows. Aplikacja Administrator ODBC umożliwia skonfigurowanie trzech różnych typów źródeł danych: użytkownika, systemowe i z plików. Więcej informacji o konfigurowaniu źródeł danych zawiera sekcja Obsługa 64-bitowego interfejsu ODBC w Podręczniku użytkownika iSeries Access for Windows.



Aplikacja. Realizuje przetwarzanie i wywołuje funkcje ODBC w celu uruchamiania instrukcji SQL.

Menedżer sterowników. Przetwarza wywołania funkcji ODBC i przesyła żądania do sterownika.

Sterownik. Przetwarza wywołania funkcji ODBC, wysyła żądania SQL do określonych źródeł danych i zwraca wyniki do aplikacji.

Źródło danych. Aby używać źródła danych, należy utworzyć nazwę źródła danych (DSN). DSN zawiera informacje o dostępie do DBMS. Można podać jedną z następujących nazw DSN:

- **DSN użytkownika:** Te źródła danych są źródłami lokalnymi dla komputera i są dostępne tylko dla użytkownika, który je utworzył. Informacje te przechowywane są w rejestrze.
- **DSN systemu:** Te źródła danych są źródłami lokalnymi dla komputera, nie są przeznaczone dla użytkownika. System lub użytkownik z odpowiednimi uprawnieniami może używać źródła danych skonfigurowanego w DNS systemu. Informacje te przechowywane są w rejestrze.


Uwaga: W komputerach PC z procesorem 64-bitowym część rejestru opisująca system jest podzielona na elementy 32- i 64-bitowe. Systemowe nazwy DSN skonfigurowane przy użyciu 32-bitowego Administratora ODBC są dostępne tylko dla aplikacji 32-bitowych. Podobnie, systemowe nazwy DSN, skonfigurowane przy użyciu 64-bitowego Administratora ODBC, są dostępne tylko dla aplikacji 64-bitowych.

- **DSN pliku:** Są to źródła danych oparte na plikach, które mogą być współużytkowane przez wszystkich użytkowników z zainstalowanymi takimi samymi sterownikami, umożliwiającymi im dostęp do bazy danych. Te źródła danych nie muszą być dedykowane dla użytkownika ani lokalne dla komputera.

Więcej informacji o ODBC można znaleźć w serwisie WWW firmy Microsoft^(R).

Konfigurowanie systemu iSeries Access do obsługi sterownika ODBC

Sterownik ODBC iSeries^(TM) Access ODBC jest zgodny z ODBC wersja 3.5. Wymaga on komponentów Microsoft^(R) Data Access Components (MDAC) w wersji 1.5 lub nowszej. Dla aplikacji korzystających z obiektów ADO (Microsoft ActiveX Data Objects) powinny być zainstalowane komponenty MDAC w wersji 2.1 lub nowszej. Środowiska wykonawcze dla MDAC w wersji 2.1 i nowszych udostępniają dodatkową funkcję dla aplikacji, które używają

obiektów ADO (Microsoft OLE DB provider for ODBC) i ODBC iSeries^(TM) Access for Windows^(R) w celu uzyskania dostępu do danych serwera iSeries. Jeśli aplikacja korzysta z obsługi pul połączeń lub serwera Microsoft Transaction Server (MTS), zaleca się zainstalowanie najnowszej wersji MDAC. Oprogramowanie MDAC można pobrać ze strony WWW firmy Microsoft: www.microsoft.com/data .

Informacje dotyczące konfigurowania sterownika ODBC zawiera sekcja Źródło danych ODBC. Wykonaj konfigurowanie postępując zgodnie z krokami opisanymi w sekcji Dodawanie systemu lokalnego do katalogu RDB.

Używanie niezależnych pul ASP z ODBC jest opcjonalne. Więcej informacji na ten temat zawiera sekcja Niezależne pule ASP (strona 12).

Aby uzyskać pomoc podczas ustawiania opcji dla określonego źródła danych, należy uruchomić program Administrowanie ODBC z grupy iSeries Access for Windows, wybrać źródło danych, które ma zostać skonfigurowane, i przeczytać pomoc online.

Dodawanie systemu lokalnego do katalogu RDB

Aby można było używać ODBC, dostawcy OLE DB lub dostawcy OLE .NET, nazwa systemu lokalnego musi znajdować się w katalogu RDB.

Aby dodać system lokalny do katalogu RDB:

1. W wierszu komend wpisz komendę CL Dodanie pozycji katalogu RDB (Add Relational Database Directory Entry - ADDRDBDIRE).
2. Gdy wyświetlony zostanie ekran ADDRDBDIRE, wpisz nazwę systemu jako parametr relacyjnej bazy danych.
3. Jako parametr miejsca zdalnego wpisz *LOCAL.

Aby podać nazwę bazy danych (RDB), gdy zainstalowana jest wersja systemu V5R2 lub nowsza, a aplikacja używa danych w niezależnych ASP, konieczne może okazać się wykonanie dodatkowych czynności. Nazwa RDB odpowiada przestrzeni nazw, która składa się z systemowej ASP i dowolnych ASP użytkownika lub dowiązanej grupy ASP, powiązanych z systemową ASP. Więcej informacji na temat niezależnych ASP zawiera sekcja Zarządzanie dyskami.

Uwaga: ODBC umożliwia użycie pełnych nazw w formacie [nazwa katalogu].[nazwa schematu].identyfikator (gdzie identyfikator może być nazwą tabeli, widoku, procedury itd.). W implementacji języka SQL w produkcie DB2^(R) UDB for iSeriesTM odpowiada to formatowi [baza RDB].[nazwa kolekcji].identyfikator.

Podanie źródła danych ODBC

Należy wskazać źródło danych dla aplikacji, aby mogły uzyskać dostęp i pracować z danymi.

Aby podać źródło danych:

1. Uruchom program Administrowanie ODBC z grupy iSeries^(TM) Access for Windows^(R).
2. Wybierz odpowiednią zakładkę dla żadanego typu źródła danych. Więcej informacji na ten temat zawiera sekcja "Ogólne informacje o sterowniku ODBC iSeries Access" na stronie 9.
3. Wybierz z listy istniejące źródło danych lub opcję **Add (Dodaj)**, aby utworzyć nowe źródło. Jeśli używasz istniejącego źródła danych, kliknij opcję **Configure (Konfiguruj)** i przejdź do kroku 5.
4. Jako źródło danych wybierz pozycję iSeries Access ODBC Driver (Sterownik ODBC iSeries Access) i kliknij przycisk **Finish (Zakończ)**.
Uwaga: Na liście sterowników pojawi się pozycja Client Access ODBC Driver (32-bit). Dzięki temu źródła danych utworzone w poprzednich wersjach produktu Client Access będą nadal działać. Obie nazwy wskazują na ten sam sterownik ODBC. Można używać obu sterowników, jednakże w przyszłych wersjach sterownik 'Client Access ODBC Driver (32-bit)' zostanie usunięty.
5. Podaj wybrane opcje w oknie konfiguracji ODBC iSeries Access for Windows. Opis elementów sterujących znajduje się w pomocy elektronicznej dla źródła danych, którą można wyświetlić naciskając klawisz F1 lub klikając przycisk Help (Pomoc).

Uwaga:Nazwa źródła danych może zawierać do 32 znaków, zaczynać się od litery i nie może zawierać następujących znaków:

Znaki, które nie mogą występować w nazwie źródła danych	
lewy nawias kwadratowy ([)	znak zapytania (?)
prawy nawias kwadratowy (])	gwiazdka (*)
lewa klamra ({)	znak równości (=)
prawa klamra (})	wykrzyknik (!)
lewy nawias (()	małpa (@)
prawy nawias ())	średnik (;)

Niezależne ASP

Aby używać **niezależnych ASP** poprzez ODBC, skonfiguruj ODBC DSN i wykonaj następujące czynności:

1. Wybierz zakładkę **Serwer**.
2. Podaj **nazwę bazy RDB** znajdującej się w **niezależnej puli ASP**, z którą chcesz się połączyć.
3. Kliknij "Nadpisanie domyślnej bazy danych z użyciem:".
4. Podaj nazwę bazy RDB znajdującej się w niezależnej puli ASP, z którą chcesz się połączyć.
5. Jeśli nie zostanie podana nazwa bazy RDB, zostanie użyta nazwa domyślna, określona na podstawie opisu zadania dla profilu użytkownika, który nawiązuje połączenie ODBC. Domyślnie sterownik korzysta z ustawień profilu użytkownika nawiązującego połączenie ODBC.

Więcej informacji na temat **niezależnych ASP** zawiera sekcja Zarządzanie dyskami.

Ochrona ODBC w iSeries Access for Windows

Poniższe informacje nie stanowią wyczerpującego omówienia strategii ochrony serwerów iSeries^(TM) ani produktu iSeries Access for Windows^(R). Zawierają tylko przegląd strategii ochrony dotyczących użytkowników programu iSeries Access for Windows i interfejsu ODBC. Bardziej szczegółowe informacje zawiera publikacja IBM^(R) Ochrona



- Ryzykowne strategie ochrony ODBC
- Strategie ochrony programu ODBC
- Pokrewne informacje o ochronie ODBC

Ryzykowne strategie ochrony ODBC

Niektórzy administratorzy systemów usiłują zapewnić ochronę dostępu do danych, a nie ochronę samych danych. Jest to niezwykle ryzykowna praktyka, ponieważ wymaga zrozumienia WSZYSTKICH metod, jakie umożliwiają użytkownikom dostęp do danych. Niektóre popularne techniki ochrony ODBC, których należy unikać, to:

Ochrona wiersza komend

Ochrona może być przydatna w przypadku aplikacji opartych na interfejsie znakowym lub na emulacji terminalu 5250. Zakłada ona, że jeśli zablokuje się użytkownikom możliwość wpisywania komend w sesji emulacji 5250, będą mogli uzyskać dostęp do danych tylko poprzez programy i menu dostarczone przez administratora systemu. Dlatego nigdy nie jest ona w pełni bezpieczna. Aby zwiększyć ochronę, należy użyć strategii iSeries^(TM) Access i administrowania aplikacjami. Dalsze zwiększenie ochrony umożliwiają uprawnienia na poziomie obiektu.

Potencjalnie, strategie iSeries^(TM) Access for Windows^(R) mogą ograniczyć dostęp ODBC do danego źródła danych, które zostało oznaczone jako tylko do odczytu. Administracja aplikacji w programie iSeries Navigator może zablokować dostęp przez ODBC.

Dodatkowe informacje zawiera publikacja IBM^(R) Ochrona .


Programy obsługi wyjścia użytkowników

Program obsługi wyjścia użytkownika umożliwia administratorowi systemu zabezpieczenie programu serwera dostarczonego przez IBM. Sterownik ODBC iSeries Access używa serwera baz danych: punkty wyjścia QIBM_QZDA_INIT; QIBM_QZDA_NDBx; i QIBM_QZDA_SQLx. Niektóre sterowniki ODBC i inne metody dostępu do danych iSeries Access for Windows (takie jak OLE DB) mogą używać innych serwerów.

Kroniki

Kronikowanie używane jest często w aplikacjach klient/serwer w celu zapewnienia kontroli transakcji. Kroniki zawierają szczegółowe informacje o każdej aktualizacji kronikowanego zbioru. Informacje w kronikach mogą być formatowane i przeszukiwane w celu otrzymania konkretnych informacji, takich jak:

- profile użytkowników, którzy aktualizowali pliki,
- rekordy, które zostały zaktualizowane,
- typ aktualizacji.

Kronikowanie umożliwia także definiowanie własnych zapisów w kronice. Opcja ta użyta wraz z programem obsługi wyjścia użytkownika lub wyzwalaczem zapewnia stosunkowo mało rozbudowaną obsługę kontroli zdefiniowanej przez użytkownika. Dodatkowe informacje zawiera artykuł Składowanie i odtwarzanie .

Ograniczenia nazwy źródła danych (DSN)

Sterownik ODBC iSeries Access obsługuje konfigurację DSN umożliwiającą dostęp do bazy danych w trybie tylko do odczytu. Sterownik ODBC iSeries Access obsługuje ustawienie tylko do odczytu i odczyt-wywołanie dla źródła danych. Mimo że ustawienia te nie są bezpieczne, mogą pomóc w zapobieganiu przypadkowym operacjom usunięcia i aktualizacji.

Strategie ochrony programu ODBC

Rozważmy następujące strategie ochrony programu ODBC:

Ograniczanie dostępu programów do bazy danych

Administratorzy systemów często muszą ograniczać dostęp do określonych plików, programów lub ich zestawów. Programista używający interfejsu znakowego ustawi ograniczenia za pomocą uprawnień adoptowanych przez programy. Podobna metoda możliwa jest w ODBC.

Procedury zapisane w bazie umożliwiają programistom ODBC implementowanie uprawnień adoptowanych przez programy. Programista może ograniczyć użytkownikom możliwość manipulowania zbiorami bazy danych przy pomocy aplikacji, takich jak Microsoft^(R) Access lub Lotus^(R) 1-2-3^(R). Może on również ograniczyć aktualizacje bazy danych wyłącznie dla aplikacji programistycznych. Aby to wprowadzić, dostęp użytkowników do bazy danych musi być ograniczony ochroną na poziomie obiektu lub przy pomocy programów obsługi wyjścia użytkownika. Aplikacja musi być napisana tak, aby wysyłała żądania danych do procedury zapisanej w bazie danych i wymuszała na tej procedurze aktualizacje bazy danych.

Ograniczenie użycia jednostki centralnej CPU dla użytkownika

ODBC znacznie ułatwia dostęp do danych iSeries^(TM). Jedną z negatywnych stron tej sytuacji jest to, że użytkownicy mogą czasem tworzyć zapytania poważnie obciążające procesor nie zdając sobie z tego sprawy. ODBC działa z

priorytetem zadania interaktywnego, co może poważnie wpłynąć na wydajność systemu. System iSeries obsługuje **menedżera zapytań**. ODBC może go uruchomić (na przykład za pomocą aplikacji PC) przez wywołanie procedury zapisanej w bazie. Także interfejsy API ODBC mogą uruchomić program zarządzający za pomocą parametru limitu czasu zapytania. Program obsługi wyjścia użytkownika może również wymusić uruchomienie zarządcy zapytań w zadaniu ODBC. Limit czasu podany jest w parametrze QRYTIMLMT komendy CL CHGQRYA. Pliku opcji zapytania (QAQQINI) można także użyć do podania tej wartości.


Książka *SQL Reference* zawiera dodatkowe informacje na ten temat. Można ją przejrzeć w wersji elektronicznej (HTML) lub wydrukować jako dokument PDF z DB2 Universal Database^(TM) for iSeries SQL Reference.

Więcej informacji na ten temat zawiera sekcja “Administrowanie serwerem hosta” na stronie 22.

Protokoły kontroli (monitorowanie ochrony)





Do monitorowania ochrony można użyć kilku protokołów. Protokół historii QHST zawiera komunikaty odnoszące się do zmian w ochronie wprowadzonych w systemie. Szczegółowe monitorowanie funkcji związanych z ochroną umożliwia udostępnienie kolejki QAUDJRN. Wartość *SECURITY protokołuje następujące funkcje:

- Zmiana uprawnień do obiektu
- Operacje utworzenia, zmiany, usunięcia, wyświetlenia i odtworzenia dla profili użytkowników
- Zmiany w prawach własności do obiektów
- Zmiany w programach (CHGPGM), które dziedziczą profile właściciela
- Zmiany w wartościach systemowych i atrybutach sieciowych
- Zmiany w routingu podsystemu
- Resetowanie hasła QSECOFR do wartości pierwotnej przez narzędzia DST
- Żądanie przywrócenia wartości domyślnej hasła szefa ochrony DST
- Zmiany w atrybucie kontroli dla obiektu

Dodatkowe informacje zawiera publikacja IBM^(R) Ochrona .

Informacje pokrewne o ochronie ODBC

Dogłębna analiza zagadnień ochrony i pomoc we wprowadzaniu powyższych strategii dostępne są poprzez usługę IBM^(R) Consultline (1-800-274-0015). Szczegółowe informacje dotyczące konkretnych zagadnień dostępne są w następujących dokumentach:

- “Administrowanie serwerem hosta” na stronie 22
- IBM Ochrona 
- Składowanie i odtwarzanie 
- DB2 Universal Database^(TM) for iSeries^(TM) SQL Reference
- Sekcja **Client Access ODBC and OLE DB Security Issues** w dokumentacji technicznej (Technical Reference) dostępna po wykonaniu następujących instrukcji: 
 - Przejdź do strony www.ibm.com/servers/eserver/iserier/support
 - Przejdź do **Szybkie wyszukiwanie!** —> **Wyszukiwanie bazy danych technicznych**
 - Jako kryterium wyszukiwania podaj tytuł (Client Access ODBC and OLE DB Security Issues). 

Rozwiązywanie problemów z ODBC

Poniższe tematy zawierają ogólne wytyczne dotyczące wyszukiwania i rozwiązywania błędów ODBC iSeries^(TM) Access for Windows^(R):

- “Diagnostyka ODBC i narzędzia wydajności” na stronie 15
- “Komunikaty o błędach ODBC iSeries Access” na stronie 16

- Rozwiązywanie problemów dotyczących połączenia z serwerem iSeries
- Typowe błędy ODBC
- Zbieranie informacji dla działu wsparcia IBM^(R)

Diagnostyka ODBC i narzędzia wydajności

Poniższe tabele opisują diagnostykę ODBC i narzędzia wydajności zarówno dla klienta, jak i serwera: **Narzędzia po stronie klienta**

Śledzenie ODBC (SQL.LOG)	Administrator ODBC firmy Microsoft ^(R) zapewnia własne narzędzie śledzenia wywołań funkcji API ODBC z aplikacji. Więcej informacji zawiera sekcja Zbieranie danych śledzenia ODBC (SQL.LOG).
Programy śledzenia ODBC	Dostępne są różne narzędzia do śledzenia ODBC, bardziej zaawansowane niż Śledzenie ODBC (SQL.LOG). Te komercyjne narzędzia zapewniają szczegółowy zapis i śledzenie punktów wejścia dla wywołań funkcji API ODBC. Dwoma takimi narzędziami są: Trace Tools (Dr. DeeBee) i SST Trace Plus (Systems Software Technology).
CWBPING	Aby użyć komendy CWBPING, w wierszu komend wpisz cwbping (nazwa systemu lub adres IP). Na przykład: cwbping testsys1 lub cwbping 127.127.127.1 CWBPING odpowiada wyświetlając listę serwerów i ich status. Aby uzyskać pomoc dotyczącą używania komendy CWBPING, uruchom ją bez parametrów. Więcej informacji na temat komendy CWBPING zawiera sekcja "Sprawdzenie statusu serwera" na stronie 18.
CWBCOTRC	Aby użyć komendy CWBCOTRC, w wierszu komend należy przejść do katalogu \Program Files\IBM\Client Access i wpisać CWBCOTRC ON . Po włączeniu śledzenia można uruchomić aplikację. Wpisanie komendy CWBCOTRC OFF zatrzymuje śledzenie. Narzędzie CWBCOTRC zbiera informacje o danych przesyłanych do i z serwera. Aby uzyskać pomoc dotyczącą używania komendy CWBCOTRC, należy uruchomić ją bez parametrów.
Śledzenie szczegółowe	Śledzenie szczegółowe umożliwia zebranie informacji zgromadzonych przez używane komponenty iSeries ^(TM) Access for Windows ^(R) . W ten sposób można zebrać informacje ODBC dotyczące punktów wyjścia sterownika, zadania prestartu, nazwy używanego pakietu oraz specjalnych warunków błędu. Więcej informacji zawiera sekcja Zbieranie szczegółowych danych śledzenia.

Narzędzia dla serwera

Śledzenie komunikacji	Narzędzie śledzenia śledzi i formatuje dane o dowolnym typie komunikacji (Token Ring i Ethernet), dla których istnieje opis linii. Narzędzie to służy do izolowania wielu problemów. Jest także użyteczną pomocą do diagnozowania wydajności. Pola datownika i eye-catcher służą dopomiarów czasu przetwarzania żądania.
-----------------------	---

Śledzenie zadań	<p>Ślad zadania pomaga w izolowaniu większości problemów z hostem i zagadnień związanych z wydajnością. Należy najpierw uruchomić zadanie usługi dla śledzonego zadania. Należy określić pełną nazwę zadania ODBC. W dowolnej sesji emulacji 5250 należy uruchomić zadanie usługi dla zadania QZDASOINIT używając komendy STRSRVJOB. Następnie należy wybrać jeden z dwóch typów śledzenia, w zależności od wymaganych informacji:</p> <p>Śledzenie zadania Służy do śledzenia wywołań wewnętrznych wysyłanych przez serwer. Należy uruchomić komendę TRCJOB *ON.</p> <p>Debugowanie Służy do przeglądania wydajności aplikacji i określania przyczyn danego problemu.</p> <p>Komenda STRDBG uruchamiana jest poprzez aktywne zadanie usługi. Komenda ta zapisuje decyzje podejmowane przez funkcję optymalizacji zapytań do protokołu zadania sesji debugowania. Na przykład, zapisuje ona szacunkowe czasy zapytań, użyte ścieżki dostępu oraz błędy kursorów.</p> <p>Prostym sposobem jej aktywowania jest skonfigurowanie ODBC DSN używanego przez Administrатора ODBC wybierając opcję Aktywowanie komendy debugowania (STRDBG) na karcie Diagnostyka. Alternatywnie, można uruchomić następującą komendę:</p> <p style="text-align: center;">STRDBG UPDPROD(*YES)</p> <p>Protokół zadania ODBC może zapisać wszystkie błędy występujące w serwerze iSeries. Gdy zadanie jest w trybie debugowania, protokół zadania będzie zawierał dane związane z wydajnością.</p>
Performance Tools	<p>Narzędzie do badania wydajności zawiera raporty i narzędzia służące do tworzenia wnikliwych analiz wydajności aplikacji. Udostępnia informacje o obciążeniu procesora i ramion dysków, stronicowaniu pamięci i wiele innych. Mimo że podstawowy system operacyjny ma możliwość zbierania danych o wydajności, wymagany będzie osobny program licencjonowany Performance Tools/400 do analizowania wyników.</p> <p>Można także używać narzędzi Database Monitor i Visual Explain. Więcej informacji zawiera pomoc online programu iSeries Navigator.</p>
Protokół zadania QZDASOINIT	<p>Aby uzyskać optymalną obsługę, należy wygenerować, znaleźć i pobrać protokół zadania QZDASOINIT. Może on zawierać komunikaty, które pomogą w określeniu i rozwiązaniu błędów zwracanych przez ODBC.</p> <p>Prostym sposobem dostępu do protokołu zadania jest skonfigurowanie ODBC DSN używanego przez Administrатора ODBC wybierając opcję Drukowanie protokołu zadania po rozłączeniu na karcie Diagnostyka. Aby znaleźć protokół zadania, należy otworzyć sesję emulacji PC5250 i uruchomić komendę WRKSPLF. Konieczne jest określenie profilu użytkownika iSeries używanego do połączenia ODBC jako parametru użytkownika dla komendy WRKSPLF.</p>
QAQQINI (Plik opcji zapytania - Query options file)	<p>Bibliotekę dla pliku opcji programu Query można ustawić konfigurując ODBC DSB używany przez Administrатора ODBC i wybierając zakładkę Diagnostyka. W odpowiednim polu pliku opcji program Query podaj nazwę biblioteki, która ma być używana.</p>

Komunikaty o błędach ODBC iSeries Access

Gdy wystąpi błąd, sterownik ODBC iSeries^(TM) Access zwraca kod SQLSTATE (kod błędu ODBC) i komunikat o błędzie. Sterownik pobiera informacje o błędach, które wykrył, i błędach zwróconych przez DBMS.

W przypadku błędów, które wystąpiły w źródle danych, sterownik ODBC iSeries Access przypisuje zwrócony rodzimy kod błędu do odpowiedniego SQLSTATE. Gdy sterownik ODBC iSeries Access i menedżer sterownika Microsoft^(R)

wykryją błąd, generują odpowiedni komunikat SQLSTATE. Sterownik ODBC iSeries Access Express zwraca komunikat o błędzie w oparciu o komunikat zwrócony przez DBMS.

W przypadku błędów, które występują w sterowniku ODBC iSeries Access lub menedżerze sterownika Microsoft, sterownik ODBC iSeries Access zwraca komunikat o błędzie w oparciu o tekst przypisany do SQLSTATE.

Format komunikatu o błędzie

Komunikaty o błędzie mają następujący format:

[dostawca] [komponent-ODBC] [źródło-danych]
komunikat-o-błędzie

Prefiksy w nawiasach kwadratowych ([]) identyfikują źródło błędu. Poniższa tabela zawiera wartości tych prefiksów zwrócone przez sterownik ODBC iSeries Access.

Gdy błąd się pojawia w źródle danych, prefiksy [dostawca] i [komponent-ODBC] identyfikują dostawcę i nazwę komponentu ODBC, które otrzymały błąd ze źródła danych.

Źródło błędu	Wartość
Menedżer sterowników	[Microsoft] [ODBC Driver Manager] [N/A]
Sterownik ODBC iSeries Access	[IBM ^(R)] [iSeries Access ODBC Driver] N/A
Komunikaty NLS	[IBM] [iSeries Access ODBC Driver] Kolumna #: Numer komunikatu o błędzie NLS Tekst komunikatu o błędzie NLS
Warstwa komunikacyjna	[IBM] [iSeries Access ODBC Driver] Brak łącza komunikacyjnego. Comm RC=xxxx - (tekst komunikatu) gdzie xxxx to numer błędu w postaci dziesiętnej, nie szesnastkowej. Tekst komunikatu opisujący rodzaj błędu wyświetlany jest wraz z numerem błędu. Uwaga: Więcej informacji dotyczących identyfikatorów komunikatów o błędach zawiera sekcja Kody powrotu iSeries Access lub dostępna online publikacja iSeries Access for Windows Podręcznik użytkownika (strona 3).
DB2 ^(R) UDB for iSeries	[IBM] [iSeries Access ODBC Driver] [DB2 UDB] Komunikat o błędzie serwera

Przeglądanie tekstów komunikatów o błędach DB2^(R) UDB for iSeries(TM)^(TM):

Dla błędów, które rozpoczynają się od:	Należy użyć komendy CL
SQL	DSPMSGD RANGE(SQLxxxx) MSGF(QSQLMSG)
IWS lub PWS	DSPMSGD RANGE(ZZZxxxx) MSGF(QIWS/QIWSMSG) gdzie ZZZ to IWS lub PWS

Pomoc dotyczącą innych komunikatów o błędach ODBC zawiera dokument "Typowe błędy ODBC" na stronie 20.

Komunikaty o błędach komunikacji lub NLS można przeglądać w temacie dotyczącym komunikatów o błędach i komunikatów śledzenia w publikacji online iSeries^(TM) Access for Windows^(R) Podręczniku użytkownika (strona 3).

Rozwiązywanie problemów dotyczących połączenia z serwerem iSeries

Każde połączenie ODBC komunikuje się z jednym programem serwera baz danych działającym w serwerze iSeries^(TM). Program ten jest określany jako **program serwera hosta**. Nazwą programu serwera baz danych używaną w protokole TCP/IP jest **QZDASOINIT**. Zwykle znajduje się ona w podsystemie QUSRWRK, jednak administrator systemu może określić inne położenie.

W normalnych warunkach program ten jest wywoływany jawnie, a użytkownik nie musi wykonywać żadnych innych czynności poza zweryfikowaniem, czy działają odpowiednie podsystemy i protokoły komunikacyjne. Szczegóły na temat administrowania zadaniami serwerów hosta zawiera sekcja "Administrowanie serwerem hosta" na stronie 22.

Najczęstszym objawem awarii połączenia jest komunikat o błędzie wysłany przez sterownik ODBC mówiący o awarii łącza komunikacyjnego.

Jeśli ODBC nie może połączyć się z serwerem iSeries, należy wykonać czynności opisane w następujących sekcjach:

- "Sprawdzenie statusu serwera"
- "Sprawdzenie, czy podsystemy są aktywne"
- "Sprawdzenie, czy działają zadania prestartu" na stronie 19
- "Dodatkowe zagadnienia związane z protokołem TCP/IP" na stronie 19

Sprawdzenie statusu serwera: Produkt iSeries^(TM) Access for Windows^(R) udostępnia specjalną komendę służącą do weryfikowania statusu serwerów:

```
CWBPING nazwa_systemu
```

gdzie `nazwa_systemu` to nazwa systemu.

Komenda powinna zwrócić informacje podobne do następujących:

```
To cancel the CWBPING request, press CTRL-C or CTRL=BREAK
I - Verifying connection to system MYSYSTEM...
I - Successfully connected to server application: Central Client
I - Successfully connected to server application: Network File
I - Successfully connected to server application: Network Print
I - Successfully connected to server application: Data Access
I - Successfully connected to server application: Data Queues
I - Successfully connected to server application: Remote Command
I - Successfully connected to server application: Security
I - Successfully connected to server application: DDM
I - Successfully connected to server application: Telnet
I - Successfully connected to server application: Management Central
I - Connection verified to system MYSYSTEM
```

Uwagi:

- Aby interfejs ODBC działał, baza danych i serwery ochrony muszą funkcjonować prawidłowo.
- Jeśli zostanie wyświetlony komunikat informujący, że połączenie jest skonfigurowane do obsługi SSL, mogą go używać tylko aplikacje 32-bitowe. W takiej sytuacji obsługa połączenia przez 64-bitowy sterownik ODBC iSeries Access lub 64-bitowego dostawcę OLE DB iSeries Access nie powiedzie się. Aby połączenie z serwerem iSeries przy użyciu aplikacji 64-bitowej powiodło się, należy najpierw skonfigurować je tak, aby nie korzystało z warstwy SSL.

Sprawdzenie, czy podsystemy są aktywne: Zadania ODBC połączone poprzez TCP/IP (QZDASOINIT) będą działały w podsystemie QUSRWRK. Należy sprawdzić, czy podsystem ten działa. Może wystąpić konieczność ręcznego uruchomienia podsystemu QSERVER. Aby to zrobić, należy wydać następującą komendę:

STRSBS QSERVER

Aby podsystem był uruchamiany automatycznie w trakcie IPL, należy umieścić w procedurze startowej IPL Start up (domyślnie QSYS/QSTRUP) komendę STRSBS QSERVER.

Oprócz podsystemu QSERVER, musi działać także podsystem QSYSWRK.

Sprawdzenie, czy działają zadania prestartu: IBM^(R) dostarcza podsystemy QSERVER/QUSRWRK skonfigurowane do używania zadań prestartu w celu zwiększenia wydajności uruchamiania zadań. Gdy w systemie skonfigurowane są zadania prestartu, zadanie MUSI być aktywne, aby można było nawiązać połączenie. Zadanie prestartu używane w połączeniu TCP/IP to:

- QZDASOINIT - program serwera

Aby sprawdzić, czy zadanie prestartu działa, wpisz:

```
WRKACTJOB SBS(QUSRWRK)
```

```
WRKACTJOB SBS('Podsystem-zdefiniowany-przez-uzytkownika)
```

Odpowiednie zadania prestartu powinny działać:

Zadanie	Użytk.	Typ	-----Status-----	
QZDASOINIT	QUSER	PJ	ACTIVE	(socket connection)
QZDASRVSD	QUSER	PJ	ACTIVE	(socket connection)

Zadania prestartu nie są wyświetlane za pomocą komendy WRKACTJOB, jeśli połączenie nie jest aktywne. Należy użyć klawisza F14 (Włączenie) z panelu WRKACTJOB

Dodatkowe zagadnienia związane z protokołem TCP/IP: Należy sprawdzić, czy protokół TCP/IP został uruchomiony następującą komendą:

```
NETSTAT *CNN
```

Uwaga: Aby za pomocą aplikacji iSeries^(TM) Navigator sprawdzić, czy protokół TCP/IP jest uruchomiony, należy uprzednio skonfigurować serwer do obsługi TCP/IP, a następnie:

1. W programie iSeries Navigator wybierz serwer —> Sieć.
2. Prawym przyciskiem myszy kliknij Konfiguracja TCP/IP i wybierz Narzędzia.
3. Wybierz Ping.
4. Podaj nazwę hosta lub adres TCP/IP i kliknij Wykonaj Ping.

Następnie, za pomocą komendy STRTCP, należy uruchomić wymagany protokół, jeśli nie jest aktywny.

Należy sprawdzić, czy wymagane demony działają, przeglądając informacje zwrócone przez komendę NETSTAT *CNN:

Zdalny adres	Zdalny port	Lokalny port	Czas bezczynn.	Stan
*	*	as-cent >	000:09:31	Listen
*	*	as-signon	000:09:41	Listen
*	*	as-svrmap	002:57:45	Listen
*	*	as-data >	002:57:45	Listen

Aby w razie potrzeby je uruchomić, należy użyć komendy STRHOSTSVR SERVER(*ALL).

- Następnie należy sprawdzić działanie QZDASRVSD, demona gniazda ODBC.
 - Demon działający jako baza danych powinien mieć status nasłuchiwania (Listen).
 - Za pomocą komendy WRKJOB QZDASRVSD należy sprawdzić, czy protokół zadania tego demona nie zawiera komunikatów o błędach.
- Należy sprawdzić, czy demon gniazda QZSOMAPD działa w podsystemie QSYSWRK.

- Demon działający jako program odwzorowujący serwer powinien mieć status nasłuchiwania (Listen), jak wskazuje komenda NETSTAT *CNN.
- Za pomocą komendy WRKJOB QZSOMAPD należy sprawdzić, czy protokół zadania tego demona nie zawiera komunikatów o błędach.

Komputer PC znajduje gniazdo używane przez serwer baz danych łącząc się z gniazdem programu odwzorowującego. Pobiera on numer gniazda używanego przez demon działający jako baza danych. Następnie łączy się z odpowiednim gniazdem monitorowanym przez demon serwera plików, QZDASRVSD. Demon serwera przypisze połączenie klienta do zadania prestartu QZDASOINIT w serwerze QUSRWRK. Po sprawdzeniu profilu użytkownika i hasła oraz wymianie profilu użytkownika z zadaniem prestartu, zostanie uruchomione zadanie podobne do zadania QZDASOINIT. Jeśli jest to pierwsze połączenie serwera z komputerem PC, wówczas używane są dwa inne serwery: serwer centralny służący do kontroli licencji i serwer wpisywania się do systemu służący do zatwierdzenia ID użytkownika i hasła.

Więcej informacji o sprawdzaniu działania warstwy TCP/IP zawiera sekcja Ogólne problemy z TCP/IP.

Typowe błędy ODBC

Poniższe tematy zawierają ogólne wytyczne dotyczące wyszukiwania i rozwiązywania najczęściej występujących błędów iSeries^(TM) Access for Windows^(R):

- Błędy SQL
- Błędy procedur zapisanych w bazie
- Niewłaściwe dane wejściowe ODBC i nieprzewidywalne błędy

Błędy SQL:

- SQL0104 - Element &1 był niepoprawny. Dopuszczalne elementy to: &2
- SQL0113 - Nazwa &1 niedopuszczalna
- SQL0114 - Relacyjna baza danych &1 nie jest taka sama jak bieżący serwer &2
- SQL0204 - MYSYSCONF jest nazwą niezdefiniowaną
- SQL0208 - Klauzula ORDER BY jest niepoprawna, ponieważ kolumna nie jest częścią tablicy wynikowej
- SQL0900 - Aplikacja jest w stanie błędu. Połączenie z bazą danych nie istnieje
- SQL0901 - Błąd systemowy SQL
- SQL5001 - Kwalifikator kolumny lub tabela &2 niezdefiniowana
- SQL5016 - Nazwa obiektu &1 niezgodna z konwencją nazewnictwa
- SQL7008 &1 in &2 nie jest poprawny dla operacji. Kod przyczyny 3

Uwaga: Więcej informacji na temat błędów SQL zawiera sekcja Komunikaty i kody SQL.

Błędy procedur zapisanych w bazie: Poniżej znajdują się błędy typowe dla procedur zapisanych w bazie:

- “SQL0444 - Program zewnętrzny &A w &B nie znaleziony (DB2 UDB for iSeries SQL)”
- “Nie zwrócono danych dla parametrów OUTPUT i INPUT_OUTPUT”
- “SQL0501 - Kursor CRSR000x nie jest otwarty” na stronie 21

SQL0444 - Program zewnętrzny &A w &B nie znaleziony (DB2 UDB for iSeries SQL): Komunikat SQL0444 jest generowany w przypadku wykonania lub bezpośredniego wykonania, gdy serwer baz danych może znaleźć deklarację procedury, ale nie może znaleźć obiektu programu. Program zewnętrzny musi być w miejscu podanym w systemowych tabelach katalogów. Należy zauważyć, że miejsce to jest definiowane za pomocą konwencji nazewnictwa i domyślnego zbierania informacji obowiązującego, gdy procedura jest definiowana (za pomocą CREATE PROCEDURE), a nie w momencie jej wywołania. Aby sprawdzić miejsce zdefiniowane dla nazwy programu zewnętrznego procedury zapisanej w bazie, należy uruchomić zapytanie dla QSYS2.SYSPROCS i zapisać wartość pola nazwy EXTERNAL_NAME”.

Nie zwrócono danych dla parametrów OUTPUT i INPUT_OUTPUT: Możliwe przyczyny błędu:

- Funkcja API ODBC **SQLBindParameter** niewłaściwie określiła **fParamType** jako SQL_PARAM_INPUT.
- Użyto procedury DECLARE PROCEDURE zamiast CREATE PROCEDURE i rozszerzona obsługa dynamiczna została wyłączona.
- Programista niewłaściwie zadeklarował parametr jako IN w procedurze CREATE lub DECLARE PROCEDURE.
- Program procedury zapisanej w bazie zwrócił niewłaściwy parametr.

SQL0501 - Cursor CRSR000x nie jest otwarty: Aby otrzymać dane, gdy używa się obiektu osadzonego SQL w programach ILE, należy podać opcję kompilacji ACTGRP(*CALLER), a nie domyślną opcję *NEW.

Należy sprawdzić, czy program wykonuje instrukcję return zamiast exit.

Gdy program procedury zapisanej w bazie uruchamia instrukcję exit zamiast return, należy zmienić opcję **Close SQL Cursor** na wartość *ENDACTGRP. Jeśli opcja Close SQL Cursor ma wartość *ENDMOD, kursor zostanie zamknięty zanim dane zostaną pobrane.

Należy także sprawdzić, czy procedura CREATE PROCEDURE określa poprawną liczbę zestawów wyników. Jest to ważne zwłaszcza w przypadku użycia zakresów zestawów wyników.

Niewłaściwe dane wejściowe ODBC i nieprzewidywalne błędy: Należy sprawdzić, czy kod sterownika ODBC iSeries^(TM) Access i programu serwera baz danych są w tej samej wersji. W zamówionych poprawkach PTF lub w pliku readme.txt w pakiecie serwisowym należy sprawdzić wymagania równoczesne. Jeśli błędy będą się powtarzać, należy sprawdzić, czy opcja pobrania z wyprzedzeniem w ODBC Data Source została wyłączona. Nie należy używać tej opcji, jeśli aplikacja używa funkcji API SQLExtendedFetch lub SQLFetchScroll ODBC lub gdy nie ma co do tego pewności.

Należy zauważyć, że *zestaw kursorów wynikowych* procedury zapisanej w bazie jest w trybie tylko do przesłania i tylko do odczytu.

Dane binarne lub szesnastkowe zamiast znaków ASCII

Wartość domyślna parametru Konwersja (Translation) nie powoduje przekształcenia danych binarnych (CCSID 65535) na tekst. Aby zidentyfikować tabelę konwersji używaną do konwersji danych, **CCSID jest przypisany do zbiorów, tabel, a nawet pól (kolumn).** CCSID 65535 często identyfikuje dane (binarne lub szesnastkowe), takie jak grafikę bitmapową, która jest niezależna od języka. Niewybranie opcji *Konwersja danych binarnych (CCSID 65535) na tekst* zapewnia, że zwykle dane nie zostaną zniszczone.

Ustawienie parametru translacji jako Konwersja danych binarnych (CCSID 65535) na tekst aktualizuje CCSID przypisany do danych zadania. Zmiana tego parametru może spowodować uszkodzenie danych, jeśli są to tylko dane binarne.

Zbieranie informacji dla działu wsparcia IBM

Aby dział wsparcia IBM^(R) mógł udzielić właściwej pomocy, zaleca się zebranie pewnych informacji, zanim zgłoszony zostanie rekord problemu. Aby zebrać te informacje, należy wykonać następujące czynności:

<p>Zapisz wersję OS/400^(R) i poziom zbiorczych poprawek PTF.</p>	<ol style="list-style-type: none"> 1. Wydadź komendę Wyświetlenie poprawek PTF (Display PTF) w wierszu komend emulatora terminalu: DSPPTF 2. Zapisz informacje o wydaniu OS/400 w formacie VxRxMx. 3. Sprawdź, czy źródłem IPL jest ##MACH#B. 4. Naciśnij klawisz F5, aby wyświetlić szczegóły poprawek PTF. 5. Zapisz pierwszy ID poprawki PTF znajdujący się na liście. Będzie on w formacie Tzxyyy, gdzie xx to rok, yyy to data juliańska, a z ma wartość L lub C.
---	---

Zapisz wersję sterownika ODBC.	<ol style="list-style-type: none"> 1. Z paska zadań wybierz opcję Start → Programy → IBM iSeries^(TM) Access for Windows^(R) → Administrowanie ODBC. Uwaga: W komputerze 64-bitowym ze sterownikiem 64-bitowym wybierz Administrowanie ODBC (64-bitowe). 2. Wybierz zakładkę Sterowniki. 3. Zapisz wersję sterownika ODBC iSeries Access.
Zapisz wersję menedżera sterownika ODBC.	<ol style="list-style-type: none"> 1. Z paska zadań wybierz opcję Start → Programy → IBM iSeries Access for Windows → Administrowanie ODBC. Uwaga: W komputerze 64-bitowym ze sterownikiem 64-bitowym wybierz Administrowanie ODBC (64-bitowe). 2. Wybierz zakładkę O programie. 3. Zapisz wersję menedżera sterownika.
Zbierz dane śledzenia.	Najczęściej zbierane do celów serwisowych dane śledzenia to: śledzenie ODBC (SQL.LOG), CWBCOTRC (Śledzenie komunikacji) oraz Śledzenie szczegółowe. Więcej informacji na temat śledzenia zawiera sekcja “Diagnostyka ODBC i narzędzia wydajności” na stronie 15.
Zapisz dodatkowe informacje.	Informacje dotyczące aplikacji, opis błędów i używanego sterownika ODBC (32- czy 64-bitowy).

Administrowanie serwerem hosta

Temat ten zawiera krótki opis funkcji serwera iSeries^(TM) oraz informacje techniczne związane z serwerami hostów używanymi przez produkt iSeries^(TM) Access for Windows^(R). Nie wszystkie serwery zostały opisane. Nie uwzględniono także wszystkich serwerów w systemie hosta (iSeries).

Serwery OS/400^(R)

Serwery hosta obsługują zgłoszenia napływające z komputerów PC lub urządzeń: uruchomienie aplikacji, wykonanie zapytania do bazy danych, wydrukowanie dokumentu czy nawet uruchomienie procedury składowania i odtwarzania. Komputery iSeries to w pełni funkcjonalne serwery, mające możliwość wykonywania wielu zadań jednocześnie, w tym zadań związanych z obsługą plików, baz danych, aplikacji, multimediiów, poczty, drukowania, wysyłania faksów i komunikacji bezprzewodowej. Gdy zadania te są obsługiwane przez kilka różnych serwerów, zarządzanie serwerami i koordynacja ich pracy stają się trudne. Umieszczenie funkcji wszystkich serwerów w jednym zintegrowanym systemie znacznie zmniejsza całkowity koszt i złożoność zarządzania siecią komputerową.

Wymienione serwery są wykorzystywane przez oprogramowanie iSeries Access for Windows, zostały jednak tak zaprojektowane, że mogą z nich korzystać także inne aplikacje klienckie. Tutaj opisujemy sposób wykorzystania tych serwerów przez oprogramowanie iSeries Access for Windows.

Dodawanie lub usuwanie opcji Serwer hosta OS/400

Omawiane tu serwery OS/400 są zoptymalizowane i zawarte w podstawowej opcji OS/400. Aby korzystać z funkcji iSeries Navigator aplikacji iSeries Access for Windows, należy zainstalować opcję Serwer hosta.

Jeśli nie jest używane oprogramowanie iSeries Access for Windows ani produkt iSeries NetServer, aby usunąć opcję Serwer hosta OS/400, należy najpierw zakończyć działanie podsystemów wykorzystywanych przez serwery. Należy zakończyć działanie podsystemu QBASE lub QCMN (w przypadku serwerów hosta obsługujących komunikację APPC), podsystemu QSYSWRK lub QUSRWRK (w przypadku serwerów hosta obsługujących komunikację za

pomocą gniazd) i podsystemu QSERVER (w przypadku serwerów baz danych i serwerów plików). Próba usunięcia opcji bez zakończenia pracy powyższych podsystemów może być przyczyną wystąpienia problemów.

“Serwery OS/400”

Opisano tu wiele serwerów hosta wspólnych dla klienta iSeries i obiektów pokrewnych. W aplikacji iSeries Access for Windows można uzyskać listę serwerów według ich typu lub funkcji.

“Korzystanie z serwerów hosta OS/400” na stronie 33

Opisano tu proces komunikacji klienta i serwera oraz sposób zarządzania nim. Ponadto dostępna jest lista poprawnych wartości i podsystemów systemu iSeries oraz opis sposobu identyfikacji, wyświetlania i zarządzania zadaniami serwerów w systemie iSeries.

“Używanie programów obsługi wyjścia serwera” na stronie 55

Opisano tu sposób tworzenia i rejestrowania programów obsługi wyjścia. Można tu także znaleźć parametry programów obsługi wyjścia i przykładowe programy.

Serwery OS/400

Poniższe informacje odnoszą się jedynie do serwerów używanych przez iSeries^(TM) Access for Windows. Nie dotyczą one wszystkich serwerów w systemie hosta (iSeries). Do serwerów hosta iSeries Access for Windows zaliczane są:

“Lista serwerów hosta według funkcji iSeries Access for Windows” na stronie 24

Serwery hosta według funkcji pełnionej w iSeries Access for Windows.

“Serwer plików” na stronie 25

Serwer plików umożliwia klientom przechowywanie i dostęp do informacji, takich jak pliki i programy, znajdujących się na serwerze iSeries.

“Serwer baz danych” na stronie 26

Służy do przesyłania danych, ODBC oraz do przechowywania baz danych iSeries Navigator, funkcji API SQL (funkcji API DB); używany także jako dostawca OLE DB i .NET Data w iSeries Access for Windows.

“Serwer kolejek danych” na stronie 30

Zapewnia dostęp do kolejek danych na serwerze iSeries.

“Wydruk sieciowy” na stronie 30

Obsługuje zdalne drukowanie i dodatkowe funkcje zarządzania drukowaniem.

“Serwer centralny” na stronie 31

Udostępnia usługi, takie jak zarządzanie licencjami i inne funkcje zarządzania klientami.

“Serwer komend zdalnych i rozproszonych wywołań programów” na stronie 32

Umożliwia aplikacjom komputerów PC wydawanie komend oraz wywoływanie programów w OS/400^(R) i zwracanie wyników do klienta.

“Serwer wpisywania się do systemu” na stronie 32

Udostępnia zarządzanie funkcjami obsługi haseł dla serwerów hosta z obsługą gniazd.

“Program odwzorowujący porty serwera” na stronie 32

Udostępnia bieżący numer portu serwera klientowi zgłaszającemu żądanie połączenia.



“Rozszerzony dynamiczny zdalny serwer SQL (QXDAEDRSQL)” na stronie 32

Streuje zdalnym dostępem do SQL oraz innymi funkcjami bazy danych.

“Serwer DRDA/DDM” na stronie 33

Umożliwia dostęp do funkcji zawartych w DB2^(R) UDB for iSeries. <<

Lista serwerów hosta według funkcji iSeries Access for Windows

Poniższa tabela zawiera podzbiór serwerów wykorzystywanych przez niektóre funkcje aplikacji iSeries^(TM) Access for Windows^(R).



Funkcja klienta	Używany serwer OS/400 ^(R)
Dostawca OLE .NET	<ul style="list-style-type: none">• “Serwer baz danych” na stronie 26• “Serwer wpisywania się do systemu” na stronie 32• “Serwer centralny” na stronie 31• “Rozszerzony dynamiczny zdalny serwer SQL (QXDAEDRSQL)” na stronie 32
IBM ^(R) Toolbox for Java ^(TM)	<ul style="list-style-type: none">• “Serwer wpisywania się do systemu” na stronie 32• “Serwer centralny” na stronie 31• “Serwer plików” na stronie 25• “Serwer baz danych” na stronie 26• “Serwer DRDA/DDM” na stronie 33• “Serwer kolejek danych” na stronie 30• “Serwer komend zdalnych i rozproszonych wywołań programów” na stronie 32• “Wydruk sieciowy” na stronie 30
Przesyłanie danych	<ul style="list-style-type: none">• “Serwer wpisywania się do systemu” na stronie 32• “Serwer centralny” na stronie 31• “Serwer baz danych” na stronie 26
Sterownik ODBC	<ul style="list-style-type: none">• “Serwer wpisywania się do systemu” na stronie 32• “Serwer baz danych” na stronie 26
Dostęp do zintegrowanego systemu plików z programu iSeries Navigator	“Serwer plików” na stronie 25
Funkcje API kolejek danych	“Serwer kolejek danych” na stronie 30
Dostawca OLE DB	<ul style="list-style-type: none">• “Serwer kolejek danych” na stronie 30• “Serwer baz danych” na stronie 26• “Serwer komend zdalnych i rozproszonych wywołań programów” na stronie 32• “Serwer wpisywania się do systemu” na stronie 32
Rozszerzony dynamiczny zdalny serwer SQL (QXDAEDRSQL)	<ul style="list-style-type: none">• “Serwer wpisywania się do systemu” na stronie 32• “Serwer centralny” na stronie 31• “Rozszerzony dynamiczny zdalny serwer SQL (QXDAEDRSQL)” na stronie 32

Funkeja klienta	Używany serwer OS/400 ^(R)
Zarządzanie licencjami Uruchamiane w momencie startu aplikacji wymagającej licencji (funkcje przesyłania danych i emulacji terminalu 5250).	“Serwer centralny” na stronie 31
Pobranie tabeli konwersji Wykonywane podczas pierwszego połączenia, jeśli klient nie zawiera wymaganych tablic konwersji.	“Serwer centralny” na stronie 31
Funkcje komend zdalnych	“Serwer komend zdalnych i rozproszonych wywołań programów” na stronie 32
Rozproszone wywołanie programu	“Serwer komend zdalnych i rozproszonych wywołań programów” na stronie 32
Wysłanie hasła w celu potwierdzenia i zmiana hasła, które wygasło (TCP/IP)	“Serwer wpisywania się do systemu” na stronie 32
Wydruk sieciowy GUI i interfejsy programistyczne	“Wydruk sieciowy” na stronie 30



Dodatkowe informacje zawiera dokument iSeries Access for Windows Servers and Ports Required, APAR II12227



Serwer plików

Serwer plików umożliwia klientom przechowywanie i dostęp do informacji, takich jak pliki i programy, znajdujących się na serwerze iSeries^(TM). Serwer plików OS/400^(R) wyposażony jest w interfejs do zintegrowanego systemu plików na serwerze iSeries. Do pracy z systemami plików klienci mogą używać własnych interfejsów zamiast interfejsów użytkowników zintegrowanego systemu plików i API.

Zintegrowany system plików jest częścią programu OS/400. Obsługuje on zarządzanie strumieniowymi danymi wejścia/wyjścia i pamięcią podobnie, jak systemy operacyjne komputerów osobistych i systemu UNIX^(R). Jednocześnie integruje wszystkie informacje przechowywane w serwerze iSeries.

Oto najważniejsze cechy zintegrowanego systemu plików:

- Obsługa informacji przechowywanych w plikach strumieniowych, zawierających długie, nieprzerwane ciągi danych. Mogą one być na przykład tekstem dokumentów lub elementami obrazów. Dokumenty przechowywane w folderach iSeries są plikami strumieniowymi. Innymi przykładami plików strumieniowych są pliki komputerów osobistych i pliki systemów UNIX. Obsługa plików strumieniowych została zaprojektowana w celu usprawnienia działania aplikacji klient/serwer.
- Hierarchiczna struktura katalogów umożliwiająca organizowanie obiektów w postaci gałęzi jednego drzewa. Aby uzyskać dostęp do obiektu, należy podać ścieżkę dostępu z katalogów do obiektu.
- Wspólny interfejs umożliwiający użytkownikom i aplikacjom dostęp do plików strumieniowych, zbiorów baz danych, dokumentów i innych obiektów przechowywanych w serwerze iSeries.

Serwery iSeries mogą obsługiwać kilka różnych systemów plików o podobnych interfejsach. System plików umożliwia użytkownikom i aplikacjom dostęp do konkretnych segmentów pamięci zorganizowanych jako jednostki logiczne. Jednostki te to pliki, katalogi, biblioteki i obiekty.

Listę systemów plików serwera iSeries zawiera sekcja Praca z systemami plików.

Więcej informacji na temat zintegrowanego systemu plików zawiera sekcja Zintegrowany system plików.

Serwer plików OS/400 może udostępniać klientom wszystkie systemy plików iSeries lub tylko system plików QDLS, zależnie od obsługi zapewnianej przez oprogramowanie klientów.

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Obiekty serwera plików

Nazwa programu	Biblioteka	Typ obiektu	Opis
QPWFSERVSO	QSYS	*PGM	Program serwera
QPWFSERVS2	QSYS	*PGM	Program serwera
QPWFSERVSD	QSYS	*PGM	Demon
QPWFSERV	QSYS	*JOB	Opis zadania używany dla zadań serwera
QPWFSEVER	QSYS	*CLS	Klasa używana dla wszystkich zadań serwera plików i baz danych
QPWFSERVSS	QSYS	*PGM	Program serwera SSL

Serwer baz danych

Serwer bazy danych umożliwia klientom dostęp do funkcji produktu **DB2^(R) UDB for iSeries(TM)^(TM)**. Serwer ten zapewnia:

- obsługę zdalnego dostępu do SQL,
- dostęp do danych za pomocą interfejsu ODBC, ADO, OLE DB oraz dostawcy OLE .NET,
- funkcje baz danych (takie jak tworzenie i usuwanie zbiorów oraz dodawanie i usuwanie podzbiorów),
- funkcje pobierania informacji o zbiorach baz danych istniejących w systemie (takie jak funkcje katalogowania SQL).

Ponadto w serwerze baz danych można używać architektury rozproszonych relacyjnych baz danych (Distributed Relational Database Architecture^(TM) - DRDA^(R)). DRDA nie pracuje z dostawcą OLE DB lub OLE .NET. Sekcja ta zawiera informacje o używaniu następujących elementów z DRDA:

- Pakiety SQL
- „Reguły i ograniczenia podczas używania DRDA” na stronie 30

Więcej informacji na temat DRDA zawiera sekcja Programowanie rozproszonych baz danych

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Programy serwera baz danych

Nazwa programu	Biblioteka	Opis
QZDASOINIT	QSYS	Program serwera
QZDASON2	QSYS	Program konfiguracyjny gniazd
QZDASRVSD	QSYS	Demon
QZDASSINIT	QSYS	Program serwera SSL

Uwaga: Obiekty *PGM QZDANDB oraz QZDACRTP wraz z obiektem *SRVPGM QZDASRV są używane przez serwer baz danych.

Pakiety SQL: Pakiety SQL przypisują instrukcje SQL w programie użytkowym do relacyjnej bazy danych. Wykorzystywane są do zwiększania wydajności aplikacji, które wykorzystują obsługę dynamicznego SQL przez umożliwienie aplikacji ponownego użycia informacji o żądaniach SQL. Serwer baz danych jest aplikacją, która wykorzystuje dynamiczne żądania SQL. Obsługuje użycie pakietów dla często używanych instrukcji SQL, co umożliwia wielokrotne użycie pewnych informacji o powiązaniach.

Więcej informacji zawierają poniższe tematy:

- “Nazwy pakietów SQL”
- “Procedura czyszcząca pakiety SQL” na stronie 29

Nazwy pakietów SQL: Serwer baz danych może być wykorzystany jako brama do innych relacyjnych baz danych wykorzystujących DRDA^(R). Serwer baz danych automatycznie tworzy jeden lub więcej pakietów SQL w docelowej relacyjnej bazie danych. Nazwy pakietów generowane są zgodnie z atrybutami aktualnie wykorzystywanymi przez serwer.

Nazwy pakietów, gdy relacyjna baza danych nie znajduje się na serwerze iSeries^(TM). Jeśli relacyjna baza danych (RDB) nie znajduje się na serwerze iSeries, pakiet tworzony jest w kolekcji o nazwie QSQL400 w serwerze aplikacji. Jeśli RDB znajduje się na serwerze iSeries, pakiet tworzony jest w bibliotece QGPL. Gdy serwer aplikacji nie jest serwerem iSeries, pakiet otrzymuje nazwę QZD**abcde**, gdzie **abcde** odpowiada zastosowanym opcjom analizatora składni. Poniższa tabela przedstawia opcje nazwy pakietu.

Pola opcji nazwy pakietu

Pole	Opis pola	Opcje
a	Format daty	<ul style="list-style-type: none"> • ISO, JIS • USA • EUR • JUL
b	Format godziny	<ul style="list-style-type: none"> • JIS • USA • EUR, ISO
c	Kontrola transakcji/separator dziesiętny	<ul style="list-style-type: none"> • *CS/kropka • *CS/przecinek • *CHG/kropka • *CHG/przecinek • *RR/kropka • *RR/przecinek
d	Ogranicznik łańcucha	<ul style="list-style-type: none"> • Apostrof • Cudzysłów
e	Maksymalna liczba instrukcji dla pakietu	<ul style="list-style-type: none"> • 0 - 64 • 1 - 256 • 2 - 512 • 3 - 1024

Nazwy pakietów, gdy relacyjna baza danych znajduje się na serwerze iSeries

Gdy serwer aplikacji jest serwerem iSeries, pakiet otrzymuje nazwę QZD**abcdef**, gdzie **abcdef** odpowiada użytej określonej opcji analizatora składni.

Pola opcji nazwy pakietu

Pole	Opis pola	Opcje
a	Format daty	<ul style="list-style-type: none">• ISO, JIS• USA• EUR• JUL• MDY• DMY• YMD
b	Format godziny i konwencja nazewnictwa	<ul style="list-style-type: none">• Nazewnictwo ISO, JIS i SQL• Nazewnictwo USA i SQL• Nazewnictwo EUR i SQL• Nazewnictwo HMS i SQL• Nazewnictwo ISO, JIS i systemowe• Nazewnictwo USA i systemowe• Nazewnictwo EUR i systemowe• Nazewnictwo HMS i systemowe
c	Poziom zatwierdzenia i separator dziesiętny	<ul style="list-style-type: none">• *CS/kropka• *CS/przecinek• *ALL/kropka• *ALL/przecinek• *CHG/kropka• *CHG/przecinek• *NONE/kropka• *NONE/przecinek
d	Ogranicznik łańcucha	<ul style="list-style-type: none">• Apostrof• Cudzysłów
e	Liczba sekcji w pakiecie	<ul style="list-style-type: none">• 0 - 64• 1 - 256• 2 - 512• 3 - 1024

Pole	Opis pola	Opcje
f	Separator daty i godziny	<ul style="list-style-type: none"> Najstarsze bity znaku: '1100'b - Jeden z formatów ISO dla daty '1101'b - Przecinek jest separatorem daty '1110'b - Kropka jest separatorem daty '1111'b - Dwukropek jest separatorem daty Najmłodsze bity znaku: '0001'b - Czas w formacie ISO '0010'b - Przecinek jest separatorem godziny '0011'b - Kropka jest separatorem godziny '0100'b - Ukośnik jest separatorem godziny '0101'b - Myślnik jest separatorem godziny '0110'b - Spacja jest separatorem godziny

Procedura czyszcząca pakiety SQL: Pakiety wykorzystywane w funkcjach DRDA^(R) tworzone są w systemie automatycznie w miarę potrzeb. Może zaistnieć potrzeba okresowego czyszczenia tych pakietów. Aby usunąć pakiety, należy użyć komendy Usunięcie pakietu SQL (Delete SQL Package - DLTSQLPKG).

Pakiety należy usuwać wtedy, gdy nie są one często używane. Jeśli pakiet będzie potrzebny, zostanie on automatycznie utworzony, ale wydajność znacznie spadnie, gdy będzie tworzony po raz drugi.

Konwencje nazewnictwa instrukcji: Poniższa tabela zawiera podsumowanie konwencji nazewnictwa wymuszonych przez serwer baz danych.

Konwencje nazewnictwa instrukcji

Instrukcja	Dynamiczny SQL	Użycie rozszerzonych dynamicznych pakietów SQL
Lokalne	<p>Nazwa instrukcji musi odpowiadać konwencji nazewnictwa iSeries^(TM), jednak zalecany jest format STMTxxxx.</p> <p>Nazwa kursora musi odpowiadać konwencji nazewnictwa iSeries.</p>	<p>Nazwa instrukcji musi odpowiadać konwencji nazewnictwa iSeries, jednak zalecany jest format STMTxxxx.</p> <p>Nazwa kursora musi odpowiadać konwencji nazewnictwa iSeries.</p>
DRDA ^(R)	<p>Nazwa instrukcji musi mieć format STMTxxxx.</p> <p>Nazwa kursora musi być w formacie:</p> <p>CRSRyyyy dla kursorów nieprzewijalnych lub SCRSRyyyy dla kursorów przewijalnych, gdzie yyyy jest równoważne xxxx.</p>	<p>Nazwa instrukcji musi mieć format Sxxxx.</p> <p>Nazwa kursora musi mieć format Cyy dla kursorów nieprzewijalnych, gdzie yy jest równoważne xxxx i yy ma wartość od 1 do 15.</p>

Uwagi:

- Konwencje nazewnictwa dla nazw instrukcji nie są wymuszane w systemie lokalnym, więc aplikacje klienckie mogą współużytkować przygotowane instrukcje z aplikacjami iSeries za pomocą systemowej funkcji API QSQPCED.
- Serwer wstawia znaki puste na początku nazwy każdej instrukcji w formacie STMTxxxx. Aplikacja hosta dodaje początkowe znaki puste do instrukcji współużytkowanych z aplikacjami klienckimi, które używają formatu STMTxxxx. Serwer nie dodaje znaków pustych na początku nazwy instrukcji, jeśli jest ona w formacie innym niż STMTxxxx.

Reguły i ograniczenia podczas używania DRDA: Architektura rozproszonych relacyjnych baz danych (Distributed Relational Database Architecture^(TM) - DRDA^(R)) jest architekturą umożliwiającą dostęp do innych baz danych obsługujących DRDA. Więcej informacji na temat DRDA zawiera sekcja Programowanie rozproszonych baz danych.

Gdy używa się serwera baz danych jako bramy do innych RDB za pomocą DRDA, na funkcje należy nałożyć pewne ograniczenia.

Poniższa tabela wymienia ograniczenia funkcji w sytuacji, gdy serwer baz danych połączony jest z serwerem zdalnym.

Limity funkcjonalne architektury DRDA

Funkcja	Ograniczenie
Utworzenie pakietu Kasowanie pakietu Usunięcie pakietu	Nieobsługiwane funkcje
Przygotowanie	Rozszerzona opcja przygotowania nie jest dostępna, gdy używany jest DRDA.
Rozszerzona obsługa pakietów dynamicznych	<ul style="list-style-type: none"> Podczas używania architektury DRDA nazwy instrukcji muszą mieć format 'STMTxxxx', gdzie xxxx jest numerem sekcji. Podczas używania architektury DRDA nazwy kursorów muszą mieć format 'CRSRxxxx', gdzie xxxx jest numerem sekcji.
Znaczniki opisu parametru	Dostępne tylko przy połączeniu z serwerem iSeries ^(TM) . Funkcja ta nie jest dostępna, gdy używany jest DRDA.
Zatwierdzenie wstrzymania	Dopuszczalne jedynie przy połączeniu z serwerem iSeries.
Poziom zatwierdzenia *NONE	Nie obsługiwany.
Poziom zatwierdzenia *CHANGE	Obsługiwany jedynie w przypadku, gdy docelowy RDB jest systemem iSeries. Wszystkie pozostałe RDB wymagają poziomu zatwierdzenia *CS lub *ALL.

Serwer kolejek danych

Kolejka danych jest obiektem używanym przez programy użytkowe iSeries^(TM) podczas komunikacji. Aplikacje mogą używać kolejek danych do przekazywania danych pomiędzy zadaniami. Jedna kolejka danych może obsługiwać dane z wielu zadań iSeries.

Oprogramowanie iSeries Access for Windows^(R) zawiera interfejsy API, które umożliwiają aplikacjom komputera PC pracę z kolejkami danych iSeries na równi z aplikacjami iSeries. Rozszerza to komunikację aplikacji iSeries o procesy działające na zdalnych komputerach PC.

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Program serwera do obsługi kolejek danych przeznaczony do pracy z obsługą gniazd

Nazwa programu	Biblioteka	Opis
QZHQSSRV	QSYS	Program serwera
QZHQSRVD	QSYS	Demon

Wydruk sieciowy

Wydruk sieciowy OS/400^(R) udostępnia zaawansowaną kontrolę klientów nad zasobami wydruków w serwerze iSeries^(TM). Serwer ten poprzez żądanie wydruku zapewnia każdemu klientowi następujące możliwości:

Zbiór buforowy

Tworzenie, przeszukiwanie, otwarcie, odczyt, zapis, zamknięcie, wstrzymanie, zwolnienie, usunięcie, przeniesienie, wysłanie, wywołanie programu obsługi wyjścia, zmiana atrybutów, wczytanie komunikatu, odpowiedź na komunikat, wczytanie atrybutów i wyświetlenie.

Program piszący

Uruchomienie, zakończenie i wyświetlenie.

Drukarka

Pobranie atrybutów i wyświetlenie.

Kolejka wyjściowa

Wstrzymanie, zwolnienie, usunięcie, wyświetlenie i pobranie atrybutów.

Biblioteka

Wyświetlenie.

Zbiór drukarkowy

Pobranie atrybutów, zmiana atrybutów i wyświetlenie.

Wydruk sieciowy

Zmiana atrybutów i pobranie atrybutów.

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Wydruk sieciowy

Nazwa programu	Biblioteka	Opis
QNPSERVS	QSYS	Program serwera
QNPSERVD	QSYS	Demon

Serwer centralny

Serwer centralny zapewnia klientom następujące usługi:

- Zarządzanie licencjami

Początkowe żądanie programu przesyłania danych lub PC5250 rezerwuje licencję dla danego użytkownika programu iSeries^(TM) Access for Windows^(R). Serwer pozostaje aktywny, dopóki limit czasu opóźnienia zwolnienia nie zostanie przekroczony. Licencja będzie zarezerwowana do momentu zwolnienia lub do czasu zakończenia zadania serwera. Aby sprawdzić, które licencje są zarezerwowane, za pomocą programu iSeries Navigator należy wyświetlić właściwości systemu iSeries.

- Pobranie tabeli konwersji

Serwer centralny pobiera mapy konwersji dla klientów, którzy ich potrzebują. Tabele te używane są zazwyczaj podczas konwersji znaków ASCII na EBCDIC i EBCDIC na ASCII. Należy podać identyfikatory CCSID. Klient może zażądać tabeli podając poprawny identyfikator CCSID źródłowego zestawu znaków oraz CCSID docelowego zestawu znaków, jak również tabelę punktów kodowych, które mają być przekształcone. Serwer zwraca klientowi poprawną tabelę konwersji.

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Programy serwera centralnego

Nazwa programu	Biblioteka	Opis
QZSCSRVS	QSYS	Program serwera
QZSCSRVSD	QSYS	Demon

Serwer komend zdalnych i rozproszonych wywołań programów

Serwer komend zdalnych i rozproszonych wywołań programów umożliwia użytkownikom i aplikacjom uruchamianie komend CL iSeries^(TM) i wywoływanie programów. Dzięki temu użytkownik może uruchomić wiele komend w tym samym zadaniu. Zapewnia to także lepszą kontrolę ochrony w przypadku użytkowników iSeries z profilem użytkownika o ograniczonych możliwościach (LMTCPB =*YES).

Obsługa rozproszonego wywołania programu umożliwia aplikacjom wywoływanie programów iSeries i przekazywanie im parametrów (wejścia i wyjścia). Gdy program działa na serwerze iSeries, wartości parametru wyjścia zwracane są aplikacji klienta. Proces ten umożliwia aplikacjom łatwy dostęp do zasobów iSeries bez konieczności obsługi komunikacji i konwersji, które muszą być przeprowadzane.

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Program serwera komend zdalnych i rozproszonych wywołań programów

Nazwa programu	Biblioteka	Opis
QZRCSRVS	QSYS	Program serwera
QZRCSRVD	QSYS	Demon

Serwer wpisywania się do systemu

Serwer wpisywania się do systemu zapewnia ochronę klientów. Ta funkcja ochrony blokuje dostęp do systemu użytkownikom, dla których wygasł okres ważności hasła, sprawdza hasła profili użytkowników i zwraca informacje o ochronie profilu użytkownika używane przez funkcje przechowywania haseł i funkcję administrowania aplikacjami iSeries^(TM) Navigator.

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Programy serwera wpisywania się do systemu

Nazwa programu	Biblioteka	Opis
QZSOSIGN	QSYS	Program serwera
QZSOSGND	QSYS	Demon

Program odwzorowujący porty serwera

Program odwzorowujący porty daje klientom możliwość odnalezienia portu danej usługi (serwera). Program odwzorowujący porty znajduje odpowiednie porty w Tabeli usług TCP/IP.

W poniższej tabeli znajduje się program włączony do tego serwera.

Serwer programu odwzorowującego porty

Nazwa programu	Biblioteka	Opis
QZSOSMAPD	QSYS	Serwer programu odwzorowującego porty

Rozszerzony dynamiczny zdalny serwer SQL (QXDAEDRSQL)



Serwer QXDAEDRSQL umożliwia klientom baz danych dostęp do funkcji produktu DB2^(R) UDB for iSeries^(TM). Serwer ten zapewnia:

- obsługę zdalnego dostępu do SQL,
- dostęp do danych za pomocą interfejsu XDA,
- funkcje baz danych (takie jak tworzenie i usuwanie zbiorów oraz dodawanie i usuwanie podzbiorów).

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Programy serwera QXDAEDRSQL

Nazwa programu	Biblioteka	Opis
QXDARECVR	QSYS	Program serwera
QXDALISTEN	QSYS	Demon

Uwaga: Obiekty QXDAEVT oraz QXDAIASP *SRVPGM są używane przez serwer QXDAEDRSQL.



Serwer DRDA/DDM



Serwer DRDA/DDM umożliwia klientom dostęp do funkcji produktu DB2^(R) UDB for iSeries(TM)^(TM). Serwer ten zapewnia:

- obsługę zdalnego dostępu do SQL,
- obsługę dostępu na poziomie rekordu,
- obsługę zdalnego dostępu do kroniki.

Więcej informacji na temat DRDA^(R) zawiera sekcja Rozproszone programowanie baz danych.

Dodatkowe informacje dotyczące DDM zawiera sekcja Zarządzanie danymi rozproszonymi.

Poniższa tabela zawiera wykaz programów wchodzących w skład serwera.

Programy serwera DRDA/DDM

Nazwa programu	Biblioteka	Opis
QRWTSRVR	QSYS	Program serwera
QRWTLSTN	QSYS	Program nasłuchujący



Korzystanie z serwerów hosta OS/400

Dokument ten opisuje zarządzanie zadaniami serwerów OS/400^(R). Opisuje podsystemy, w których uruchamiane są te serwery, obiekty, które mają na nie wpływ, oraz sposoby zarządzania tymi zasobami.

Serwery dostarczane z programem OS/400 zazwyczaj nie wymagają wprowadzania zmian w konfiguracji systemu i działają poprawnie. Są instalowane i konfigurowane w trakcie instalacji OS/400. Sposób, w jaki system zarządza zadaniami serwerów, można dostosować do własnych potrzeb, aby rozwiązać problemy, poprawić wydajność systemu lub po prostu przeglądać te zadania w systemie. Aby wprowadzić takie zmiany i spełnić wymagania przetwarzania, potrzebna jest wiedza o tym, które obiekty wpływają na które części systemu i jak zmienić te obiekty. Aby poznać zasady zarządzania systemem, przed wykonaniem czynności opisanych w niniejszym dokumencie należy zapoznać się z sekcją Zarządzanie pracą.

“Nawiązywanie komunikacji klient/serwer” na stronie 34

Proces uruchamiania i kończenia komunikacji pomiędzy klientami a serwerami. Dokument ten opisuje także “Numery portów serwerów hosta” na stronie 34 oraz demony serwera i ich rolę w komunikacji.

“Podsystemy w serwerze iSeries” na stronie 38

Informacje o podsystemach OS/400 oraz zadaniach autostartu i prestartu.

“Wartości systemowe serwera iSeries” na stronie 49

Informacje dotyczące wartości systemowych mających znaczenie w środowisku klient/serwer.

“Identyfikowanie zadań serwera w serwerze iSeries” na stronie 52

Informacje dotyczące sposobu wyświetlania zadań serwera za pomocą programu iSeries Navigator lub na ekranie tekstowym.

“Używanie programów EZ-Setup i iSeries Navigator z serwerami hosta” na stronie 54

Informacje dotyczące sposobu sprawdzania, czy wymagana ścieżka komunikacyjna jest aktywna, i uruchamiania jej w razie potrzeby.

Nawiązywanie komunikacji klient/serwer

Komunikację klient/serwer nawiązuje się wykonując następujące kroki:

1. Aby zainicjować zadanie serwera, które używa obsługi komunikacji gniazd, system klienta łączy się z portem serwera o danym numerze.
2. Demon serwera musi być uruchomiony (komenda STRHOSTSVR). Jego zadaniem jest nasłuchiwanie i akceptowanie żądań połączenia pochodzących od klienta. Po zaakceptowaniu żądania połączenia demon serwera wydaje wewnętrzne żądanie przypisania połączenia klienta do zadania serwera.
3. Zadanie to może być zadaniem prestartu lub, jeśli zadania prestartu nie są używane, zadaniem wsadowym uruchamianym, gdy przetwarzane jest żądanie połączenia klienta. Zadanie serwera obsługuje dalszą komunikację z klientem. Wstępna wymiana danych obejmuje żądanie identyfikujące profil użytkownika i hasło przypisane do użytkownika klienta.
4. Po sprawdzeniu poprawności profilu użytkownika i hasła, zadanie serwera przełącza się na dany profil użytkownika i zmienia zadanie za pomocą wielu atrybutów zdefiniowanych dla danego profilu, takich jak kod rozliczeniowy i kolejka wyjściowa.

Więcej informacji zawierają poniższe tematy:

- “Numery portów serwerów hosta”
- “Włączanie serwerów hosta” na stronie 35
- “Zamykanie serwerów hosta” na stronie 36

Komunikacja serwera z klientem

iSeries^(TM) Access for Windows^(R) używa protokołu TCP/IP w komunikacji z serwerami systemu iSeries. Serwery zoptymalizowane do komunikacji z klientem używają obsługi gniazd OS/400^(R). Obsługa gniazd OS/400 jest zgodna ze specyfikacją Berkeley Software Distributions 4.3 sockets over TCP/IP. Obsługa ta jest zapewniana wraz z produktem 5722-TC1 instalowanym na serwerze iSeries.

Więcej informacji dotyczących komunikacji zawiera podręcznik Konfigurowanie i obsługa TCP/IP.

Numery portów serwerów hosta: Każdy typ serwera ma swojego demona serwera, który nasłuchuje na porcie i obsługuje zgłoszenia klientów, którzy chcą się połączyć z serwerem. Są jednak wyjątki od tej zasady. Na przykład funkcja przesyłania za pomocą gniazd wykorzystuje demona serwera baz danych; serwer dysku sieciowego wykorzystuje demona serwera plików; wirtualny serwer wydruków wykorzystuje demona sieciowego serwera wydruków. Ponadto demon serwera programu odwzorowującego również nasłuchuje na określonym porcie i umożliwia klientowi uzyskanie bieżącego numeru portu dla określonego serwera.

Każdy z demonów serwera nasłuchuje na porcie o numerze określonym w tabeli usług dla danej nazwy usługi. Na przykład demon sieciowego serwera wydruków zgodnie z konfiguracją początkową nasłuchuje na porcie o numerze 8474, który jest skojarzony z usługą o nazwie 'as-netprt'. Demon programu odwzorowującego serwera nasłuchuje na ogólnie znanym porcie. Ogólnie znany port programu odwzorowującego serwera ma numer 449. Ogólnie znany numer portu zarezerwowany jest wyłącznie do użytku serwerów hosta OS/400^(R). Dlatego też pozycja 'as-svrmap' nie powinna być usuwana z tabeli usług.

Numery portów dla każdego demona serwera nie są stałe. Tabelę usług można modyfikować zmieniając numery portów, jeśli jest to konieczne. Numery portów można zmienić w programie iSeries^(TM) Navigator we właściwościach

systemu w zakładce połączenia. Nazwa usługi musi jednak pozostać taka sama, jak w poniższej tabeli. W przeciwnym razie demony serwera nie będą mogły ustalić gniazda, aby zaakceptować przychodzące zgłoszenia połączeń klientów.

Jeśli do tabeli usług zostanie dodana nowa pozycja identyfikująca inny port dla danej usługi, wszelkie istniejące wsześniejsze pozycje tabeli usług dla tej nazwy usługi powinny być usunięte. Usunięcie tych pozycji wyeliminuje zduplikowanie nazwy usługi w tabeli i wyeliminuje możliwość wystąpienia nieoczekiwanych wyników podczas uruchamiania demona serwera.

Numery portów dla serwerów hosta i programu odwzorowującego serwera

Numery portów zoptymalizowanych serwerów i programów odwzorowujących serwera wykorzystujących gniazda w obsłudze komunikacji TCP i warstwy SSL (Secure Sockets Layer).

Włączanie serwerów hosta: Aby uruchomić serwery hosta OS/400^(R), należy użyć komendy CL STRHOSTSVR. Rozpoczyna ona działanie demonów serwera hosta i demona programu odwzorowującego serwera. Komenda ta próbuje też uruchomić zadanie prestartu związane z danym serwerem.

Uwaga: Przy użyciu programu iSeries^(TM) Navigator można skonfigurować system w taki sposób, aby serwery uruchamiały się automatycznie po uruchomieniu protokołu TCP za pomocą komendy STRTCP. Nowe systemy będą robiły to domyślnie.

Każdy typ serwera hosta ma odpowiedni demon serwera. W systemie istnieje jeden demon programu odwzorowującego. Aplikacja klienta PC wykorzystuje numer portu do połączenia się z demonem serwera hosta. Demon serwera akceptuje nadchodzące żądanie połączenia i kieruje je do zadania serwera w celu dalszego przetwarzania.

Wartości dozwolone w komendzie STRHOSTSVR:



Typ serwera

***ALL** Uruchamia wszystkie demony serwera hosta i demon programu odwzorowującego serwera.

***CENTRAL**

Uruchamia demon serwera centralnego w podsystemie QSYSWRK. Zadanie demona to QZSCSRVSD, a związane z nim zadanie prestartu serwera to QZSCSRVS.

***DATABASE**

Uruchamia demon serwera baz danych w podsystemie QSERVER. Zadanie demona to QZDASRVSD, a związane z nim zadania prestartu serwera to QZDASOINIT i QTFPJTCP. QTFPJTCP działa w podsystemie QSERVER.

***DTAQ**

Uruchamia demon serwera kolejek danych w podsystemie QSYSWRK. Zadanie demona to QZHQSRVD, a związane z nim zadanie prestartu serwera to QZHQSSRV.

***FILE**

Uruchamia demon serwera plików w podsystemie QSERVER. Zadanie demona to QPWFSERVSD, a związane z nim zadania prestartu serwera to QPWFSERVSO, QPWFSERVSS oraz QPWFSERVS2.

***NETPRT**

Uruchamia demon sieciowego serwera drukarek w podsystemie QSYSWRK. Zadanie demona to QNPSERVSD, a związane z nim zadania prestartu serwera to QNPSERVS i QIWVPPJT. QIWVPPJT działa w podsystemie QSYSWRK.

***RMTCMD**

Uruchamia demon serwera komend zdalnych i wywołań programów w podsystemie QSYSWRK. Zadanie demona to QZRCSRVD, a związane z nim zadanie prestartu serwera to QZRCSRVS.

***SIGNON**

Uruchamia demon serwera logowania w podsystemie QSYSWRK. Zadanie demona to QZSOSGND, a związane z nim zadanie prestartu serwera to QZSOSIGN.

***SVRMAP**

Uruchamia demon programu odwzorowującego serwera w podsystemie QSYSWRK. Zadanie demona to QZSOSMAPD.

Uwaga: Jeśli zadanie demona jest uruchamiane w katalogu QSYSWRK, to związane z nim zadania prestartu serwera będą domyślnie uruchamiane również w katalogu QUSRWRK. Dodatkowo, w podsystemie QUSRWRK domyślnie będą działały zadania prestartu serwera baz danych.

Protokół wymagany

(Parametr opcjonalny określający, które protokoły komunikacyjne muszą być aktywne, aby można było uruchomić demony serwerów hosta).

***ANY** W momencie, gdy wydawana jest komenda STRHOSTSVR, musi być aktywny protokół komunikacyjny TCP/IP. Jeśli nie jest on aktywny, zostanie wysłany komunikat diagnostyczny PWS3008 i komunikat o przedwczesnym zakończeniu PWS300D, a demony serwera hosta nie zostaną uruchomione.

***NONE**

Aby demony serwerów hosta zostały uruchomione, w momencie uruchamiania komendy STRHOSTSVR protokoły komunikacyjne nie muszą być aktywne. Nie zostaną wysłane żadne komunikaty dotyczące nieaktywnych protokołów.

***TCP** W momencie, gdy wydawana jest komenda STRHOSTSVR, musi być aktywny protokół komunikacyjny TCP/IP. Jeśli nie jest on aktywny, zostanie wysłany komunikat diagnostyczny PWS3008 i komunikat o przedwczesnym zakończeniu PWS300D, a demony serwera hosta nie zostaną uruchomione.



“Przykład: Komenda STRHOSTSVR”.

Przykład: Komenda STRHOSTSVR: **Przykład 1: Uruchamianie wszystkich demonów serwera hosta:**

```
STRHOSTSVR(*ALL)
```

Ta komenda uruchamia wszystkie demony serwera i demon programu odwzorowującego serwera, jeśli aktywny jest choć jeden protokół komunikacyjny.

Przykład 2: Uruchamianie określonych demonów serwera:

```
STRHOSTSVR SERVER(*CENTRAL *SVRMAP) RQDPCL(*NONE)
```

Komenda ta rozpoczyna działanie demona serwera centralnego oraz demona programu odwzorowującego serwera nawet wtedy, gdy protokoły komunikacyjne nie są aktywne.

Przykład 3: Określenie jednego wymaganego protokołu:

```
STRHOSTSVR SERVER(*ALL) RQDPCL(*TCP)
```

Ta komenda rozpoczyna działanie demona serwera centralnego oraz demona programu odwzorowującego serwer, jeśli aktywny jest protokół TCP/IP.

Zamykanie serwerów hosta: Aby zakończyć działanie serwerów hosta OS/400^(R), należy użyć komendy CL ENDDHOSTSVR. Kończy ona działanie demonów serwerów hosta i demona programu odwzorowującego serwera. Jeśli demon serwera kończy pracę w momencie, gdy serwer komunikuje się z aplikacjami klienckimi, zadania serwera

pozostaną aktywne do czasu zakończenia komunikacji, chyba że zostanie podany opcjonalny parametr ENDACTCNN. Żądania połączenia z serwerem, napływające z aplikacji klienckich, nie będą obsługiwane do momentu ponownego uruchomienia demona tego serwera.

Zakończenie działania demona programu odwzorowującego serwera nie ma wpływu na istniejące połączenia między klientami a zadaniami serwera. Żądania połączenia z programem odwzorowującym, napływające z aplikacji klienckich, nie będą obsługiwane do momentu ponownego uruchomienia tego programu.

Aby zakończyć aktywne połączenia z serwerami *DATABASE i *FILE, można podać parametr ENDACTCNN. Spowoduje to zakończenie obsługujących te połączenia zadań serwera. Aktywne połączenia można zakończyć tylko pod warunkiem zakończenia odpowiadającego im zadania demona. Jeśli zostanie podane słowo kluczowe *DATABASE, zostaną zakończone zadania QZDASOINIT i QZDASSINIT z aktywnymi połączeniami. Jeśli zostanie podane słowo kluczowe *FILE, zostaną zakończone zadania QPWFSERVSO i QPWFSERVSS z aktywnymi połączeniami.

Uwaga:Jeśli użyje się komendy ENDDHOSTSVR do zakończenia działania demona, który nie jest aktywny, zostanie wygenerowany komunikat diagnostyczny. Aby zakończyć wszystkie aktywne demony, należy użyć komendy ENDDHOSTSVR SERVER(*ALL). W przypadku wartości *ALL komunikaty diagnostyczne nie są generowane.

Wartości używane w komendzie ENDDHOSTSVR: >>

Typ serwera

- *ALL** Kończy działanie demonów serwerów i demona programu odwzorowującego, jeśli jest on aktywny. Użycie tej wartości wyklucza jednoczesne użycie jakiegokolwiek innej wartości.
- *CENTRAL**
Kończy działanie centralnego demona serwera w podsystemie QSYSWRK.
- *DATABASE**
Kończy działanie demona serwera baz danych w podsystemie QSERVER.
- *DTAQ**
Kończy działanie demona serwera kolejek danych w podsystemie QSYSWRK.
- *FILE** Kończy działanie demona serwera plików w podsystemie QSERVER.
- *NETPRT**
Kończy działanie demona sieciowego serwera drukarek w podsystemie QSYSWRK.
- *RMTCMD**
Kończy działanie demona serwera komend zdalnych i rozproszonych wywoływań programów w podsystemie QSYSWRK.
- *SIGNON**
Kończy działanie demona serwera logowania w podsystemie QSYSWRK.
- *SVRMAP**
Kończy działanie demona programu odwzorowującego serwera w podsystemie QSYSWRK.

Koniec aktywnych połączeń

(Parametr opcjonalny, który określa, czy zostaną zakończone aktywne połączenia dla podanych serwerów).

Pojedyncze wartości:

- *NONE**
Aktywne połączenia nie zostaną zakończone.

Inne wartości:

***DATABASE**

Kończy działanie aktywnych połączeń obsługiwanych przez zadania serwera QZDASOINIT i QZDASSINIT. Spowoduje również zakończenie obsługujących te połączenia zadań serwera.

***FILE** Kończy działanie aktywnych połączeń obsługiwanych przez zadania serwera QPWFSERVO i QPWFSERVSS. Spowoduje również zakończenie obsługujących te połączenia zadań serwera.



“Przykład: komenda ENDDHOSTSVR” użycia komendy ENDDHOSTSVR.

Przykład: komenda ENDDHOSTSVR: **Przykład 1: Zakończenie wszystkich demonów serwera hosta**
ENDDHOSTSVR SERVER(*ALL)

Powoduje zakończenie działania wszystkich demonów serwera i demona programu odwzorowującego.

Przykład 2: Zakończenie działania określonych demonów

ENDDHOSTSVR SERVER(*CENTRAL *SVRMAP)

Powoduje zakończenie działania centralnego demona serwera oraz demona programu odwzorowującego.

Przykład 3: Zakończenie poszczególnych demonów serwera i aktywnych połączeń

ENDDHOSTSVR SERVER(*CENTRAL *DATABASE) ENDDACTCNN(*DATABASE)

Komenda ta powoduje zakończenie działania demona serwera centralnego w podsystemie QSYSWRK i demona serwera baz danych w podsystemie QSERVER. Ponadto zakończone zostaną aktywne połączenia z serwerem *DATABASE i zadania serwera QZDASOINIT i QZDASSINIT, które obsługują te połączenia.

Podsystemy w serwerze iSeries

Poniższe tematy opisują dostarczane wraz z systemem podsystemy, używane przez poszczególne funkcje serwera. Przedstawiają one szczegółowo zależności między opisami podsystemów i zadaniami serwera.

Opis podsystemu określa sposób pracy podsystemu i wykorzystywane przez niego zasoby.

Zadania autostartu wykonują jednorazowe inicjowanie lub wielokrotną pracę związaną z określonym podsystemem. Zadania autostartu powiązane z określonym podsystemem są automatycznie uruchamiane za każdym razem, gdy uruchamiany jest podsystem.

- Podsystemy używane dla zadań serwera
- Użycie zadań autostartu
- Użycie zadań prestartu

Podsystemy używane dla zadań serwera: Zadania serwera są skonfigurowane tak, aby uruchamiały się w różnych podsystemach, w zależności od wykonywanej funkcji. Zadania serwera wykorzystują następujące podsystemy.

QSYSWRK

Wszystkie zadania demonów (z wyjątkiem zadania demona serwera plików i zadania demona serwera baz danych) uruchamiane są w tym podsystemie. Zadania demonów serwera plików i baz danych uruchomione są w podsystemie QSERVER.

QUSRWRK

W tym podsystemie uruchamiane są zadania następujących serwerów:

- Wydruk sieciowy
- Serwer komend zdalnych i rozproszonych wywołań programów

- Serwer centralny
- Serwer kolejki danych
- Serwer wpisywania się
- Serwer baz danych

QSERVER

Zadanie demona serwera plików, powiązane z nim zadania prestartu serwera oraz zadanie demona serwera baz danych muszą być uruchomione w tym samym podsystemie.

Jeśli ten podsystem nie jest aktywny, zgłoszenia nawiązania połączenia z serwerem plików lub z serwerem baz danych nie powiedzą się.

Podsystemy uruchamiane automatycznie

Podsystem QSYSWRK uruchamia się automatycznie podczas IPL, niezależnie od wartości podanej dla podsystemu sterującego.

Jeśli jest używany domyślny program startowy dostarczony z systemem, podsystemy QSERVER i QUSRWRK uruchamiane są podczas IPL. Systemowy program startowy zdefiniowany jest przez wartość systemową QSTRUPPGM, która domyślnie ma wartość QSTRUP QSYS.

Aby zmienić sposób uruchamiania systemu, można zmienić wartość systemową QSTRUPPGM i wywoływać swój własny program. Jako bazę do stworzenia programu startowego można użyć dostarczonego w QSYS programu QSTRUP.

Uwaga: Jeśli używany jest serwer baz danych lub serwer plików i wprowadzono zmiany do programu startowego systemu, należy upewnić się, że program startowy uruchamia podsystem QSERVER.

Począwszy od wersji V5R1 protokół TCP/IP uruchamiany jest automatycznie przez system i nie ma potrzeby zmiany systemowego programu startowego. Serwery hosta uruchamiane są automatycznie podczas uruchamiania TCP/IP. Gdy protokół TCP/IP jest uruchomiony, podsystemy QUSRWRK i QSERVER zostaną uruchomione przed serwerami hosta. Jeśli instalacja V5R1 (lub nowszej wersji) nie powiodła się w systemie z zainstalowaną wcześniejszą wersją systemu operacyjnego i jeśli program startowy używany przez system został zmieniony tak, aby uruchamiał TCP/IP, system automatycznie uruchomi TCP/IP, a próba uruchomienia programu startowego nie powiedzie się. Odpowiednie ustawienie atrybutu IPL o nazwie STRTCP może sprawić, że system nie będzie automatycznie uruchamiał TCP/IP podczas IPL. Nie zaleca się pozostawiania jego wartości początkowej równej *YES (uruchomienie TCP/IP), ale opcja ta jest dostępna, w razie gdyby była potrzebna.

Użycie zadań autostartu: Podsystem QSERVER ma zadanie autostartu zdefiniowane dla zadań serwera plików i serwera baz danych. Jeśli zadanie to nie jest uruchomione, serwery nie mogą się uruchomić. Po zakończeniu zadania podsystem nie zostanie zakończony. Jeśli podczas wykonywania tego zadania wystąpi błąd, czasem trzeba zakończyć pracę podsystemu QSERVER i uruchomić go ponownie.

Podsystem QSYSWRK na zadanie autostartu zdefiniowane dla wszystkich zoptymalizowanych serwerów. Zadanie to monitoruje zdarzenia wysyłane podczas wydawania komendy STRTCP. Dzięki temu zadania demona serwera mogą dynamicznie określić, czy protokół TCP/IP jest aktywny. Zadania demona zaczynają wtedy nasłuchiwanie na odpowiednich portach. Jeśli zadanie autostartu nie jest aktywne, a protokół TCP/IP zostanie uruchomiony w trakcie działania serwerów hosta, to aby wykorzystać ten protokół, należy wydać następującą sekwencję komend:

1. ENHOSTSVR *ALL
2. STRHOSTSVR *ALL

Zadanie autostartu ma nazwę QZBSEVTM. Jeśli zadanie nie jest aktywne, można je uruchomić wydając następującą komendę:

```
QSYS/SBMJOB CMD(QSYS/CALL PGM(QSYS/QZBSEVTM)) JOB(QZBSEVTM) JOBD(QSYS/QZBSEJBD)
PRTDEV(*USRPRF) OUTQ(*USRPRF) USER(QUSER) PRTTXT(*SYSVAL)
        SYSLIBL(*SYSVAL)
CURLIB(*CRTDFT) INLLIBL(*JOBDD) SRTSEQ (*SYSVAL)
        LANGID(*SYSVAL) CNTRYID(*SYSVAL)
CCSID(*SYSVAL)
```

Uwaga: W tym samym czasie może być uruchomiona tylko jedna instancja programu QZBSEVTM.

Użycie zadań prestartu: Zadanie prestartu jest zadaniem wsadowym, które rozpoczyna działanie zanim program zdalnego systemu zainicjuje komunikację z serwerem. Zadania prestartu korzystają z pozycji zadań prestartu w opisie podsystemu, aby określić, który program, klasa i pula pamięci mają być używane podczas uruchamiania zadań. W pozycji zadania prestartu należy podać atrybuty, których podsystem ma użyć podczas tworzenia i zarządzania pulą zadań prestartu.

Zadania prestartu zwiększają wydajność w momencie rozpoczynania połączenia z serwerem. Zadania prestartu zdefiniowane są w podsystemie. Stają się one aktywne, gdy uruchamiany jest podsystem, ale także można je kontrolować przy pomocy komend Uruchomienie zadania prestartu (Start Prestart Job - STRPJ) oraz Zakończenie zadania prestartu (End Prestart Job - ENDPJ).

Informacja o systemie należąca do zadań prestartu (takich jak DSPACTPJ) używa terminu 'zgłoszenie uruchomienia programu' do wskazania zgłoszeń wysyłanych w celu uruchomienia zadań prestartu. Nawet wówczas informacja ta należy do zadania prestartu, które zostało uruchomione w wyniku zgłoszenia połączenia przez gniazdo.

Uwagi:

- Zadania prestartu mogą być powtórnie wykorzystane, ale nie są automatycznie czyszczone po zwróceniu do puli. Liczba powtórnych wykorzystania zadania prestartu określona jest przez wartość maksymalnej liczby użyć (MAXUSE) w komendach CL ADDPJE lub CHGPJE. Oznacza to, że zasób wykorzystany przez jednego użytkownika zadania prestartu musi być wyczyszczony zanim zadanie prestartu zakończy się. W przeciwnym razie zasoby będą miały ten sam status dla następnego użytkownika, który użyje zadania prestartu. Na przykład otwarty plik, który nigdy nie został zamknięty przez jednego użytkownika zadania prestartu, pozostanie otwarty i dostępny dla kolejnych użytkowników tego zadania prestartu.
- Domyślnie niektóre zadania serwera uruchomione są w podsystemie QUSRWRK lub QSERVER. Używając programu iSeries Navigator^(TM) można skonfigurować niektóre lub wszystkie serwery tak, aby pracowały w wybranym podsystemie.
 1. Kliknij dwukrotnie **iSeries Navigator** → **Sieć** → **Serwery** → **iSeries Access**.
 2. Kliknij prawym przyciskiem myszy serwer, dla którego chcesz skonfigurować podsystem, i wybierz **Właściwości**.
 3. Skonfiguruj serwer przy użyciu strony Podsystemy.
Przenosząc zadania z domyślnego podsystemu:
 1. Utwórz własny opis podsystemu.
 2. Za pomocą komendy ADDPJE dodaj swoje własne zadania prestartu. Parametr STRJOBS ustaw na *YES.Niewykonanie tych operacji spowoduje uruchomienie zadania w domyślnym podsystemie.

Wszystkie serwery OS/400^(R), które obsługiwane są przez interfejs komunikacji za pomocą gniazd, obsługują zadania prestartu.

Te serwery to:

- Sieciowy serwer wydruków
- Serwer komend zdalnych i rozproszonych wywołań programów
- Serwer centralny
- Serwer baz danych
- Serwer ochrony baz danych

Serwer plików
 Serwer ochrony plików
 Serwer kolejek danych
 Serwer wpisywania się do systemu (unikalny w porównaniu z serwerami wykorzystującymi obsługę komunikacji za pomocą gniazd)

Poniższe listy zawierają wszystkie atrybuty pozycji zadania prestartu oraz wartości początkowe, które są ustawiane dla serwerów hosta podczas obsługi komunikacji przy użyciu gniazd.

Opis podsystemu

Podsystem zawierający pozycje zadania prestartu.

Serwer OS/400	Wartość
Wydruk sieciowy	QUSRWRK
Serwer komend zdalnych i rozproszonych wywołań programów	QUSRWRK
Serwer centralny	QUSRWRK
Serwer baz danych	QUSRWRK
Serwer chronionych baz danych	QUSRWRK
Serwer plików	QSERVER
Serwer plików chronionych	QSERVER
Serwer kolejki danych	QUSRWRK
Serwer wpisywania się	QUSRWRK

Nazwa/Biblioteka programu

Program wywoływany jest podczas uruchamiania zadania prestartu.

Serwer OS/400	Wartość
Wydruk sieciowy	QSYS/QNPSERVS
Serwer komend zdalnych i rozproszonych wywołań programów	QSYS/QZRC SRVS
Serwer centralny	QSYS/QZSC SRVS
Serwer baz danych	QSYS/QZDASOINIT
Serwer chronionych baz danych	QSYS/QZDASSINIT
Serwer plików	QSYS/QPWFSE RVS
Serwer plików chronionych	QSYS/QPWFSE RVS
Serwer kolejki danych	QSYS/QZHQS SRV
Serwer wpisywania się	QSYS/QZSOSIGN

Profil użytkownika

Profil użytkownika, w którym uruchomione jest zadanie. Jest to wartość zadania profilu użytkownika. Gdy klient wysła zgłoszenie uruchomienia serwera, funkcje zadania prestartu przesyłane są na profil użytkownika, który wysłał zgłoszenie.

Serwer OS/400	Wartość
Wydruk sieciowy	QUSER
Serwer komend zdalnych i rozproszonych wywołań programów	QUSER
Serwer centralny	QUSER

Serwer OS/400	Wartość
Serwer baz danych	QUSER
Serwer chronionych baz danych	QUSER
Serwer plików	QUSER
Serwer plików chronionych	QUSER
Serwer kolejki danych	QUSER
Serwer wpisywania się	QUSER

Nazwa zadania

Nazwa uruchomionego zadania.

Serwer OS/400	Wartość
Wydruk sieciowy	*PGM
Serwer komend zdalnych i rozproszonych wywołań programów	*PGM
Serwer centralny	*PGM
Serwer baz danych	*PGM
Serwer chronionych baz danych	*PGM
Serwer plików	*PGM
Serwer plików chronionych	*PGM
Serwer kolejki danych	*PGM
Serwer wpisywania się	*PGM

Opis zadania

Opis zadania dla zadania prestartu. Po podaniu *USRPRF, użyty zostanie opis zadania dla profilu, w którym zadanie to jest uruchomione. Oznacza to, że użyty zostanie opis zadania QUSER. Z opisu zadania użytkownika, który wysłał zgłoszenie, pobierane są również inne atrybuty, na przykład drukarka czy kolejka wyjściowa.

Serwer OS/400	Wartość
Wydruk sieciowy	QSYS/QZBSJOB
Serwer komend zdalnych i rozproszonych wywołań programów	QSYS/QZBSJOB
Serwer centralny	QSYS/QZBSJOB
Serwer baz danych	QGPL/QDFTSVR
Serwer chronionych baz danych	QGPL/QDFTSVR
Serwer plików	QGPL/QDFTSVR
Serwer plików chronionych	QGPL/QDFTSVR
Serwer kolejki danych	QSYS/QZBSJOB
Serwer wpisywania się	QSYS/QZBSJOB

Zadania uruchamiania

Wskazują, czy zadania prestartu mają być automatycznie uruchamiane podczas uruchamiania podsystemu. Te pozycje zadań prestartu dostarczone są z wartością zadań startu równą *YES, co zapewnia, że zadania serwera są dostępne. Wszystkie zadania prestartu są traktowane jako elementy komendy STRHOSTSVR.

Serwer OS/400	Wartość
Wydruk sieciowy	*YES

Serwer OS/400	Wartość
Serwer komend zdalnych i rozproszonych wywołań programów	*YES
Serwer centralny	*YES
Serwer baz danych	*YES
Serwer chronionych baz danych	*YES
Serwer plików	*YES
Serwer plików chronionych	*YES
Serwer kolejki danych	*YES
Serwer wpisywania się	*YES

Początkowa liczba zadań

Liczba zadań uruchomionych podczas startu podsystemu. Wartość tę można dostosować do potrzeb określonego środowiska.

Serwer OS/400	Wartość
Wydruk sieciowy	1
Serwer komend zdalnych i rozproszonych wywołań programów	1
Serwer centralny	1
Serwer baz danych	1
Serwer chronionych baz danych	1
Serwer plików	1
Serwer plików chronionych	1
Serwer kolejki danych	1
Serwer wpisywania się	1

Próg

Minimalna liczba dostępnych zadań prestartu dla pozycji zadania prestartu. Gdy ten próg zostanie osiągnięty, automatycznie uruchomione zostaną dodatkowe zadania prestartu. Próg obsługuje określoną liczbę zadań w puli.

Serwer OS/400	Wartość
Wydruk sieciowy	1
Serwer komend zdalnych i rozproszonych wywołań programów	1
Serwer centralny	1
Serwer baz danych	1
Serwer chronionych baz danych	1
Serwer plików	1
Serwer plików chronionych	1
Serwer kolejki danych	1
Serwer wpisywania się	1

Dodatkowa liczba zadań

Liczba dodatkowych zadań prestartu uruchamianych, gdy zostanie osiągnięty próg.

Serwer OS/400	Wartość
Wydruk sieciowy	2
Serwer komend zdalnych i rozproszonych wywołań programów	2
Serwer centralny	2
Serwer baz danych	2
Serwer chronionych baz danych	2
Serwer plików	2
Serwer plików chronionych	2
Serwer kolejki danych	2
Serwer wpisywania się	2

Maksymalna liczba zadań

Maksymalna liczba zadań prestartu, które są aktywne dla tej pozycji.

Serwer OS/400	Wartość
Wydruk sieciowy	*NOMAX
Serwer komend zdalnych i rozproszonych wywołań programów	*NOMAX
Serwer centralny	*NOMAX
Serwer baz danych	*NOMAX
Serwer chronionych baz danych	*NOMAX
Serwer plików	*NOMAX
Serwer plików chronionych	*NOMAX
Serwer kolejki danych	*NOMAX
Serwer wpisywania się	*NOMAX

Maksymalna liczba użycia

Maksymalna liczba uruchomień zadania. Wartość 200 oznacza, że zadanie prestartu zakończy się po przetworzeniu 200 zgłoszeń uruchomienia serwera.

Serwer OS/400	Wartość
Wydruk sieciowy	200
Serwer komend zdalnych i rozproszonych wywołań programów	1
Serwer centralny	200
Serwer baz danych	200
Serwer chronionych baz danych	200
Serwer plików	*NOMAX
Serwer plików chronionych	*NOMAX
Serwer kolejki danych	200
Serwer wpisywania się	200

Oczekiwanie na zadanie

Jeśli osiągnięta została maksymalna liczba zadań, zgłoszenie połączenia klienta czeka na dostępne zadanie serwera.

Serwer OS/400	Wartość
Wydruk sieciowy	*YES
Serwer komend zdalnych i rozproszonych wywołań programów	*YES
Serwer centralny	*YES
Serwer baz danych	*YES
Serwer chronionych baz danych	*YES
Serwer plików	*YES
Serwer plików chronionych	*YES
Serwer kolejki danych	*YES
Serwer wpisywania się	*YES

Identyfikator puli

Identyfikator puli podsystemu, w którym uruchomione jest zadanie prestartu.

Serwer OS/400	Wartość
Wydruk sieciowy	1
Serwer komend zdalnych i rozproszonych wywołań programów	1
Serwer centralny	1
Serwer baz danych	1
Chroniona baza danych	1
Serwer plików	1
Chroniony plik	1
Kolejka danych	1
Serwer wpisywania się	1

Klasa

Nazwa i biblioteka klasy, w której uruchomione jest zadanie prestartu.

Serwer OS/400	Wartość
Wydruk sieciowy	QGPL/QCASERVER
Serwer komend zdalnych i rozproszonych wywołań programów	QGPL/QCASERVER
Serwer centralny	QGPL/QCASERVER
Serwer baz danych	QSYS/QPWFSEVER
Serwer chronionych baz danych	QSYS/QPWFSEVER
Serwer plików	QSYS/QPWFSEVER
Serwer plików chronionych	QSYS/QPWFSEVER
Serwer kolejki danych	QGPL/QCASERVER
Serwer wpisywania się	QGPL/QCASERVER

Gdy wartość zadań startu dla tej pozycji zadania prestartu ustawiona jest na *YES, a pozostałe parametry mają swoje wartości początkowe, dla każdej pozycji zadania prestartu wykonywane są następujące działania:

- Podczas uruchamiania podsystemu dla każdego serwera uruchomione zostaje jedno zadanie prestartu.
- Podczas przetwarzania pierwszego zgłoszenia połączenia klienta z określonym serwerem użyte zostaje pierwotne zadanie, co powoduje przekroczenie progu.

- Dla tego serwera uruchamiane są dodatkowe zadania, w oparciu o liczbę zdefiniowaną w pozycji zadania prestartu.
- Liczba dostępnych zadań zawsze wynosi przynajmniej jeden.
- Podsystem okresowo sprawdza liczbę zadań prestartu, które gotowe są do przetwarzania zgłoszeń, oraz kończy zadania przekroczone. Podsystem zawsze zostawia przynajmniej tyle zadań prestartu, ile określono w początkowym parametrze tych zadań.

Monitorowanie zadań prestartu

Do monitorowania zadań prestartu używa się komendy Wyświetlanie aktywnych zadań prestartu (Display Active Prestart Jobs - DSPACTPJ). Aby na przykład monitorować zadania prestartu serwera wpisując się do systemu, należy znać podsystem, w którym znajdują się zadania prestartu (QUSRWRK lub podsystem zdefiniowany przez użytkownika) oraz program (na przykład QZSOSIGN).

Komenda DSPACTPJ podaje następujące informacje:

```

+-----+
|                               Wyświetlanie aktywnych zadań prestartu                               |
|                               (Display Active Prestart Jobs)                               |
|                               AS400597                               |
|                               01/12/95 16:39:25                               |
| Podsystem . . . . . : QUSRWRK      Data resetu. . . . . : 01/11/95       |
| Program . . . . . : QZSOSIGN      Czas resetu. . . . . : 16:54:50       |
| Biblioteka . . . . . : QSYS       Upłynęło . . . . . : 0023:12:21       |
|                               |
| Zadania prestartu:                               |
| Bieżąca liczba . . . . . : 10                               |
| Średnia liczba . . . . . : 8.5                               |
| Szczytowa liczba . . . . . : 25                               |
|                               |
| Zadania prestartu w użyciu:                       |
| Bieżąca liczba . . . . . : 5                               |
| Średnia liczba . . . . . : 4.3                               |
| Szczytowa liczba . . . . . : 25                               |
|                               |
|                               Dalej...                               |
+-----+

```

```

+-----+
|                               01/12/95 16:39:25                               |
| Podsystem . . . . . : QUSRWRK      Data resetu. . . . . : 01/11/95       |
| Program . . . . . : QZSOSIGN      Czas resetu. . . . . : 16:54:50       |
| Biblioteka . . . . . : QSYS       Upłynęło . . . . . : 0023:12:21       |
|                               |
| Zgłoszenia uruchomienia programu:                 |
| Bieżąca liczba oczekujących. . . . . : 0                               |
| Średnia liczba oczekujących. . . . . : .2                               |
| Szczytowa liczba oczekujących. . . . . : 4                               |
| Średni czas oczekiwania. . . . . : 00:00:20.0                               |
| Liczba zaakceptowanych . . . . . : 0                               |
| Liczba odrzuconych . . . . . : 0                               |
|                               |
|                               Koniec                               |
| Naciśnij klawisz Enter, aby kontynuować.         |
| F3=Wyjdz F5=Odśwież F12=Anuluj F13=Resetuj statystykę |
+-----+

```

Zarządzanie zadaniami prestartu

Naciśnięcie klawisza **F5** na ekranie Wyświetlanie aktywnych zadań prestartu (Display Active Prestart Jobs) powoduje odświeżenie informacji dotyczących aktywnego zadania prestartu. Informacja o zgłoszeniach uruchomienia programu może wskazywać, czy należy zmienić liczbę dostępnych zadań prestartu. Jeśli informacja wskazuje, że zgłoszenia uruchomienia programu oczekują na dostępne zadanie prestartu, możesz zmienić zadania prestartu za pomocą komendy Zmiana pozycji zadania prestartu (Change Prestart Job Entry - CHGPJE).

Jeśli zgłoszenia uruchomienia programu nie są wykonywane dość szybko, można:

- zwiększyć próg,
- zmniejszyć wartość parametru początkowej liczby zadań (INLJOBS),
- zmniejszyć wartość parametru dodatkowej liczby zadań (ADLJOBS).

Najważniejsze jest zapewnienie istnienia dostępnego zadania prestartu dla każdego zgłoszenia.

Usuwanie pozycji zadania prestartu

Jeśli nie chcesz, aby serwery wykonywały zadania prestartu, wykonaj następujące działania:

1. Za pomocą komendy Zakończenie zadania prestartu (End Prestart Job - ENDPJ) zakończ zadania prestartu.

Zadania prestartu zakończone komendą ENDPJ zostaną uruchomione podczas następnego uruchomienia podsystemu, jeśli w pozycji zadania prestartu uruchamianie zadań ustawione jest na *YES lub jeśli dla określonego typu serwera wydano komendę STRHOSTSVR. Jeśli tylko zakończysz zadanie prestartu i nie wykonasz kolejnego kroku, wszelkie żądania uruchomienia określonego serwera nie powiodą się.

2. Za pomocą komendy Usuwanie pozycji zadania prestartu (Remove Prestart Job Entry - RMVPJE) usuń z opisu podsystemu pozycje zadań prestartu.

Pozycje zadań prestartu usunięte za pomocą komendy RMVPJE są całkowicie usuwane z opisu podsystemu. Po usunięciu pozycji, będą wykonywane nowe zgłoszenia dla serwera.

Użycie pozycji routingu

Gdy do podsystemu zostanie skierowane zadanie demona, wykorzystuje ono pozycje routingu z opisu podsystemu. Pozycje routingu dla zadań demona serwera hosta dodawane są do opisu podsystemu po wydaniu komendy STRHOSTSVR. Te zadania uruchamiane są w profilu użytkownika QUSER. W przypadku zadań demona wysłanych do podsystemu QSYSWRK, wykorzystywana jest kolejka zadań QSYSNOMAX. W przypadku zadań demona wysłanych do podsystemu QSERVER, wykorzystywana jest kolejka zadań QPWFSERVER.

Charakterystyka zadań serwera pobierana jest z pozycji ich zadania prestartu. Jeśli serwery nie wykorzystują zadań prestartu, zadania serwera uruchamiane są z charakterystyką odpowiadających mu zadań demona.

Poniżej podano konfiguracje początkowe wszystkich zadań demonów serwera w podsystemach dostarczonych przez IBM^(R).

Demon sieciowego serwera wydruków

Podsystem	QSYS/QSYSWRK
Kolejka zadań	QSYSNOMAX
Użytkownik	QUSER
Dane routingu	QNPSERVD
Nazwa zadania	QNPSERVD
Klasa	QGPL/QCASERVER
Numer kolejny	2538

Demon serwera komend zdalnych i rozproszonych wywołań programów

Podsystem	QSYS/QSYSWRK
Kolejka zadań	QSYSNOMAX
Użytkownik	QUSER
Dane routingu	QZRCRVSD
Nazwa zadania	QZRCRVSD
Klasa	QGPL/QCASERVER
Numer kolejny	2539

Demon serwera centralnego

Podsystem	QSYS/QSYSWRK
Kolejka zadań	QSYSNOMAX
Użytkownik	QUSER
Dane routingu	QZSCRVSD
Nazwa zadania	QZSCRVSD
Klasa	QGPL/QCASERVER
Numer kolejny	2536

Demon serwera baz danych

Podsystem	QSYS/QSERVER
Kolejka zadań	QPWFSEVER
Użytkownik	QUSER
Dane routingu	QZDASRVSD
Nazwa zadania	QZDASRVSD
Klasa	QSYS/QPWFSEVER
Numer kolejny	600

Demon serwera plików

Podsystem	QSYS/QSERVER
Kolejka zadań	QPWFSEVER
Użytkownik	QUSER
Dane routingu	QPWFSEVSD
Nazwa zadania	QPWFSEVSD
Klasa	QSYS/QPWFSEVER
Numer kolejny	200

Demon serwera kolejek danych

Podsystem	QSYS/QSYSWRK
Kolejka zadań	QSYSNOMAX
Użytkownik	QUSER
Dane routingu	QZHQRVD

Nazwa zadania	QZHQSRVD
Klasa	QGPL/QCASERVER
Numer kolejny	2537

Demon serwera wpisywania się

Podsystem	QSYS/QSYSWRK
Kolejka zadań	QSYSNOMAX
Użytkownik	QUSER
Dane routingu	QZSOSGND
Nazwa zadania	QZSOSGND
Klasa	QGPL/QCASERVER
Numer kolejny	2540

Demon programu odwzorowującego serwera

Podsystem	QSYS/QSYSWRK
Kolejka zadań	QSYSNOMAX
Użytkownik	QUSER
Dane routingu	QZSOSMAPD
Nazwa zadania	QZSOSMAPD
Klasa	QGPL/QCASERVER
Numer kolejny	2541

Wartości systemowe serwera iSeries

Wartość systemowa zawiera informacje sterujące pewnymi częściami systemu. Użytkownik może zmieniać te wartości i w ten sposób definiować środowisko robocze. Przykładami wartości systemowych są data systemowa i lista bibliotek.

Serwer iSeries^(TM) korzysta z wielu wartości systemowych. Szczególnie interesujące w środowisku klient/serwer są następujące wartości.

QAUDCTL

Sterowanie kontrolą. Ta wartość systemowa zawiera przełączniki włączające i wyłączające kontrolę na poziomie obiektu i użytkownika. Zmiany tej wartości systemowej mają natychmiastowy wpływ na system.

QAUDENDACN

Działanie po wystąpieniu błędu kroniki kontroli. Ta wartość systemowa określa działanie podejmowane przez system w przypadku wystąpienia błędu podczas wysyłania zapisu systemowej kroniki kontroli ochrony. Zmiany tej wartości systemowej mają natychmiastowy wpływ na system.

QAUDFRCLVL

Wymuszenie kroniki kontroli. Ta wartość systemowa określa liczbę zapisów, które mogą być umieszczone w kronice kontroli ochrony zanim nastąpi wymuszenie zapisu na dysk. Zmiany tej wartości systemowej mają natychmiastowy wpływ na system.

QAUDLVL

Poziom kontroli ochrony. Zmiany tej wartości systemowej mają natychmiastowy wpływ na wszystkie zadania działające w systemie.

QAUTOVRT

Określa, czy system powinien automatycznie tworzyć urządzenia wirtualne. Wartość ta używana jest dla funkcji tranzytu terminalu i sesji TELNET.

QCCSID

Identyfikator kodowanego zestawu znaków identyfikujący:

- konkretny zestaw identyfikatorów schematu kodowania,
- identyfikatory zestawu znaków,
- identyfikatory strony kodowej,
- dodatkowe informacje związane z kodowaniem, które jednoznacznie identyfikują reprezentację zestawu znaków graficznych wymaganą przez system.

Wartość ta oparta jest na języku zainstalowanym w systemie. Określa, czy dane muszą być przekształcone do innego formatu zanim zostaną przedstawione użytkownikowi. Wartością domyślną jest 65535 oznaczająca, że dane nie są przekształcane.

QCTLSBSD

Opis podsystemu sterującego.

QDSPSGNINF

Określa, czy po wpisaniu się do systemu za pomocą funkcji emulacji 5250 (funkcja stacji roboczej, PC5250) jest wyświetlany ekran informacji wpisywania się.

QLANGID

Domyślny identyfikator języka dla systemu. Określa on domyślny CCSID dla zadania użytkownika, jeśli CCSID zadania wynosi 65535. Klienci i serwery używają wartości domyślnego CCSID zadania w celu określenia odpowiedniej konwersji danych wymienianych między klientem a serwerem.

QLMTSECOFR

Określa, czy użytkownik z uprawnieniami specjalnymi do wszystkich obiektów (*ALLOBJ) lub serwisowymi (*SERVICE) może używać każdego urządzenia. Jeśli wartość ta wynosi 1, wszyscy użytkownicy z uprawnieniami specjalnymi *ALLOBJ lub *SERVICE muszą mieć nadane konkretne uprawnienia *CHANGE, aby używać danego urządzenia.

Wartość ta wpływa na urządzenia wirtualne w emulacji 5250. Wartość początkowa tego parametru wynosi 1. Aby umożliwić uprawnionym użytkownikom wpisywanie się do komputerów PC, należy nadać im konkretne uprawnienia do urządzenia i kontrolera używanego przez komputer PC lub zmienić tę wartość na 0.

QMAXSIGN

Określa liczbę kolejnych nieprawidłowych prób wpisania się do systemu dla użytkowników lokalnych i zdalnych. Gdy wartość QMAXSIGN zostanie osiągnięta, system określa podejmowane działanie na podstawie wartości systemowej QMAXSGNACN.

Jeśli wartość QMAXSGNACN wynosi 1 (zablokowanie urządzenia), wartość QMAXSIGN nie ma wpływu na użytkowników, którzy wpiszą niewłaściwe hasło na komputerze PC, gdy uruchamiają połączenie.

Jest to potencjalne zagrożenie ochrony ze strony użytkowników PC. Wartość QMAXSGNACN powinna wynosić 2 lub 3.

QMAXSGNACN

Wartość ta określa działanie podejmowane przez system, gdy liczba kolejnych nieprawidłowych prób wpisania się do systemu zostanie przekroczona dla dowolnego urządzenia. Można podać wartości: 1 (zablokowanie urządzenia), 2 (wyłączenie profilu użytkownika) lub 3 (zablokowanie urządzenia i wyłączenie profilu użytkownika). Wartość początkowa wynosi 3.

QPWDEXPITV

Okres ważności hasła (w dniach). Zmiany tej wartości systemowej mają natychmiastowy wpływ na system.

QPWDLMTAJC

Ogranicza użycie w hasle cyfr na sąsiednich miejscach. Zmiany tej wartości systemowej wprowadzane są w systemie podczas kolejnej zmiany hasła.

QPWDLMTCHR

Ogranicza użycie w hasle określonych znaków. Zmiany tej wartości systemowej wprowadzane są w systemie podczas kolejnej zmiany hasła.

QPWDLMTREP

Ogranicza użycie w hasle powtarzających się znaków. Zmiany tej wartości systemowej wprowadzane są w systemie podczas kolejnej zmiany hasła.

QPWDLVL

Określa poziom obsługi hasel w systemie, który obejmuje długość hasła obsługiwanego przez serwer iSeries, typ szyfrowania hasel i decyzję, czy hasła iSeries NetServer dla klientów Windows^(R) zostaną usunięte z systemu. Zmiany tej wartości systemowej wprowadzane są w systemie podczas najbliższego IPL.

Uwaga: Aby ustawić obsługę długich hasel, należy uprzednio zaktualizować we wszystkich klientach obsługę takich hasel (Express V5R1). W przeciwnym razie wszyscy klienci wersji starszych niż V5R1 nie będą mogli wpisać się do serwera iSeries.

QPWDMAXLEN

Maksymalna liczba znaków w hasle. Zmiany tej wartości systemowej wprowadzane są w systemie podczas kolejnej zmiany hasła.

QPWDMINLEN

Minimalna liczba znaków w hasle. Zmiany tej wartości systemowej wprowadzane są w systemie podczas kolejnej zmiany hasła.

QPWDPOSDIF

Steruje położeniem znaków w nowym hasle. Zmiany tej wartości systemowej wprowadzane są w systemie podczas kolejnej zmiany hasła.

QPWDRQDDGT

Wymaga wpisania cyfry w nowym hasle. Zmiany tej wartości systemowej wprowadzane są w systemie podczas kolejnej zmiany hasła.

QPWDRQDDIF

Określa, czy hasło musi się różnić od poprzednich hasel.

QPWDVLDPGM

Nazwa i biblioteka programu potwierdzania hasel w systemie komputerowym. Można podać nazwę obiektu i biblioteki. Zmiany tej wartości systemowej wprowadzane są w systemie podczas kolejnej zmiany hasła.

QRMTSIGN

Określa sposób, w jaki system obsługuje zdalne żądania wpisania się do systemu. Sesja TELNET jest przykładem zdalnego żądania wpisania się do systemu. Wartość ta określa następujące działania:

- ***FRCSIGNON***: Wszystkie zdalne sesje wpisania się do systemu muszą przeprowadzić standardowy proces wpisania się.
- ***SAMEPRF***: Dla funkcji tranzytu terminali 5250 lub funkcji stacji roboczych, gdy nazwy źródłowego i docelowego profilu użytkownika są takie same, zdalne logowanie się może zostać pominięte. Gdy używa się usługi TELNET, wpisanie się może zostać pominięte.
- ***VERIFY***: Po sprawdzeniu, czy użytkownik ma dostęp do systemu, system zezwala na pominięcie wpisania się użytkownika.
- ***REJECT***: Nie zezwala na żadne zdalne wpisanie się dla funkcji tranzytu terminalu 5250 lub funkcji stacji roboczej. Gdy parametr QRMTSIGN ma wartość *REJECT, użytkownik może wciąż wpisać się do systemu używając usługi TELNET. Sesje te będą podlegać normalnemu przetwarzaniu. Aby odrzucić wszystkie żądania usługi TELNET, należy zakończyć działanie serwerów TELNET.
- **'biblioteka programów'**: Użytkownik może podać program i bibliotekę (lub *LIBL), aby zdecydować, które zdalne sesje są akceptowane i które profile użytkowników mogą automatycznie wpisywać się do systemu, oraz z których miejsc. Opcja ta jest poprawna tylko dla tranzytu.

Wartość ta określa także nazwę uruchamianego programu, który określa dopuszczalne sesje zdalne.

Wartość początkowa to *FRCSIGNON. Aby użytkownicy mogli używać funkcji pominięcia wpisywania się do systemu emulatora 5250, należy zmienić tę wartość na *VERIFY.

QSECURITY

Poziom ochrony systemu. Zmiany tej wartości systemowej wprowadzane są w systemie podczas najbliższego IPL.

- 20 oznacza, że system do wpisania się wymaga hasła.
- 30 oznacza, że system wymaga zabezpieczenia hasłem do wpisania się i zabezpieczenia obiektu podczas każdego dostępu. Aby uzyskać dostęp do wszystkich zasobów systemu, wymagane są odpowiednie uprawnienia.
- 40 oznacza, że system wymaga zabezpieczenia hasłem do wpisania się i zabezpieczenia obiektu podczas każdego dostępu. Programy, które usiłują uzyskać dostęp do obiektów poprzez nieobsługiwane interfejsy, nie uzyskują go.
- 50 oznacza, że system wymaga zabezpieczenia hasłem do wpisania się, a użytkownicy muszą mieć odpowiednie uprawnienia, aby uzyskać dostęp do obiektów i zasobów systemu. Ochrona i integralność biblioteki QTEMP i obiektów z dziedziny użytkownika są wymuszane. Programy, które usiłują uzyskać dostęp do obiektów poprzez nieobsługiwane interfejsy lub usiłują przekazać obsługiwany interfejsom nieobsługiwane wartości parametrów, nie uzyskują dostępu.

QSTRUPGM

Program, który działa, gdy uruchamiany jest system lub podsystem sterujący. Program ten wykonuje funkcje konfiguracyjne, takie jak uruchamianie podsystemów.

QSYSLIBL

Systemowa część listy bibliotek. Ta część listy bibliotek jest przeszukiwana jako pierwsza. Niektóre funkcje klientów używają jej do wyszukiwania obiektów.

Identyfikowanie zadań serwera w serwerze iSeries

Używanie emulatora lub zwykłego ekranu utrudnia powiązanie zadania z konkretnym komputerem osobistym lub pojedynczą funkcją klienta. Zdolność identyfikowania konkretnych zadań jest podstawą do wykrywania przyczyn problemów i określania ich wpływu na wydajność. Za pomocą interfejsu iSeries Navigator można łatwo zidentyfikować zadania serwera.

1. Kliknij dwukrotnie ikonę **iSeries^(TM) Navigator**.
2. Otwórz **Sieć** klikając znak **plus (+)**.
3. Otwórz **Serwery** klikając znak **plus (+)**.
4. Wybierz typ serwerów, których zadania mają być wyświetlone (na przykład, TCP/IP lub iSeries^(TM) Access for Windows^(R)).
5. Gdy serwery zostaną wyświetlone w prawym panelu, prawym przyciskiem myszy należy kliknąć serwer, którego zadania mają być przeglądane, i wybrać opcję **Zadania serwera**. Otwarte zostanie następnie okno z zadaniami serwera i użytkownikiem, typem zadania, statusem, czasem i datą wpisania do systemu tego serwera.

Podane poniżej tematy zawierają informacje o sposobach identyfikowania zadań serwera za pomocą tradycyjnego interfejsu:

- “Podsystemy w serwerze iSeries” na stronie 38
- “Nazwy zadań iSeries”
- “Wyświetlanie zadań serwera” na stronie 53
- “Wyświetlanie protokołu historii” na stronie 54
- “Wyświetlanie zadań serwera dla użytkownika” na stronie 54

Nazwy zadań iSeries: Nazwa zadania stosowana w iSeries^(TM) składa się z trzech części:

- Uproszczona nazwa zadania
- ID użytkownika
- Numer zadania (w porządku rosnącym)

Zadania serwera spełniają następujące kryteria:

- Nazwa zadania
 - W przypadku zadań nie będących zadaniami prestartu, nazwa zadania serwera jest nazwą programu serwera.
 - zadania prestartu wykorzystują nazwę zdefiniowaną we wpisie zadania prestartu,
 - zadania uruchomione przez serwer wykorzystują nazwę opisu zadania lub w przypadku gdy są zadaniami wsadowymi, wykorzystują nazwę podaną (wykonuje to serwer plików).
- ID użytkownika
 - jest to zawsze QUSER, niezależnie od tego, czy użyto zadania prestartu, czy nie,
 - protokół zadania pokazuje, którzy użytkownicy korzystali z tego zadania.
- Numery zadania tworzą funkcje zarządzania pracą

Wyświetlanie zadań serwera: Aby zidentyfikować zadania serwera, można użyć dwóch metod. Pierwsza metoda polega na użyciu komendy WRKACTJOB. Druga metoda polega na wyświetleniu protokołu historii i określeniu, które zadanie używane jest przez którego klienta.

Wyświetlanie aktywnych zadań za pomocą komendy WRKACTJOB. Komenda WRKACTJOB pokazuje wszystkie aktywne zadania, a także demony serwera i demon programu odwzorowującego serwera.

Następujące rysunki pokazują przykładowy status komendy WRKACTJOB. Na rysunkach pokazane są tylko zadania związane z serwerami. Aby zobaczyć dostępne zadania prestartu, należy nacisnąć klawisz **F14**.

Na rysunkach pokazano następujące typy zadań:

- **(1)** - Demon programu odwzorowującego serwera
- **(2)** - Demony serwera
- **(3)** - Zadania prestartu serwera

```
+-----+
                Praca z zadaniami aktywnymi
                (Work with Active Jobs)                    AS400597
01/12/95  10:25:40
CPU %:    3.1      Upłynęło: 21:38:40  Zadania aktywne: 77

Wpisz opcje i naciśnij klawisz Enter.
 2=Zmiana 3=Wstrzymaj 4=Koniec 5=Praca z 6=Zwolnij 7=Wyświetlenie komunikatów
 8=Praca ze zbiorami buforowymi 13=Odłącz ...

Opc  Podsystem/Zad.  Użytkownik  Typ   CPU %  Funkcja      Status
-----
   .
   .
___  QSYSWRK          QSYS        SBS   .0     DEQW
___  (1) QZSOSMAPD     QUSER       BCH   .0     SELW
   .
___  (2) QZSOSGND     QUSER       BCH   .0     SELW
___  QZSCSRVSD       QUSER       BCH   .0     SELW
___  QZRCSRVD        QUSER       BCH   .0     SELW
___  QZHQSRVD        QUSER       BCH   .0     SELW
___  QNPSEVRD        QUSER       BCH   .0     SELW
   .
   .
___  QUSRWRK          QSYS        SBS   .0     DEQW
___  (3) QZSOSIGN     QUSER       PJ    .0     PSRW
___  QZSCSRVS        QUSER       PJ    .0     PSRW
___  QZRCSRVS        QUSER       PJ    .0     PSRW
___  QZHQSSRV        QUSER       PJ    .0     PSRW
___  QNPSEVRD        QUSER       PJ    .0     PSRW
___  QZDASOINIT      QUSER       PJ    .0     PSRW
   .
                                          Dalej...
+-----+
```


Informacje dotyczące konfigurowania połączenia używanego przez EZ-Setup znajdują się w pomocy online programu EZ-Setup.

Ścieżka komunikacyjna używana przez EZ-Setup wymaga trzech zadań QNEOSOEM, działających w podsystemie QSYSWRK. Podsystem QSYSWRK zawiera uruchamiane automatycznie zadanie dla tej ścieżki komunikacyjnej. Zadanie to, o nazwie QNEOSOEM, uruchamia dwa inne zadania o nazwie QNEOSOEM w podsystemie QSYSWRK. Jeśli jedno z zadań nie jest aktywne, należy je uruchomić przy pomocy następującej komendy:

```
QSYS/SBMJOB CMD(QSYS/CALL PGM(QSYS/QNEOSOEM)) JOB(QNEOSOEM)
JOB(QSYS/QNEOJOB)
    JOBQ(QSYS/QSYSNOMAX) PRTDEV(*JOB) OUTQ(*JOB)
USER(*JOB)
    PRRTXT(*JOB) SYSLIBL(*SYSVAL) INLLIBL(*JOB)
LOGCLPGM(*YES)
    MSGQ(*NONE) SRTSEQ(*SYSVAL) LANGID(*SYSVAL)
CNTRYID(*SYSVAL)
    CCSID(*SYSVAL)
```

Komenda ta w razie potrzeby uruchomi wszystkie trzy zadania QNEOSOEM.

Używanie programów obsługi wyjścia serwera

Programy obsługi wyjścia umożliwiają administratorom systemów kontrolowanie działań dostępnych dla użytkowników w każdym z podanych serwerów. Wszystkie serwery obsługują programy obsługi wyjścia utworzone przez użytkowników. Temat ten opisuje sposoby używania i konfigurowania programów obsługi wyjścia. Zawiera także przykłady programów, które mogą pomóc w kontroli dostępu do funkcji serwera.

- “Rejestrowanie programów obsługi wyjścia”
- “Pisanie programów obsługi wyjścia” na stronie 58
- “Parametry programów obsługi wyjścia” na stronie 59
- “Przykłady: Programy obsługi wyjścia” na stronie 75

Uwaga: Należy przeczytać sekcję Informacje dotyczące kodu, która zawiera istotne informacje prawne.

Rejestrowanie programów obsługi wyjścia

Aby wybrać programy obsługi wyjścia, które serwery mają wywołać, należy je zarejestrować. Program obsługi wyjścia można zarejestrować za pomocą narzędzia do rejestracji OS/400^(R).

Praca z narzędziem do rejestracji

Aby zarejestrować program obsługi wyjścia za pomocą narzędzia do rejestracji, należy użyć komendy Praca z informacjami rejestracyjnymi (Work with Registration Information - WRKREGINF).

```
+-----+
|                Praca z informacjami rejestracyjnymi                |
|                (Work with Registration Information - WRKREGINF)      |
|                                                                    |
| Wprowadź wartości i naciśnij klawisz Enter.                        |
|                                                                    |
| Punkt wyjścia . . . . . *REGISTERED                               |
| Format punktu wyjścia . . . . . *ALL                               |
| Wyjście . . . . . *                               Nazwa, generic*, *ALL |
|                                                                    |
|                                                                    |
+-----+
```

Naciśnij klawisz Enter, aby przejrzeć zarejestrowane punkty wyjścia.

```
+-----+
|                Praca z informacjami rejestracyjnymi                |
|                (Work with Registration Information)                  |
|                                                                    |
+-----+
```

```

Wpisz opcje i naciśnij klawisz Enter.
      5=Wyśw. punkt wyjścia 8=Praca z programami obsł. wyjścia

      Punkt          Format
      wyjścia       punktu
Opc  wyjścia       Zarejestr.  Tekst
-    QIBM_QCA_CHG_COMMAND  CHGC0100  *YES  Zmiana prog.obsługi wyj.komend
-    QIBM_QCA_RTV_COMMAND  RTVC0100  *YES  Odtw. progr.obsługi wyj.komend
-    QIBM_QHQ_DTAQ         DTAQ0100  *YES  Pier. serwer kolejki danych
-    QIBM_QIMG_TRANSFORMS  XFRM0100  *YES
-    QIBM_QJO_DLT_JRNRCV   DRCV0100  *YES  Usunięcie dziennika
-    QIBM_QLZP_LICENSE     LICM0100  *YES  Pierwotny serwer zarz. licenc.
-    QIBM_QMF_MESSAGE      MESS0100  *YES  Pierwotny serwer komunikatów
-    QIBM_QMH_REPLY_INQ    RPYI0100  *YES  Odp. na komunikat zapytania
8    QIBM_QNPS_ENTRY       ENTR0100  *YES  Wydruk sieciowy - pozycja
-    QIBM_QNPS_SPLF        SPLF0100  *YES  Wydruk sieciowy - bufor
-    QIBM_QOE_OV_USR_ADM   UADM0100  *YES  OfficeVision/400 Administrat

Komenda
===>

```

Aby pracować z programami obsługi wyjścia dla punktu wyjścia zdefiniowanego dla wybranego serwera, wybierz opcję 8.

```

Praca z programami obsługi wyjścia
(Work with Exit Programs)

Punkt wyjścia:  QIBM_QNPS_ENTRY          Format:  ENTR0100

Wpisz opcje i naciśnij klawisz Enter.
  1=Dodaj 4=Usuń  5=Wyświetl 10=Zastąp

      Numer          Format
      programu       punktu
Opc  programu       wyjścia   Biblioteka
1_
_____

(Nie znaleziono programów obsługi wyjścia)

```

Aby dodać program obsługi wyjścia do punktu wyjścia, użyj opcji 1.

Uwagi:

- Jeśli program obsługi wyjścia jest już zdefiniowany, należy usunąć go zanim zmieniona zostanie nazwa programu.
- Mimo że narzędzie do rejestracji może obsługiwać wiele programów zewnętrznych dla danego punktu wyjścia i nazwy formatu, serwery zawsze pobierają pierwszy program obsługi wyjścia.
- Należy zakończyć i restartować zadania prestartu, aby zmiana została uwzględniona.

```

Dodanie programu obsługi wyjścia (ADDEXITPGM)
(Add exit program )

Wprowadź wartości i naciśnij klawisz Enter.

Punkt wyjścia. . . . . > QIBM_QNPS_ENTRY
Format punktu wyjścia . . . . . > ENTR0100   Nazwa
Numer programu . . . . . > 1                 1-2147483647, *LOW, *HIGH
Program. . . . . MYPGM                       Nazwa
Biblioteka . . . . . MYLIB                     Nazwa, *CURLIB
THREADSAFE . . . . . *UNKNOWN                 *UNKNOWN, *NO, *YES
Działanie dla zad. wielowątk . . . *SYSVAL   *SYSVAL, *RUN, *MSG,

```

Tekst opisu	*BLANK
-----------------------	--------

-----+

Wpisz nazwę i bibliotekę programu dla danego punktu wyjścia.

Ten sam program może być używany dla wielu punktów wyjścia. Może on wykorzystywać dane przysyłane jako dane wejściowe do określenia sposobu obsługi różnych typów żądań.

Poniższy przykład udostępnia punkt wyjścia i nazwy formatu dla każdego z serwerów OS/400.

QIBM_QPWFS_FILE_SERV (Serwer plików)

Nazwa formatu	PWFS0100
Nazwa aplikacji	*FILESRV

QIBM_QZDA_INIT (Inicjowanie serwera baz danych)

Nazwa formatu	ZDAI0100
Nazwa aplikacji	*SQL

QIBM_QZDA_NDBI (Inicjowanie serwera baz danych)

Nazwy formatów	ZDAQ0100 ZDAQ0200
Nazwa aplikacji	*NDB

QIBM_QZDA_ROI1 (Żądanie serwera baz danych pobrania informacji o obiektach)

Nazwy formatów	ZDAR0100 ZDAR0200
Nazwa aplikacji	*RTVOBJINF

QIBM_QZDA_SQL1 (Żądania SQL serwera baz danych)

Nazwy formatów	ZDAQ0100
Nazwa aplikacji	*SQLSRV

QIBM_QZDA_SQL2 (Żądania SQL serwera baz danych)

Nazwy formatów	ZDAQ0200
Nazwa aplikacji	*SQLSRV

QIBM_QZHQ_DATA_QUEUE (Serwer kolejek danych)

Nazwa formatu	ZHQ00100
Nazwa aplikacji	*DATAQSRV

QIBM_QNPS_ENTRY (Wydruk sieciowy)

Nazwa formatu	ENTR0100
---------------	----------

Nazwa aplikacji	QNPSEVR
-----------------	---------

QIBM_QNPS_SPLF (Sieciowy serwer wydruków)

Nazwa formatu	SPLF0100
Nazwa aplikacji	QNPSEVR

QIBM_QZSC_LM (Żądania zarządzania licencjami serwera centralnego)

Nazwa formatu	ZSCL0100
Nazwa aplikacji	*CNTRLSRV

QIBM_QZSC_NLS (Żądania NLS serwera centralnego)

Nazwa formatu	ZSCN0100
Nazwa aplikacji	*CNTRLSRV

QIBM_QZSC_SM (Serwer licencji)

Nazwa formatu	ZSCS0100
Nazwa aplikacji	*CNTRLSRV

QIBM_QZRC_RMT (Serwer komend zdalnych i rozproszonych wywołań programów)

Nazwa formatu	CZRC0100
Nazwa aplikacji	*RMTSRV

QIBM_QZSO_SIGNONSRV (Serwer wpisywania się do systemu)

Nazwa formatu	ZSOY0100
Nazwa aplikacji	*SIGNON

Pisanie programów obsługi wyjścia

Po podaniu programu obsługi wyjścia serwery przed wykonaniem żądania przekazują następujące dwa parametry do programu obsługi wyjścia:

- 1-bajtowa wartość kodu zakończenia,
- struktura zawierająca informacje o żądaniu, inna dla każdego punktu wyjścia.

Te dwa parametry umożliwiają programowi obsługi wyjścia określenie, czy możliwe jest spełnienie żądania. Jeśli program obsługi wyjścia zwróci kod zakończenia X'F1', serwer akceptuje żądanie. Jeśli kod powrotu ma wartość X'F0', serwer odrzuca żądanie. Jeśli zwrócone zostaną wartości inne niż X'F1' lub X'F0', wynik będzie się różnił w zależności od serwera, do którego uzyskano dostęp.

Dla wielu serwerów i punktów wyjścia można używać tego samego programu. Program może określić, który serwer jest wywoływany i która funkcja używana na podstawie danych w strukturze drugiego parametru.

W sekcji "Parametry programów obsługi wyjścia" na stronie 59 opisane są struktury drugiego parametru wysyłanego do programów obsługi wyjścia. Informacje te będą pomocne w pisaniu własnych programów obsługi wyjścia.

Parametry programów obsługi wyjścia

W poniższych tematach opisano strukturę danych dla drugiego parametru w formatach punktów wyjścia dla każdego serwera OS/400^(R).

- “Serwer plików”
- “Serwer baz danych” na stronie 60
- “Serwer kolejek danych” na stronie 68
- “Sieciowy serwer wydruków” na stronie 69
- “Serwer centralny” na stronie 71
- “Serwer komend zdalnych i rozproszonych wywołań programów” na stronie 73
- “Serwer wpisywania się do systemu” na stronie 74

Serwer plików: Serwer plików ma jeden zdefiniowany punkt wyjścia:

Format PWFS0100 punktu wyjścia QIBM_QPWFS_FILE_SERV

Punkt wyjścia QIBM_QPWFS_FILE_SERV zdefiniowany jest tak, aby uruchamiał program obsługi wyjścia dla następujących typów zgłoszeń serwera plików:

- Zmiana atrybutów zbioru
- Utworzenie pliku strumieniowego lub utworzenie katalogu
- Usunięcie zbioru lub katalogu
- Wypisanie atrybutów zbioru
- Przeniesienie
- Otworzenie pliku strumieniowego
- Zmiana nazwy
- Przydzielenie konwersacji

Uwaga:

W przypadku serwera plików nazwa programu obsługi wyjścia rozpoznawana jest wtedy, gdy podsystem QSERVER jest uaktywniony. Po zmianie nazwy programu należy zatrzymać i zrestartować podsystem, aby zmiana odniosła skutek.

Format PWFS0100 punktu wyjścia QIBM_QPWFS_FILE_SERV

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku serwera plików wartość ta wynosi *FILESRV.
20	14	BINARY(4)	Żądana funkcja	Wykonywana funkcja: <ul style="list-style-type: none"> • X'0000' - Zgłoszenie zmiany atrybutów pliku/zbioru • X'0001' - Zgłoszenie utworzenia pliku strumieniowego lub katalogu • X'0002' - Zgłoszenie usunięcia zbioru lub katalogu • X'0003' - Zgłoszenie pokazania listy atrybutów pliku • X'0004' - Zgłoszenie przeniesienia • X'0005' - Zgłoszenie otwarcia pliku strumieniowego • X'0006' - Zgłoszenie zmiany nazwy • X'0007' - Zgłoszenie przydzielenia konwersacji

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
24	18	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla QIBM_QPWFS_FILE_SERV nazwa formatu to PWFS0100.
32	20	CHAR(4)	Dostęp do pliku	Jeśli zgłaszana funkcja ma wartość '5' (otwarcie), pole to posiada następującą strukturę: <ul style="list-style-type: none"> • Dostęp do odczytu, CHAR(1) X'F1' - Tak X'F0' - Nie • Dostęp do zapisu, CHAR(1) X'F1' - Tak X'F0' - Nie • Dostęp do odczytu/zapisu, CHAR(1) X'F1' - Tak X'F0' - Nie • Usuwanie dozwolone, CHAR(1) X'F1' - Tak X'F0' - Nie
36	24	BINARY(4)	Długość nazwy pliku	Długość nazwy pliku (następne pole). Maksymalnie, długość ta może wynosić 16MB.
40	28	CHAR(*)	Nazwa pliku	Nazwa pliku. Długość tego pola określona jest w polu Długość nazwy pliku (poprzednie pole). Nazwa pliku zwracana jest w zestawie znaków ISO/IEC 10646 (UCS — 2 Level 1), CCSID 61952.
<p>Uwaga:</p> <ul style="list-style-type: none"> • Format ten zdefiniowany jest przez podzbiór EPWFSEP w zbiorach H, QRPGRSRC, QRPGLSRC, QLBLSRC oraz QCBLESRC w bibliotece QSYSINC. • Więcej informacji o zestawie znaków ISO/IEC 10646 (UCS — 2 Level 1) znaleźć można w książce <i>Information Standard, ISO/IEC 10646 — 1: Information technology — Universal — Octet Character Set (UCS) — Part 1: Architecture and Basic Multilingual Plane</i>, numer odniesienia ISO/IEC 10646 — 1: 1993(E). <p>Funkcje API dokonujące konwersji z zestawu znaków UCS — 2 Level 1 to iconv() oraz CDRCVRT.</p>				

Serwer baz danych: W serwerze baz danych jest zdefiniowanych pięć różnych punktów wyjścia:

1. QIBM_QZDA_INIT
 - Wywoływany podczas inicjowania serwera
2. QIBM_QZDA_NDB1
 - Wywoływany podczas rodzimych zgłoszeń do bazy danych
3. QIBM_QZDA_SQL1
 - Wywoływany podczas zgłoszeń SQL
4. QIBM_QZDA_SQL2
 - Wywoływany podczas zgłoszeń SQL
5. QIBM_QZDA_ROI1
 - Wywoływany podczas zgłoszeń pobrania informacji o obiekcie i podczas funkcji katalogowania SQL

Punkty wyjścia z rodzimej bazy danych i z pobierania informacji o obiekcie mają zdefiniowane dwa formaty, w zależności od typu żądanej funkcji.

Punkt wyjścia QIBM_QZDA_INIT zdefiniowany jest tak, aby uruchamiał program obsługi wyjścia podczas inicjowania serwera. Jeśli dla tego punktu wyjścia jest zdefiniowany program, wywoływany jest on za każdym razem, gdy inicjowany jest serwer baz danych.

Format ZDAI0100 punktu wyjścia QIBM_QZDA_INIT

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	Dla tego punktu wyjściowego wartością tą jest *SQL.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. W przypadku QIBM_QZDA_INIT nazwa formatu to ZDAI0100.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. Jedyna poprawna wartość dla tego punktu wyjścia to 0.
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZDAEP w zbiorach H, QRPGRS, QRPGLSRC, QLBSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Punkt wyjścia QIBM_QZDA_NDB1 jest zdefiniowany w taki sposób, aby uruchamiał program obsługi wyjścia dla zgłoszeń rodzimej bazy danych do serwera bazy danych. Dla tego punktu wyjścia zdefiniowane są dwa formaty. Format ZDAD0100 używany jest w przypadku następujących funkcji:

- Tworzenie źródłowego zbioru fizycznego
- Tworzenie zbioru bazy danych w oparciu o istniejący zbiór
- Dodawanie, czyszczenie i usuwanie podzbioru zbioru bazy danych
- Nadpisywanie zbioru bazy danych
- Usuwanie nadpisanego zbioru bazy danych
- Usuwanie zbioru

Format ZDAD0200 jest używany, gdy odebrane zostało zgłoszenie dodania bibliotek do listy bibliotek.

Format ZDAD0100 punktu wyjścia QIBM_QZDA_NDB1

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	Dla tego punktu wyjściowego wartością tą jest *NDB.
20	14	CHAR(8)	Nazwa formatu	Użyta nazwa formatu wyjścia obsługiwanego przez użytkownika. Dla następujących funkcji nazwa formatu to ZDAD0100.

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. Pole to zawiera jedną z następujących wartości: <ul style="list-style-type: none"> • X'00001800' - Tworzenie źródłowego zbioru fizycznego • X'00001801' - Tworzenie zbioru bazy danych • X'00001802' - Dodawanie podzbioru bazy danych • X'00001803' - Czyszczenie podzbioru bazy danych • X'00001804' - Usunięcie podzbioru bazy danych • X'00001805' - Przesłonięcie zbioru bazy danych • X'00001806' - Usunięcie przesłonięcia zbioru bazy danych • X'00001807' - Tworzenie zbioru składowania • X'00001808' - Czyszczenie zbioru składowania • X'00001809' - Usunięcie zbioru
32	20	CHAR(128)	Nazwa pliku	Nazwa zbioru użytego do żądanej funkcji.
160	A0	CHAR(10)	Nazwa biblioteki	Nazwa biblioteki zawierającej ten zbiór.
170	AA	CHAR(10)	Nazwa podzbioru	Nazwa dodawanego, czyszczonego lub usuwanego podzbioru.
180	B4	CHAR(10)	Uprawnienia	Uprawnienia do tworzonego zbioru.
190	BE	CHAR(128)	Na podstawie nazwy zbioru	Nazwa zbioru użytego podczas tworzenia zbioru w oparciu o istniejący zbiór.
318	13E	CHAR(10)	Na podstawie nazwy biblioteki	Nazwa biblioteki zawierającej zbiór, w oparciu o który tworzy się nowy zbiór.
328	148	CHAR(10)	Nazwa nadpisywanego zbioru	Nazwa zbioru, który ma zostać nadpisany.
338	152	CHAR(10)	Nazwa nadpisywanej biblioteki	Nazwa biblioteki zawierającej nadpisywany zbiór.
348	15C	CHAR(10)	Nazwa nadpisywanego podzbioru	Nazwa podzbioru, który ma zostać nadpisany.
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZDAEP w zbiorach H, QRPGRS, QRPGLSRC, QLBLSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Format ZDAD0200 punktu wyjścia QIBM_QZDA_NDB1

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	Dla tego punktu wyjściowego wartością tą jest *NDB.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla funkcji dodawania do listy bibliotek, nazwa formatu to ZDAD0200.

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. X'0000180C' - Dodawanie listy bibliotek
32	20	BINARY(4)	Liczba bibliotek	Liczba bibliotek (następne pole).
36	24	CHAR(10)	Nazwa biblioteki	Nazwy wszystkich bibliotek.
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZDAEP w zbiorach H, QRPGRS, QRPGLSRC, QLBSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Punkt wyjścia QIBM_QZDA_SQL1 jest zdefiniowany w taki sposób, aby uruchamiał program obsługi wyjścia dla określonych zgłoszeń SQL, które są odbierane przez serwer baz danych. Dla tego punktu wyjścia zdefiniowany jest tylko jeden format. Następujące funkcje powodują wywołanie programu obsługi wyjścia:

- Przygotowanie
- Otwarcie
- Wykonanie
- Połączenie
- Utworzenie pakietu
- Kasowanie pakietu
- Usunięcie pakietu
- Przechwytywanie strumienia
- Natychmiastowe wykonanie
- Przygotowanie i opis
- Przygotowanie i wykonanie lub przygotowanie i otwarcie
- Otwarcie i przechwycenie
- Wykonanie lub otwarcie
- Pobranie informacji o pakiecie SQL

Format ZDAQ0100 punktu wyjścia QIBM_QZDA_SQL1

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	Dla tego punktu wyjściowego wartością tą jest *SQLSRV.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla QIBM_QZDA_SQL1 format nazywa się ZDAQ0100.

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. Pole to zawiera jedną z następujących wartości: <ul style="list-style-type: none"> • X'00001800' - Przygotowanie • X'00001803' - Przygotowanie i opis • X'00001804' - Otwarcie/Opis • X'00001805' - Wykonanie • X'00001806' - Natychmiastowe wykonanie • X'00001809' - Połączenie • X'0000180C' - Przechwycenie strumienia • X'0000180D' - Przygotowanie i wykonanie • X'0000180E' - Otwarcie i przechwycenie • X'0000180F' - Utworzenie pakietu • X'00001810' - Wyczyszczenie pakietu • X'00001811' - Usunięcie pakietu • X'00001812' - Wykonanie lub otwarcie • X'00001815' - Pobranie informacji o pakiecie SQL
32	20	CHAR(18)	Nazwa instrukcji	Nazwa instrukcji użytej w funkcjach przygotowania lub wykonania.
50	32	CHAR(18)	Nazwa kursora	Nazwa kursora użytego w funkcji otwarcia.
68	44	CHAR(2)	Opcja przygotowania	Opcja użyta w funkcji przygotowania.
70	46	CHAR(2)	Atrybuty otwierania	Opcja użyta podczas funkcji otwierania.
72	48	CHAR(10)	Rozszerzona dynamiczna nazwa pakietu	Nazwa rozszerzonego dynamicznego pakietu SQL.
82	52	CHAR(10)	Nazwa biblioteki pakietu	Nazwa biblioteki dla rozszerzonego dynamicznego pakietu SQL.
92	5C	Binary(2)	Indykator DRDA ^(R)	<ul style="list-style-type: none"> • 0 - Podłączony do lokalnej bazy RDB • 1 - Podłączony do zdalnej bazy RDB
94	5E	CHAR(1)	Poziom kontroli transakcji	<ul style="list-style-type: none"> • 'A' - Zatwierdzenie *ALL • 'C' - Zatwierdzenie *CHANGE • 'N' - Zatwierdzenie *NONE • 'S' - Zatwierdzenie *CS (stabilność kursora - cursor stability) • 'L' - Zatwierdź *RR (odczyt powtarzalny)
95	5F	CHAR(512)	Pierwsze 512 bajtów tekstu instrukcji SQL	Pierwsze 512 bajtów instrukcji SQL.
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZDAEP w zbiorach H, QRPGRSRC, QRPGLSRC, QLBSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Punkt wyjścia QIBM_QZDA_SQL2 zdefiniowany jest tak, aby uruchamiał punkt wyjścia określonych zgłoszeń SQL, jakie odbierane są przez serwer baz danych. Punkt wyjścia QIBM_QZDA_SQL2 ma pierwszeństwo przed punktem wyjścia QIBM_QZDA_SQL1. Jeśli program zarejestrowany jest dla punktu wyjścia QIBM_QZDA_SQL2, będzie on wywołany, a program dla punktu wyjścia QIBM_QZDA_SQL1 nie będzie uruchomiony. Następujące funkcje powodują wywołanie programu obsługi wyjścia:

- Przygotowanie
- Otwarcie
- Wykonanie
- Połączenie
- Utworzenie pakietu
- Kasowanie pakietu
- Usunięcie pakietu
- Przechwytywanie strumienia
- Natychmiastowe wykonanie
- Przygotowanie i opis
- Przygotowanie i wykonanie lub przygotowanie i otwarcie
- Otwarcie i przechwycenie
- Wykonanie lub otwarcie
- Pobranie informacji o pakiecie SQL

Tabela A-6. Format ZDAQ0200 punktu wyjścia QIBM_QZDA_SQL2

0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	Dla tego punktu wyjściowego wartością tą jest *SQLSRV.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla punktu wyjścia QIBM_QZDA_SQL2 nazwą formatu jest ZDAQ0200.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. Pole to zawiera jedną z następujących wartości: <ul style="list-style-type: none"> • X'00001800' - Przygotowanie • X'00001803' - Przygotowanie i opis • X'00001804' - Otwarcie/Opis • X'00001805' - Wykonanie • X'00001806' - Natychmiastowe wykonanie • X'00001809' - Połączenie • X'0000180C' - Przechwycenie strumienia • X'0000180D' - Przygotowanie i wykonanie • X'0000180E' - Otwarcie i przechwycenie • X'0000180F' - Utworzenie pakietu • X'00001810' - Wyczyszczenie pakietu • X'00001811' - Usunięcie pakietu • X'00001812' - Wykonanie lub otwarcie • X'00001815' - Pobranie informacji o pakiecie SQL
32	20	CHAR(18)	Nazwa instrukcji	Nazwa instrukcji użytej w funkcjach przygotowania lub wykonania.
50	32	CHAR(18)	Nazwa kursora	Nazwa kursora użytego w funkcji otwarcia.
68	44	CHAR(2)	Opcja przygotowania	Opcja użyta w funkcji przygotowania.

70	46	CHAR(2)	Atrybuty otwierania	Opcja użyta podczas funkcji otwierania.
72	48	CHAR(10)	Rozszerzona dynamiczna nazwa pakietu	Nazwa rozszerzonego dynamicznego pakietu SQL.
82	52	CHAR(10)	Nazwa biblioteki pakietu	Nazwa biblioteki dla rozszerzonego dynamicznego pakietu SQL.
92	5C	Binary(2)	Indykator DRDA	<ul style="list-style-type: none"> • 0 - Podłączony do lokalnej bazy RDB • 1 - Podłączony do zdalnej bazy RDB
94	5E	CHAR(1)	Poziom kontroli transakcji	<ul style="list-style-type: none"> • 'A' - Zatwierdzenie *ALL • 'C' - Zatwierdzenie *CHANGE • 'N' - Zatwierdzenie *NONE • 'S' - Zatwierdzenie *CS (stabilność kursora - cursor stability) • 'L' - Zatwierdź *RR (odczyt powtarzalny)
95	5F	CHAR(10)	Domyślna kolekcja SQL	Nazwa domyślnej kolekcji SQL użytej przez serwer baz danych iSeries ^(TM) .
105	69	CHAR(129)	Zastrzeżone	Zastrzeżone dla przyszłych parametrów.
234	EA	BINARY(4)	Długość tekstu instrukcji SQL	Długość tekstu instrukcji SQL znajdującej się w następnym polu. Maksymalna długość może wynosić 64 K.
238	EE	CHAR(*)	Tekst instrukcji SQL	Kompletna instrukcja SQL.
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZDAEP w zbiorach H, QRPGRSRC, QRPGLSRC, QLBSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Punkt wyjścia QIBM_QZDA_ROI1 zdefiniowany jest tak, aby uruchamiał program obsługi wyjścia dla zgłoszeń pobierających informacje o obiektach dla serwera baz danych. Jest on także używany przy funkcjach katalogowania SQL.

Ten punkt wyjścia ma zdefiniowane dwa formaty. Formaty te opisane są poniżej.

Format ZDAR0100 używany jest przy zgłoszeniach pobierających informacje o następujących obiektach:

- Biblioteka (lub kolekcja)
- Zbiór (lub tabela)
- Pole (lub kolumna)
- Indeks
- Relacyjna baza danych (RDB)
- Pakiet SQL
- Instrukcja pakietu SQL
- Podzbiór zbioru
- Format zapisu
- Kolumny specjalne

Format ZDAR0200 używany jest podczas zgłoszeń pobierających informacje o następujących obiektach:

- Klucze obce

- Klucze podstawowe

Format ZDAR0100 punktu wyjścia QIBM_QZDA_ROI1

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku serwera baz danych wartością tą jest *RTVOBJINF.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla następujących funkcji nazwa formatu to ZDAR0100.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. Pole to zawiera jedną z następujących wartości: <ul style="list-style-type: none"> • X'00001800' - Pobranie informacji o bibliotece • X'00001801' - Pobranie informacji o relacyjnej bazie danych • X'00001802' - Pobranie informacji o pakiecie SQL • X'00001803' - Pobranie instrukcji pakietu SQL • X'00001804' - Pobranie informacji o zbiorze • X'00001805' - Pobranie informacji o podziorze zbioru • X'00001806' - Pobranie informacji o formacie rekordu • X'00001807' - Pobranie informacji o polu • X'00001808' - Pobranie informacji o indeksie • X'0000180B' - Pobranie informacji o kolumnie specjalnej
32	20	CHAR(20)	Nazwa biblioteki	Biblioteka lub wzorzec wyszukiwania użyty podczas pobierania informacji o bibliotekach, pakietach, instrukcjach pakietów, zbiorach, podziorach, formatach rekordu, polach, indeksach i kolumnach specjalnych.
52	34	CHAR(36)	Nazwa relacyjnej bazy danych	Nazwa relacyjnej bazy danych lub wzorzec wyszukiwania użyty podczas pobierania informacji o bazie RDB.
88	58	CHAR(20)	Nazwa pakietu	Nazwa pakietu lub wzorzec wyszukiwania użyty do pobierania informacji o pakiecie lub instrukcji pakietu.
108	6C	CHAR(256)	Nazwa zbioru (alias SQL)	Nazwa pliku lub wzorzec wyszukiwania użyty do pobrania informacji o zbiorze, podziorze, formacie rekordu, polu, indeksie lub kolumnie specjalnej.
364	16C	CHAR(20)	Nazwa podzioru	Podzior lub wzorzec wyszukiwania użyty do pobrania informacji o podziorze zbioru.
384	180	CHAR(20)	Nazwa formatu	Nazwa formatu lub wzorzec wyszukiwania użyty do pobrania informacji o formacie rekordu.
<p>Uwaga: Format ten zdefiniowany jest przez podzior EZDAEP w zbiorach H, QRPGRSRC, QRPGLSRC, QLBLSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Format ZDAR0200 punktu wyjścia QIBM_QZDA_ROI1

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku serwera baz danych wartością tą jest *RTVOBJINF.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla następujących funkcji nazwa formatu to ZDAR0200.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. Pole to zawiera jedną z następujących wartości: <ul style="list-style-type: none"> • X'00001809' - Pobranie informacji o kluczu obcym • X'0000180A' - Pobranie informacji o kluczu podstawowym
32	20	CHAR(10)	Nazwa biblioteki tabeli kluczy podstawowych	Nazwa biblioteki zawierającej tabelę kluczy podstawowych, która używana jest podczas pobierania informacji o kluczu podstawowym i obcym.
42	2A	CHAR(128)	Nazwa tabeli kluczy podstawowych (alias)	Nazwa tabeli zawierającej klucz podstawowy, która używana jest podczas pobierania informacji o kluczu podstawowym lub obcym.
170	AA	CHAR(10)	Nazwa biblioteki tabeli kluczy obcych	Nazwa biblioteki zawierającej tabelę kluczy obcych, która używana jest podczas pobierania informacji o kluczach obcych.
180	64	CHAR(128)	Nazwa tabeli kluczy obcych (alias)	Nazwa tabeli zawierającej klucz obcy, która używana jest podczas pobierania informacji o kluczu obcym.
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZDAEP w zbiorach H, QRPGRS, QRPGLSRC, QLBSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Serwer kolejek danych: Serwer kolejek danych ma jeden zdefiniowany punkt wyjścia:

Format ZHQ00100 punktu wyjścia QIBM_QZHQ_DATA_QUEUE

Punkt wyjścia QIBM_QZHQ_DATA_QUEUE jest definiowany, aby uruchamiał program punktu wyjścia, gdy odebrane zostaną następujące zgłoszenia serwera kolejek danych:

- Zapytanie
- Odbiór
- Tworzenie
- Usuwanie
- Wysyłanie
- Czyszczenie
- Anulowanie
- Podglądanie

Format ZHQ00100 punktu wyjścia QIBM_QZHQ_DATA_QUEUE

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	w przypadku serwera kolejek danych wartością tą jest *DATAQSRV.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. W przypadku QIBM_DATA_QUEUE nazwa formatu to ZHQ00100.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. <ul style="list-style-type: none"> • X'0001' - Zapytanie o atrybuty kolejki danych • X'0002' - Odbiór komunikatu z kolejki danych • X'0003' - Utworzenie kolejki danych • X'0004' - Usunięcie kolejki danych • X'0005' - Wysłanie komunikatu do kolejki danych • X'0006' - Wyczyszczenie komunikatu w kolejce danych • X'0007' - Anulowanie oczekującego zgłoszenia odbioru • X'0012' - Odbiór komunikatu z kolejki danych bez jego usuwania
32	20	CHAR(10)	Nazwa obiektu	Nazwa kolejki danych.
42	2A	CHAR(10)	Nazwa biblioteki	Biblioteka kolejki danych.
52	34	CHAR(2)	Operacja relacji	Operator relacji operacji odbieranych w zgłoszeniu przy pomocy klucza. <p>X'0000' - Brak operatora</p> <p>'EQ' - Równy</p> <p>'NE' - Różny</p> <p>'GE' - Większy lub równy</p> <p>'GT' - Większy niż</p> <p>'LE' - Mniejszy lub równy</p> <p>'LT' - Mniejszy niż</p>
54	36	BINARY(4)	Długość klucza	Długość klucza określona w zgłoszeniu.
58	3A	CHAR(256)	Wartość klucza	Wartość klucza określona w zgłoszeniu.
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZHQEP w zbiorach H, QRPGRS, QRPGLSRC, QLBSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Sieciowy serwer wydruków: Sieciowy serwer wydruków ma zdefiniowane dwa punkty wyjścia:

1. QIBM_QNPS_ENTRY o formacie ENTR0100
 - Wywoływany podczas inicjowania serwera
2. QIBM_QNPS_SPLF o formacie SPLF0100
 - Wywoływany, aby przetworzyć istniejący buforowy zbiór wyjściowy

Punkt wyjścia QIBM_QNPS_ENTRY jest zdefiniowany w taki sposób, aby uruchamiał program obsługi wyjścia, gdy uruchomiony jest sieciowy serwer wydruków. Program obsługi wyjścia może być użyty do weryfikacji praw dostępu do serwera. Więcej informacji znaleźć można w książce *Printer Device Programming*, SC41-5713-03.

Format ENTR0100 punktu wyjścia QIBM_QNPS_ENTRY

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku sieciowego serwera wydruków wartość ta wynosi QNPSERVER.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. W przypadku QIBM_QNPS_ENTRY nazwa formatu to ENTR0100.
28	1C	BINARY(4)	Identyfikator funkcji	Wykonywana funkcja. W przypadku QIBM_QNPS_ENTRY wartość ta wynosi X'0802'.
Uwaga: Format ten zdefiniowany jest przez podzbiór ENPSEP w zbiorach H, QRPGRSRC, QRPGLSRC, QLBLSRC oraz QCBLESRC w bibliotece QSYSINC.				

Punkt wyjścia QIBM_QNPS_SPLF jest zdefiniowany w taki sposób, aby uruchamiał program obsługi wyjścia, gdy sieciowy serwer wydruków odbierze zgłoszenie przetworzenia istniejącego buforowego zbioru wyjściowego. Program może być użyty do wykonywania funkcji na zbiorze buforowym, takich jak wysłanie zbioru faksem. Więcej informacji znaleźć można w książce *Printer Device Programming*, SC41-5713-03.

Format SPLF0100 punktu wyjścia QIBM_QNPS_SPLF

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku sieciowego serwera wydruków wartość ta wynosi QNPSERVER.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. W przypadku QIBM_QNPS_SPLF nazwa formatu to SPLF0100.
28	1C	BINARY(4)	Identyfikator funkcji	Wykonywana funkcja. Dla QIBM_QNPS_SPLF wartość ta wynosi X'010D'.
32	20	CHAR(10)	Nazwa zadania	Nazwa zadania, które utworzyło zbiór buforowy.
42	2A	CHAR(10)	Nazwa użytkownika	Nazwa profilu użytkownika zadania, które utworzyło zbiór buforowy.
52	34	CHAR(6)	Numer zadania	Numer zadania, które utworzyło zbiór buforowy.
58	3A	CHAR(10)	Nazwa zbioru buforowego	Nazwa żadanego zbioru buforowego.
68	44	BINARY(4)	Numer zbioru buforowego	Numer żadanego zbioru buforowego.
72	48	BINARY(4)	Długość	Długość danych programu obsługi wyjścia zbioru buforowego.

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
76	4C	CHAR(*)	Dane programu obsługi wyjścia zbioru buforowego	Dane programu obsługi wyjścia zbioru buforowego zawierają dodatkowe informacje wykorzystywane przez program obsługi wyjścia przyporządkowany do punktu wyjścia QIBM_QNPS_SPLF. Dane programu obsługi wyjścia zbioru buforowego udostępnia aplikacja kliencka.
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór ENPSEP w zbiorach H, QRPGRS, QRPGLSRC, QLBLSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Serwer centralny: Serwer centralny ma zdefiniowane trzy punkty wyjścia:

1. QIBM_QZSC_LM o formacie ZSCL0100
 - Wywoływany w przypadku zgłoszeń zarządzania licencjami
2. QIBM_QZSC_SM o formacie ZSCS0100
 - Wywoływany w przypadku zgłoszeń zarządzania systemem
3. QIBM_QZSC_NLS o formacie ZSCN0100
 - Wywoływany w przypadku zgłoszeń tabeli konwersji

Punkt wyjścia QIBM_QZSC_LM zdefiniowany jest tak, aby uruchamiał program obsługi wyjścia dla wszystkich zgłoszeń zarządzania licencjami odebranych przez serwer centralny.

Format ZSCL0100 programu obsługi wyjścia QIBM_QZSC_LM

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku serwera centralnego wartością tą jest *CNTRLSRV.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla QIBM_QZSC_LM nazwa formatu to ZSCL0100.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. Pole to zawiera jedną z następujących wartości: <ul style="list-style-type: none"> • X'1001' - Zgłoszenie licencji • X'1002' - Wydanie licencji • X'1003' - Pobranie informacji licencyjnych
32	20	CHAR(255)	Unikalna nazwa klienta	Unikalna nazwa klienta używana jest do identyfikacji określonej stacji roboczej w sieci. Możliwość użycia produktu licencjonowanego przypisywana jest do stacji roboczej identyfikowanej przez unikalną nazwę klienta.
287	11F	CHAR(8)	Uchwyt licencji użytkownika	Uchwyt licencji użytkownika używany jest do zapewnienia, że requester i releaser licencji są takie same. Wartość ta musi być taka sama jak wtedy, gdy żądano licencji.
295	127	CHAR(7)	Identyfikacja produktu	Identyfikacja produktu, dla którego żąda się licencji.
302	12E	CHAR(4)	Identyfikacja opcji	Opcja produktu.
306	132	CHAR(6)	Identyfikacja wydania	Wersja, wydanie i poziom modyfikacji produktu lub opcji

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
312	138	Binary(2)	Typ informacji	<p>Typ informacji, jaka ma być pobrana.</p> <p>Pole Typ informacji jest poprawne tylko w przypadku funkcji pobierania informacji licencyjnej.</p> <p>Pole to zawiera jedną z następujących wartości:</p> <ul style="list-style-type: none"> • X'0000' - Podstawowe informacje o licencji • X'0001' - Szczegółowe informacje o licencji
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZSCEP w zbiorach H, QRPGRSRC, QRPGLSRC, QLBLSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Punkt wyjścia QIBM_QZSC_SM jest zdefiniowany w taki sposób, aby uruchamiał program obsługi wyjścia dla wszystkich zgłoszeń zarządzania klientem odebranych przez serwer centralny.

Format ZSCS0100 programu obsługi wyjścia QIBM_QZSC_SM

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku serwera centralnego wartością tą jest *CNTRLSRV.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. W przypadku QIBM_QZSC_SM nazwa formatu to ZSCS0100.
28	1C	BINARY(4)	Żądana funkcja	<p>Wykonywana funkcja.</p> <p>Pole to zawiera jedną z następujących wartości:</p> <ul style="list-style-type: none"> • X'1101' - Klient aktywny • X'1102' - Klient nieaktywny
32	20	CHAR(255)	Unikalna nazwa klienta	Nazwa stacji roboczej klienta przyporządkowana jest do produktu licencjonowanego.
287	11F	CHAR(255)	Nazwa grupy	Pole nazwy grupy konfiguracji SNMP wykorzystywane jest w celach uwierzytelniania.
542	21E	CHAR(1)	Typ węzła	<p>Typ połączenia</p> <ul style="list-style-type: none"> • 3 - Internet
543	21F	CHAR(255)	Nazwa węzła	<p>Nazwa węzła.</p> <p>W przypadku węzła typu 3, nazwa będzie adresem internetowym.</p>
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZSCEP w zbiorach H, QRPGRSRC, QRPGLSRC, QLBLSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Punkt wyjścia QIBM_QZSC-NLS jest zdefiniowany w taki sposób, aby uruchamiał program obsługi wyjścia, gdy serwer centralny odbierze zgłoszenie pobrania mapy konwersji.

Format ZSCN0100 programu obsługi wyjścia QIBM_QZSC-NLS

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku serwera centralnego wartością tą jest *CNTRLSRV.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla QIBM_QZSC-NLS nazwa formatu to ZSCN0100.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. <ul style="list-style-type: none"> • X'1201' - Pobranie mapy konwersji
32	20	BINARY(4)	Z identyfikatora kodowanego zestawu znaków (CCSID)	Bieżący CCSID danych.
36	24	BINARY(4)	Do identyfikatora kodowanego zestawu znaków (CCSID)	CCSID, do którego dane będą przekształcane.
40	28	Binary(2)	Typ konwersji	Żądany typ odwzorowania. <ul style="list-style-type: none"> • X'0001' - Odwzorowanie przez obrót • X'0002' - Odwzorowanie przez podstawienie • X'0003' - Odwzorowanie typu Best-fit
<p>Uwaga: Format ten zdefiniowany jest przez podzbiór EZSCEP w zbiorach H, QRPGRSRC, QRPGLSRC, QLBLSRC oraz QCBLESRC w bibliotece QSYSINC.</p>				

Serwer komend zdalnych i rozproszonych wywołań programów: Serwer komend zdalnych i rozproszonych wywołań programów ma zdefiniowany jeden punkt wyjścia:

Format CZRC0100 punktu wyjścia QIBM_QZRC_RMT

Punkt wyjścia QIBM_QZRC_RMT zdefiniowany jest tak, aby wywoływał program dla zgłoszeń rozproszonego wywołania programu lub dla komendy zdalnej.

Format pól parametrów różni się w zależności od typu zgłoszenia.

Żądania komend zdalnych dla formatu CZRC0100 punktu wyjścia QIBM_QZRC_RMT

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku serwera komend zdalnych wartość ta wynosi *RMTSRV.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla QIBM_QZRC_RMT nazwa formatu to CZRC0100.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. <p style="text-align: center;">X'1002' - Komenda zdalna</p>
32	20	CHAR(10)	Zastrzeżone	Nie jest wykorzystywane w przypadku zgłoszeń komend zdalnych.

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
42	2A	CHAR(10)	Zastrzeżone	Nie jest wykorzystywane w przypadku zgłoszeń komend zdalnych.
52	34	BINARY(4)	Długość następnego pola	Długość następnego łańcucha komendy.
56	38	CHAR (6000)	Łańcuch komendy	Łańcuch komendy dla zgłoszeń komend zdalnych.

Żądania rozproszonych wywołań programów dla formatu CZRC0100 punktu wyjścia QIBM_QZRC_RMT

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika łączącego się z serwerem.
10	A	CHAR(10)	Identyfikator serwera	W przypadku serwera rozproszonych wywołań programu wartość ta wynosi *RMTSRV.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla QIBM_QZRC_RMT nazwa formatu to CZRC0100.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. X'1003' - Rozproszone wywołanie programu
32	20	CHAR(10)	Nazwa programu	Nazwa wywoływanego programu.
42	2A	CHAR(10)	Nazwa biblioteki	Biblioteka określonego programu.
52	34	BINARY(4)	Liczba parametrów	Całkowita liczba parametrów wywołania programu. Nie zawsze oznacza ona aktualną liczbę parametrów.
56	38	CHAR(*)	Informacja o parametrze	Informacja o parametrach przekazywanych do określonego programu. Wszystkie łańcuchy parametrów mają następujący format, niezależnie od typu używania parametru. Ostatnie pole w strukturze określa typy używania parametrów wejścia/wyjścia. <ul style="list-style-type: none"> • BINARY(4) - Długość informacji o parametrze dla danego parametru • BINARY(4) - Maksymalna długość parametru • BINARY(2) - Typ używania parametru <ul style="list-style-type: none"> – 1 - Wejście – 2 - Wyjście – 3 - Wejście/Wyjście • CHAR(*) - Łańcuch parametru

Serwer wpisywania się do systemu: Serwer wpisywania się do systemu ma zdefiniowany jeden punkt wyjścia:

Format ZSOY0100 punktu wyjścia QIBM_QZSO_SIGNONSRV

Punkt wyjścia QIBM_QZSO_SIGNONSRV_QUEUE definiowany jest po to, aby uruchamiał program punktu wyjścia, gdy odebrane zostaną następujące zgłoszenia serwera wpisywania się do systemu:

- Żądanie uruchomienia serwera
- Wczytanie informacji o wpisaniu się
- Zmiana hasła

- Generowanie znacznika uwierzytelniania
- Generowanie znacznika uwierzytelniania w imieniu innego użytkownika

Format ZSOY0100 punktu wyjścia QIBM_QZSO_SIGNONSRV

Przesunięcie		Typ	Pole	Opis
Dzies.	Szest.			
0	0	CHAR(10)	Nazwa profilu użytkownika	Nazwa profilu użytkownika powiązanego ze zgłoszeniem.
10	A	CHAR(10)	Identyfikator serwera	Dla serwera wpisywania się do systemu wartość ta wynosi *SIGNON.
20	14	CHAR(8)	Nazwa formatu	Nazwa używanego formatu obsługi wyjścia użytkownika. Dla QIBM_QZSO_SIGNONSRV nazwa formatu to ZSOY0100.
28	1C	BINARY(4)	Żądana funkcja	Wykonywana funkcja. <ul style="list-style-type: none"> • X'7002' - Żądanie uruchomienia serwera • X'7004' - Wczytanie informacji o wpisaniu się • X'7005' - Zmiana hasła • X'7007' - Generowanie znacznika uwierzytelniania • X'7008' - Generowanie znacznika uwierzytelniania w imieniu innego użytkownika

Przykłady: Programy obsługi wyjścia

Przykładowe programy obsługi wyjścia w tej sekcji nie ilustrują wszystkich możliwych technik i zagadnień programowania, ale można je przejrzeć zanim zaczniesz się je tworzyć samemu.

Informacje dotyczące kodu

IBM^(R) udziela niewyłącznej licencji na prawa autorskie, stosowanej przy używaniu wszelkich przykładowych kodów programów, na podstawie których można wygenerować podobne funkcje dostosowane do indywidualnych wymagań.

Cały kod przykładowy jest udostępniany przez IBM jedynie do celów ilustracyjnych. Programy przykładowe nie zostały gruntownie przetestowane. IBM nie może zatem gwarantować lub sugerować niezawodności, użyteczności i funkcjonalności tych programów.

Wszelkie zawarte tutaj programy są dostarczane w stanie, w jakim się znajdują ("AS IS") bez udzielania jakichkolwiek gwarancji. Nie udziela się domniemanych gwarancji nienaruszania praw osób trzecich, gwarancji przydatności handlowej oraz przydatności do określonego celu.

- Przykłady: Tworzenie programów obsługi wyjścia w języku RPG
- Przykłady: Tworzenie programów obsługi wyjścia w języku CL

Przykłady: Tworzenie programów obsługi wyjścia w języku RPG: Poniższy przykład pokazuje, w jaki sposób można napisać program obsługi wyjścia użytkownika w języku RPG*.

Uwaga: Należy przeczytać sekcję Informacje dotyczące kodu, która zawiera istotne informacje prawne.

```

**
** OS/400 SERVERS - PRZYKŁADOWY PROGRAM OBSŁUGI WYJŚCIA UŻYTKOWNIKA
**
** PONIŻSZY PROGRAM RPG BEZWARUNKOWO AKCEPTUJE WSZYSTKIE
** ZGŁOSZENIA. MOŻE ON BYĆ UŻYTY JAKO PODSTAWA DO TWORZENIA
** OKREŚLONYCH APLIKACJI. UWAGA: ABY UZYSKAĆ WIĘKSZĄ WYDAJNOŚĆ,

```

** NALEŻY USUNĄĆ WSZYSTKIE PODPROGRAMY I POZYCJE INSTRUKCJI
 ** DLA SERWERA, KTÓRE NIE POTRZEBUJĄ SPECJALNEGO
 ** PROGRAMU OBSŁUGI WYJŚCIA.
 **

E*
 E* KONIECZNA DEFINICJA TABLICY DLA FUNKCJI PRZESYŁANIA
 E* I ZDALNEGO SQL

E*
 E TFREQ 4096 1
 E RSREQ 4107 1

I*
 I*
 IPCSDTA DS
 I 1 10 USERID
 I 11 20 APPLID

I* PARAMETRY DRUKARKI WIRTUALNEJ

I*
 I 21 30 VPFUNC
 I 31 40 VPOBJ
 I 41 50 VPLIB
 I 71 750VPIFN
 I 76 85 VPOUTQ
 I 86 95 VPQLIB

I* PARAMETRY FUNKCJI PRZESYŁANIA KOMUNIKATÓW

I 21 30 MFFUNC

I* PARAMETRY FUNKCJI PRZESYŁANIA

I*
 I 21 30 TFFUNC
 I 31 40 TFOBJ
 I 41 50 TFLIB
 I 51 60 TFMBR
 I 61 70 TFFMT
 I 71 750TFLEN
 I 764171 TFREQ

I* PARAMETRY SERWERA PLIKÓW

I*
 I* UWAGA: FSNAME MOŻE WYNOŚĆ AŻ 16 MB.
 I* FSNLEN ZAWIERA RZECZYWISTĄ WIELKOŚĆ FSNAME.

I*
 I B 21 240FSFID
 I 25 32 FSFMT
 I 33 33 FSREAD
 I 34 34 FSWRIT
 I 35 35 FSRDWR
 I 36 36 FSDLT
 I B 37 400FSNLEN
 I 41 296 FSNAME

I* PARAMETRY KOLEJEK DANYCH

I*
 I 21 30 DQFUNC
 I 31 40 DQQ
 I 41 50 DQLIB
 I 70 750DQLEN
 I 76 77 DQROP
 I 78 820DQKLEN
 I 83 338 DQKEY

I* PARAMETRY ZDALNEGO SQL

I*
 I 21 30 RSFUNC
 I 31 40 RSOBJ
 I 41 50 RSLIB

```

I          51 51 RSCMT
I          52 52 RSMODE
I          53 53 RSCID
I          54 71 RSSTN
I          72 75 RSRSV
I          764182 RSREQ
I*
I* PARAMETRY SIECIOWEGO SERWERA WYDRUKÓW
I*
I          21 28 NPFT
I          B 29 320NPFID
I* DODATKOWE PARAMETRY DLA FORMATU SPLF0100
I          33 42 NPJOB#
I          43 52 NPUSR#
I          53 58 NPJOB#
I          59 68 NPFILE
I          B 69 720NPFIL#
I          B 73 760NPLEN
I          77 332 NPDATA
I*
I* Serwer kolejki danych:
I*
I* Format ZHQ00100 dla QIBM_QZHQ_DATA_QUEUE
I*
I          21 28 DQOFMT
I          B 29 320DQOFID
I          33 42 DQO0BJ
I          43 52 DQOLIB
I          53 54 DQOROP
I          B 55 580DQOLEN
I          59 314 DQOKEY
I*
I* PARAMETRY SERWERA CENTRALNEGO
I*
I          21 28 CSFMT
I          B 29 320CSFID
I* Serwer centralny:
I*
I* QIBM_QZSC_LM format ZSCL0100 wywołań zarządzania licencją
I*
I*
I          33 287 CSLCNM
I          288 295 CSLUSR
I          296 302 CSLPID
I          303 306 CSLFID
I          307 312 CSLRID
I          B 313 3140CSLTYP
I*
I* Serwer centralny:
I*
I* QIBM_QZSC_LM format ZSCS0100 wywołań zarządzania systemem
I*
I*
I          33 287 CSSCNM
I          288 542 CSSCMY
I          543 543 CSSNDE
I          544 798 CSSNNM
I*
I* Serwer centralny:
I*
I* QIBM_QZSC_LM format ZSCN0100 wywołań pobierania mapy konwersji
I*
I*
I          21 30 CSNXFM
I          29 320CSNFNC
I          B 33 360CSNFRM

```

```

I          B 37 400CSNTO
I          B 41 420CSNCNT
I*
I* PARAMETRY SERWERA BAZ DANYCH
I*
I          21 28 DBFMT
I          B 29 320DBFID
I*
I* DODATKOWE PARAMETRY DLA FORMATU ZDAD0100
I          33 160 DBDFIL
I          161 170 DBDLIB
I          171 180 DBDMBR
I          181 190 DBDAUT
I          191 318 DBDBFL
I          319 328 DBDBLB
I          329 338 DBDOFL
I          339 348 DBDOLB
I          349 358 DBDOMB
I*
I* DODATKOWE PARAMETRY DLA FORMATU ZDAD0200
I          B 33 360DBNUM
I          37 46 DBLIB2
I*
I* DODATKOWE PARAMETRY DLA FORMATU ZDAQ0100
I          33 50 DBSTMT
I          51 68 DBCRSR
I          69 70 DBOPI
I          71 72 DBATTR
I          73 82 DBPKG
I          83 92 DBPLIB
I          B 93 940DBDRDA
I          95 95 DBCMT
I          96 351 DBTEXT
I* PARAMETRY WYMIENIAJĄCE DBTEXT DLA FORMATU ZDAQ0200
I          96 105 DBSQCL
I          B 133 1360DBSQLN
I          137 392 DBSQTX
I* DODATKOWE PARAMETRY DLA FORMATU ZDAR0100
I          33 52 DBLIBR
I          53 88 DBRDBN
I          89 108 DBPKGR
I          109 364 DBFILR
I          365 384 DBMBRR
I          385 404 DBFFT
I* DODATKOWE PARAMETRY DLA FORMATU ZDAR0200
I          33 42 DBRPLB
I          43 170 DBRPTB
I          171 180 DBRFLB
I          181 308 DBRFTB
I*
I* Serwer komend zdalnych i rozproszonych wywołań programów:
I*
I* Format CZRC0100 dla QIBM_QZRC_RMT
I* RCPGM I RCLIB NIE SĄ WYKORZYSTYWANE PODCZAS WYWOŁAŃ KOMEND ZDALNYCH
I*
I          21 28 RCFMT
I          B 29 320RCFID
I          33 42 RCPGM
I          43 52 RCLIB
I          B 53 560RCNUM
I          57 312 RCDATA
I*
I* Serwer wpisywania się:
I*
I* Format ZSOY0100 punktu wyjścia QIBM_QZSO_SIGNONSRV dla serwera TCP/IP wpisywania się
I*
I          21 28 SOXFMT

```

```

I                                     B 29 320SOFID
I*
I*****
I*
I          '*VPRT      '          C          #VPRT
I          '*TFRFCL   '          C          #TRFCL
I          '*FILESRV  '          C          #FILE
I          '*MSGFCL   '          C          #MSGF
I          '*DQSRV   '          C          #DQSRV
I          '*RQSRV   '          C          #RQSRV
I          '*SQL     '          C          #SQL
I          '*NDB     '          C          #NDBSV
I          '*SQLSRV  '          C          #SQLSV
I          '*RTVOBJINF'          C          #RTVOB
I          '*DATAQSRV'          C          #DATAQ
I          '*QNPSERV  '          C          #QNPSV
I          '*CNTRLSRV'          C          #CNTRL
I          '*RMTSRV  '          C          #RMTSV
I          '*SIGNON  '          C          #SIGN
I*
C*
C* PARAMETRY WYWOŁANIA PROGRAMU OBSŁUGI WYJŚCIA
C*
C          *ENTRY   PLIST
C                   PARM          RTNCD  1
C                   PARM          PCSDTA
C*
C* INICJOWANIE WARTOŚCI ZWRACANEJ DLA AKCEPTOWANIA ZGŁOSZEŃ
C*
C                   MOVE '1'      RTNCD
C*
C* WSPÓLNE PRZETWARZANIE
C*
C*           TU ZNAJDUJE SIĘ LOGIKA WSPÓLNA
C*
C* PRZETWARZANIE W OPARCIU O ID SERWERA
C*
C          APPLID   CASEQ#VPRT      VPRT
C          APPLID   CASEQ#TRFCL     TFR
C          APPLID   CASEQ#FILE      FILE
C          APPLID   CASEQ#MSGF      MSG
C          APPLID   CASEQ#DQSRV     DATAQ
C          APPLID   CASEQ#RQSRV     RSQL
C          APPLID   CASEQ#SQL       SQLINT
C          APPLID   CASEQ#NDBSV     NDB
C          APPLID   CASEQ#SQLSV     SQLSRV
C          APPLID   CASEQ#RTVOB     RTVOBJ
C          APPLID   CASEQ#DATAQ     ODATAQ
C          APPLID   CASEQ#QNPSV     NETPRT
C          APPLID   CASEQ#CNTRL     CENTRL
C          APPLID   CASEQ#RMTSV     RMTCMD
C          APPLID   CASEQ#SIGN     SIGNON
C          END
C          SETON          LR
C          RETRN
C*
C* PODPROGRAMY
C*
C*
C* DRUK WIRTUALNY
C*
C          VPRT      BEGSR
C*           TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C          ENDSR
C*
C* FUNKCJA PRZESYŁANIA

```

```

C*
C* PONIŻEJ PRZEDSTAWIONY JEST PRZYKŁAD KONKRETNEGO PRZETWARZANIA,
C* JAKIE PROGRAM OBSŁUGI WYJŚCIA WYKONAĆ MOŻE DLA FUNKCJI PRZESYŁANIA.
C*
C* W TYM WYPADKU UŻYTKOWNICY NIE MAJĄ ZEZWOLENIA NA
C* ZAZNACZANIE DANYCH W PLIKACH ZNAJDUJĄCYCH SIĘ W BIBLIOTECE QIWS.
C*
C          TFR          BEGSR
C          TFFUNC       IFEQ 'SELECT'
C          TFLIB        ANDEQ'QIWS'
C                   MOVE '0'          RTNCD
C                   END
C                   ENDSR
C*
C*
C* SERWER PLIKÓW
C*
C          FILE          BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C                   ENDSR
C*
C* FUNKCJA PRZESYŁANIA KOMUNIKATÓW
C*
C          MSG          BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C                   ENDSR
C* KOLEJKI DANYCH
C*
C          DATAQ       BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C                   ENDSR
C*
C* ZDALNE SQL
C*
C          RSQL         BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C                   ENDSR
C*
C* SERWERY
C*
C*
C* INICJOWANIE BAZY DANYCH
C*
C          SQLINT       BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C                   ENDSR
C*
C* RODZIMA BAZA DANYCH
C*
C          NDB          BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C                   ENDSR
C*
C* BAZA DANYCH SQL
C*
C          SQLSRV       BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C                   ENDSR
C*
C* INFORMACJA O OBIEKCIE POBIERANYM Z BAZY DANYCH
C*
C          RTVOBJ       BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C                   ENDSR
C*
C* SERWER KOLEJKI DANYCH
C*

```

```

C          ODATAQ   BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C          ENDSR
C*
C* SERWER WYDRUKÓW SIECIOWYCH
C*
C          NETPRT   BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C          ENDSR
C*
C* SERWER CENTRALNY
C*
C*
C* PONIŻEJ PRZEDSTAWIONY JEST PRZYKŁAD KONKRETNEGO PRZETWARZANIA,
C* JAKIE PROGRAM OBSŁUGI WYJŚCIA WYKONAĆ MOŻE DLA ZARZĄDZANIA LICENCJĄ.
C*
C* W TYM WYPADKU UŻYTKOWNIK "USERALL" NIE BĘDZIE MIAŁ ZEZWOLENIA NA
C* WYKONYWANIE JAKICHKOLWIEK FUNKCJI UDOSTĘPNIANYCH PRZEZ SERWER
C* CENTRALNY, DLA KTÓRYCH PROGRAM TEN JEST ZAREJESTROWANYM PROGRAMEM
C* OBSŁUGI WYJŚCIA - INFORMACJI O LICENCJI, ZARZĄDZANIU SYSTEMEM LUB
C* POBRANIEM MAPY KONWERSJI.
C*
C          CENTRL   BEGSR
C          USERID   IFEQ 'USERALL'
C                   MOVE '0'          RTNCD
C                   ENDIF
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C          ENDSR
C*
C* KOMENDA ZDALNA I ROZPROSZONE WYWOŁANIE PROGRAMU
C*
C* W TYM WYPADKU UŻYTKOWNIK "USERALL" NIE BĘDZIE MIAŁ ZEZWOLENIA NA
C* WYKONYWANIE JAKICHKOLWIEK KOMEND ZDALNYCH LUB ROZPROSZONYCH WYWOŁAŃ
C* PROGRAMU.
C*
C          RMTCMD   BEGSR
C          USERID   IFEQ 'USERALL'
C                   MOVE '0'          RTNCD
C                   ENDIF
C          ENDSR
C*
C* SERWER WPISYWANIA SIĘ
C*
C          SIGNON   BEGSR
C*          TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA
C          ENDSR

```

Przykłady: Tworzenie programów obsługi wyjścia w języku CL: Podany przykład pokazuje, w jaki sposób można napisać program obsługi wyjścia użytkownika w języku CL.

Uwaga: Należy przeczytać sekcję Informacje dotyczące kodu, która zawiera istotne informacje prawne.

```

/*****
/*
/* SERWERY iSeries- PRZYKŁADOWY PROGRAM OBSŁ. WYJŚCIA UŻYTKOWNIKA */
/*
/* PONIŻSZY PROGRAM W JĘZYKU CL BEZWARUNKOWO AKCEPTUJE WSZELKIE */
/* ZGŁOSZENIA. MOŻE BYĆ ON UŻYTY JAKO PODSTAWA DO TWORZENIA */
/* PROGRAMÓW OBSŁUGI WYJŚCIA DOSTOSOWANYCH DO ŚRODOWISKA SYSTEMU. */
/*
/*
/*
/*****

```

PGM PARM(&STATUS &ZGŁOSZENIA)

```

/* * * * * * * * * * * * * * * * * * * * * */
/*

```

```

/*      DEKLARACJE PARAMETRÓW      */
/*      WYWOŁANIA PROGRAMU      */
/*      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */
DCL VAR(&STATUS) TYPE(*CHAR) LEN(1) /* Indykator akceptacji/odrzućenia */
DCL VAR(&REQUEST) TYPE(*CHAR) LEN(9999) /* Struktura parametru. LEN(9999) CL */

/*****/
/*      */
/* DEKLARACJE PARAMETRÓW      */
/*      */
/*****/

/* WSPÓLNE DEKLARACJE */
DCL VAR(&USER) TYPE(*CHAR) LEN(10)
/* ID u¿ytkownika      */
DCL VAR(&APPLIC) TYPE(*CHAR) LEN(10)
/* ID serwera      */
DCL VAR(&FUNCTN) TYPE(*CHAR) LEN(10) /* Wykonywana funkcja */

/* DEKLARACJE DRUKU WIRTUALNEGO */
DCL VAR(&VPOBJ) TYPE(*CHAR) LEN(10) /* Nazwa obiektu */
DCL VAR(&VPLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki obiektu */
DCL VAR(&VPLEN) TYPE(*DEC) LEN(5 0) /* Długość kolejnych pól */
DCL VAR(&VPOUTQ) TYPE(*CHAR) LEN(10) /* Nazwa kolejki wyjściowej */
DCL VAR(&VPQLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki kolejki wyjściowej */

/* DEKLARACJE FUNKCJI PRZESYŁANIA */
DCL VAR(&TFOBJ) TYPE(*CHAR) LEN(10) /* Nazwa obiektu */
DCL VAR(&TFLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki obiektu */
DCL VAR(&TFMBR) TYPE(*CHAR) LEN(10) /* Nazwa podzbioru */
DCL VAR(&TFMT) TYPE(*CHAR) LEN(10) /* Nazwa formatu rekordu */
DCL VAR(&TFLEN) TYPE(*DEC) LEN(5 0) /* Długość zgłoszenia */
DCL VAR(&TFREQ) TYPE(*CHAR) LEN(1925) /* Instrukcja żądania transferu */

/* DEKLARACJE SERWERA PLIKÓW */
DCL VAR(&FSFID) TYPE(*CHAR) LEN(4) /* Identyfikator funkcji */
DCL VAR(&FSFMT) TYPE(*CHAR) LEN(8) /* Format parametru */
DCL VAR(&FSREAD) TYPE(*CHAR) LEN(1) /* Otwarty do odczytu */
DCL VAR(&FSWRITE) TYPE(*CHAR) LEN(1) /* Otwarty do zapisu */
DCL VAR(&FSRDWRT) TYPE(*CHAR) LEN(1) /* Otwarty do zapisu/odczytu */
DCL VAR(&FSDLT) TYPE(*CHAR) LEN(1) /* Otwarty do usuwania */
DCL VAR(&FSLEN) TYPE(*CHAR) LEN(4) /* Długość nazwy pliku */
DCL VAR(&FSNAME) TYPE(*CHAR) LEN(2000) /* Kwalifikowana nazwa pliku */

/* DEKLARACJE KOLEJKI DANYCH */
DCL VAR(&DQQ) TYPE(*CHAR) LEN(10) /* Nazwa kolejki danych */
DCL VAR(&DQLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki kolejki danych */
DCL VAR(&DQLEN) TYPE(*DEC) LEN(5 0) /* Całkowita długość zgłoszenia */
DCL VAR(&DQROP) TYPE(*CHAR) LEN(2) /* Operator relacji */
DCL VAR(&DQKLEN) TYPE(*DEC) LEN(5 0) /* Długość klucza */
DCL VAR(&DQKEY) TYPE(*CHAR) LEN(256) /* Wartość klucza */

/* DEKLARACJE ZDALNEGO SQL */
DCL VAR(&RSOBJ) TYPE(*CHAR) LEN(10) /* Nazwa obiektu */
DCL VAR(&RSLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki obiektu */
DCL VAR(&RSCMT) TYPE(*CHAR) LEN(1) /* Poziom kontroli transakcji */
DCL VAR(&RSMODE) TYPE(*CHAR) LEN(1) /* Indykator trybu blok/aktualizacja */
DCL VAR(&RSCID) TYPE(*CHAR) LEN(1) /* ID kursora */

```



```

DCL VAR(&RSTN) TYPE(*CHAR) LEN(18) /* Nazwa instrukcji */
DCL VAR(&RSRSU) TYPE(*CHAR) LEN(4) /* Zarezerwowane */
DCL VAR(&RSREQ) TYPE(*CHAR) LEN(1925)/* Instrukcja SQL */

/* DEKLARACJE SIECIOWEGO SERWERA WYDRUKÓW */
DCL VAR(&NPFMT) TYPE(*CHAR) LEN(8) /* Nazwa formatu */
DCL VAR(&NPFID) TYPE(*CHAR) LEN(4) /* Identyfikator funkcji */
/* DODATKOWE PARAMETRY DLA FORMATU SPLF0100 */
DCL VAR(&NPJOBN) TYPE(*CHAR) LEN(10) /* Nazwa zadania */
DCL VAR(&NPUSRN) TYPE(*CHAR) LEN(10) /* Nazwa użytkownika */
DCL VAR(&NPJOB#) TYPE(*CHAR) LEN(6) /* Nazwa zadania */
DCL VAR(&NPFILE) TYPE(*CHAR) LEN(10) /* Nazwa zbioru */
DCL VAR(&NPFIL#) TYPE(*CHAR) LEN(4) /* Numer zbioru */
DCL VAR(&NPLEN) TYPE(*CHAR) LEN(4) /* Długość danych */
DCL VAR(&NPDATA) TYPE(*CHAR) LEN(2000) /* Data */

DCL VAR(&DBNUM) TYPE(*CHAR) LEN(4) /* Liczba bibliotek */
DCL VAR(&DBLIB2) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki

/* DEKLARACJE SERWERA KOLEJEK DANYCH */
DCL VAR(&DQFMT) TYPE(*CHAR) LEN(8) /* Nazwa formatu */
DCL VAR(&DQFID) TYPE(*CHAR) LEN(4) /* IDENTYFIKATOR funkcji */
DCL VAR(&DQOOBJ) TYPE(*CHAR) LEN(10) /* Nazwa obiektu */
DCL VAR(&DQOLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki */
DCL VAR(&DQOROP) TYPE(*CHAR) LEN(2) /* Operator relacji */
DCL VAR(&DQOLEN) TYPE(*CHAR) LEN(4) /* Długość klucza */
DCL VAR(&DQOKEY) TYPE(*CHAR) LEN(256) /* Klucz

/* DEKLARACJE SERWERA CENTRALNEGO */
DCL VAR(&CSFMT) TYPE(*CHAR) LEN(8) /* Nazwa formatu */
DCL VAR(&CSFID) TYPE(*CHAR) LEN(4) /* Identyfikator funkcji */
/* DODATKOWE PARAMETRY DLA FORMATU ZSCL0100 */
DCL VAR(&CSCNAM) TYPE(*CHAR) LEN(255) /* Unikalna nazwa klienta */
DCL VAR(&CSLUSR) TYPE(*CHAR) LEN(8) /* Uchwyt licencji użytkownika */
DCL VAR(&CSPID) TYPE(*CHAR) LEN(7) /* Identyfikator produktu */
DCL VAR(&CSFID) TYPE(*CHAR) LEN(4) /* Identyfikator opcji */
DCL VAR(&CSRID) TYPE(*CHAR) LEN(6) /* Identyfikator wydania */
DCL VAR(&CSTYPE) TYPE(*CHAR) LEN(2) /* Typ żądanej informacji */
/* DODATKOWE PARAMETRY DLA FORMATU ZSCS0100 */
DCL VAR(&CSCNAM) TYPE(*CHAR) LEN(255) /* Unikalna nazwa klienta */
DCL VAR(&CSCMTY) TYPE(*CHAR) LEN(255) /* Nazwa grupy */
DCL VAR(&CSNODE) TYPE(*CHAR) LEN(1) /* Typ węzła */
DCL VAR(&CSNNAM) TYPE(*CHAR) LEN(255) /* Nazwa węzła */
/* DODATKOWE PARAMETRY DLA FORMATU ZSCN0100 */
DCL VAR(&CSFROM) TYPE(*CHAR) LEN(4) /* Źródłowy CCSID */
DCL VAR(&CSTO) TYPE(*CHAR) LEN(4) /* Docelowy CCSID */
DCL VAR(&CSCTYP) TYPE(*CHAR) LEN(2) /* Typ konwersji */
/* DEKLARACJE SERWERA BAZ DANYCH */
DCL VAR(&DBFMT) TYPE(*CHAR) LEN(8) /* Nazwa formatu */
DCL VAR(&DBFID) TYPE(*CHAR) LEN(4) /* Identyfikator opcji

/* DODATKOWE PARAMETRY DLA FORMATU ZDAD0100 */
DCL VAR(&DBFILE) TYPE(*CHAR) LEN(128) /* Nazwa zbioru */
DCL VAR(&DBLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki */
DCL VAR(&DBMBR) TYPE(*CHAR) LEN(10) /* Nazwa podzbioru */
DCL VAR(&DBAUT) TYPE(*CHAR) LEN(10) /* Uprawnienia do zbioru */
DCL VAR(&DBBFIL) TYPE(*CHAR) LEN(128) /* Nazwa zbioru podstawowego */
DCL VAR(&DBBLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki podstawowej */
DCL VAR(&DBOFIL) TYPE(*CHAR) LEN(10) /* Nazwa zbioru nadpisującego */

```

```

DCL VAR(&DBOLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki nadpisującej */
DCL VAR(&DBOMBR) TYPE(*CHAR) LEN(10) /* Nazwa podzbioru nadpisującego */

/* DODATKOWE PARAMETRY DLA FORMATU ZDAD0200 */
DCL VAR(&DBNUM) TYPE(*CHAR) LEN(4) /* Liczba bibliotek */
DCL VAR(&DBLIB2) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki

/* DODATKOWE PARAMETRY DLA FORMATU ZDAQ0100 */
DCL VAR(&DBSTMT) TYPE(*CHAR) LEN(18) /* Nazwa instrukcji */
DCL VAR(&DBCRRS) TYPE(*CHAR) LEN(18) /* Nazwa kursora */
DCL VAR(&DBOPT) TYPE(*CHAR) LEN(2) /* Opcja przygotowania */
DCL VAR(&DBATTR) TYPE(*CHAR) LEN(2) /* Atrybuty otwarcia */
DCL VAR(&DBPKG) TYPE(*CHAR) LEN(10) /* Nazwa pakietu */
DCL VAR(&DBPLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki pakietu */
DCL VAR(&DBDRDA) TYPE(*CHAR) LEN(2) /* Indykator DRDA(R) */
DCL VAR(&DBCMT) TYPE(*CHAR) LEN(1) /* Poziom kontroli transakcji */
DCL VAR(&DBTEXT) TYPE(*CHAR) LEN(512) /* Pierwsze 512 bajtów instrukcji

/* DODATKOWE PARAMETRY DLA FORMATU ZDAR0100 */
DCL VAR(&DBLIBR) TYPE(*CHAR) LEN(20) /* Nazwa biblioteki */
DCL VAR(&DBRDBN) TYPE(*CHAR) LEN(36) /* Relacyjna baza danych */
DCL VAR(&DBPKGR) TYPE(*CHAR) LEN(20) /* Nazwa pakietu */
DCL VAR(&DBFILR) TYPE(*CHAR) LEN(256) /* Nazwa zbioru (alias SQL) */
DCL VAR(&DBMBRR) TYPE(*CHAR) LEN(20) /* Nazwa podzbioru */
DCL VAR(&DBFFMT) TYPE(*CHAR) LEN(20) /* Nazwa formatu

/* DODATKOWE PARAMETRY DLA FORMATU ZDAR0200 */
DCL VAR(&DBPLIB) TYPE(*CHAR) LEN(10) /* Biblioteka tabeli kluczy pierwotnych */
DCL VAR(&DBPTBL) TYPE(*CHAR) LEN(128) /* Tabela kluczy pierwotnych */
DCL VAR(&DBFLIB) TYPE(*CHAR) LEN(10) /* Biblioteka tabeli kluczy obcych */
DCL VAR(&DBFTBL) TYPE(*CHAR) LEN(128) /* Tabela kluczy obcych

/* DEKLARACJE SERWERA KOMEND ZDALNYCH */
DCL VAR(&RCFMT) TYPE(*CHAR) LEN(8) /* Nazwa formatu */
DCL VAR(&RCFID) TYPE(*CHAR) LEN(4) /* Identyfikator funkcji */
DCL VAR(&RCPGM) TYPE(*CHAR) LEN(10) /* Nazwa programu */
DCL VAR(&RCLIB) TYPE(*CHAR) LEN(10) /* Nazwa biblioteki programów */
DCL VAR(&RCNUM) TYPE(*CHAR) LEN(4) /* Liczba parametrów wiersza komend

DCL VAR(&RCDATA) TYPE(*CHAR) LEN(9999) /* Liczba parametrów łańcucha komendy

/* DEKLARACJE SERWERA WPISYWANIA SIĘ DO SYSTEMU */

DCL VAR(&SOFMT) TYPE(*CHAR) LEN(8) /* Nazwa formatu */
DCL VAR(&SOFID) TYPE(*CHAR) LEN(4) /* Identyfikator funkcji

/*****/
/*
/* INNE DEKLARACJE
/*
/*****/
DCL VAR(&WRKLEN) TYPE(*CHAR) LEN(5)
DCL VAR(&DECLLEN) TYPE(*DEC) LEN(8 0)

/* * * * * *
/*

```

```

/* PARAMETRY WYODRĘBNIONE ZE STRUKTURY          */
/*
/* * * * * * * * * * * * * * * * * * * * * * * */
/* NAGŁÓWEK */
CHGVAR VAR(&USER) VALUE(%SST(&REQUEST 1 10))
CHGVAR VAR(&APPLIC) VALUE(%SST(&REQUEST 11 10))
CHGVAR VAR(&FUNCTN) VALUE(%SST(&REQUEST 21 10))

/* DRUKARKA WIRTUALNA */
CHGVAR VAR(&VPOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&VPLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&VPLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&VPOUTQ) VALUE(%SST(&REQUEST 76 10))
CHGVAR VAR(&VPQLIB) VALUE(%SST(&REQUEST 86 10))

/* FUNKCJA PRZESYŁANIA */
CHGVAR VAR(&TFOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&TFLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&TFMBR) VALUE(%SST(&REQUEST 51 10))
CHGVAR VAR(&TFMT) VALUE(%SST(&REQUEST 61 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&TFLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&TFREQ) VALUE(%SST(&REQUEST 76 1925))

/* SERWER PLIKÓW */
CHGVAR VAR(&FSFID) VALUE(%SST(&REQUEST 21 4))
CHGVAR VAR(&FSFMT) VALUE(%SST(&REQUEST 25 8))
CHGVAR VAR(&FSREAD) VALUE(%SST(&REQUEST 33 1))
CHGVAR VAR(&FSWRITE) VALUE(%SST(&REQUEST 34 1))
CHGVAR VAR(&FSRDWRT) VALUE(%SST(&REQUEST 35 1))
CHGVAR VAR(&FSDLT) VALUE(%SST(&REQUEST 36 1))
CHGVAR VAR(&FSLEN) VALUE(%SST(&REQUEST 37 4))
CHGVAR VAR(&DECLEN) VALUE(%BINARY(&FSLEN 1 4))
CHGVAR VAR(&FSNAME) VALUE(%SST(&REQUEST 41 &DECLEN))

/* KOLEJKI DANYCH */
CHGVAR VAR(&DQO) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&DQLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 71 5))
CHGVAR VAR(&DQLEN) VALUE(%BINARY(&WRKLEN 1 4))
CHGVAR VAR(&DQROP) VALUE(%SST(&REQUEST 76 2))
CHGVAR VAR(&WRKLEN) VALUE(%SST(&REQUEST 78 5))
CHGVAR VAR(&DQKLEN) VALUE(&WRKLEN)
CHGVAR VAR(&DQKEY) VALUE(%SST(&REQUEST 83 &DQKLEN))

/* ZDALNE SQL */
CHGVAR VAR(&RSOBJ) VALUE(%SST(&REQUEST 31 10))
CHGVAR VAR(&RSLIB) VALUE(%SST(&REQUEST 41 10))
CHGVAR VAR(&RSCMT) VALUE(%SST(&REQUEST 51 1))
CHGVAR VAR(&RSMODE) VALUE(%SST(&REQUEST 52 1))
CHGVAR VAR(&RSCID) VALUE(%SST(&REQUEST 53 1))
CHGVAR VAR(&RSSSTN) VALUE(%SST(&REQUEST 54 18))
CHGVAR VAR(&RSRSU) VALUE(%SST(&REQUEST 72 4))
CHGVAR VAR(&RSREQ) VALUE(%SST(&REQUEST 76 1925))

```

```

/* SIECIOWY SERWER WYDRUKÓW */
  CHGVAR VAR(&NPFMT)    VALUE(%SST(&REQUEST 21 8))
  CHGVAR VAR(&NPFID)    VALUE(%SST(&REQUEST 29 4))

/* JEŚLI FORMAT JEST SPLF0100 */
IF COND(&NPFMT *EQ 'SPLF0100') THEN(DO)
  CHGVAR VAR(&NPJOB#)   VALUE(%SST(&REQUEST 33 10))
  CHGVAR VAR(&NPUSR#)   VALUE(%SST(&REQUEST 43 10))
  CHGVAR VAR(&NPJOB#)   VALUE(%SST(&REQUEST 53 6))
  CHGVAR VAR(&NPFILE)   VALUE(%SST(&REQUEST 59 10))
  CHGVAR VAR(&NPFIL#)   VALUE(%SST(&REQUEST 69 4))
  CHGVAR VAR(&NPLEN)    VALUE(%SST(&REQUEST 73 4))
  CHGVAR VAR(&DECLN)    VALUE(%BINARY(&NPLEN 1 4))
  CHGVAR VAR(&NPDATA)   VALUE(%SST(&REQUEST 77 &DECLN))
ENDDO

```

```

/* SERWER KOLEJKI DANYCH */
  CHGVAR VAR(&DQFMT)    VALUE(%SST(&REQUEST 21 8))
  CHGVAR VAR(&DQFID)    VALUE(%SST(&REQUEST 29 4))
  CHGVAR VAR(&DQOOBJ)   VALUE(%SST(&REQUEST 33 10))
  CHGVAR VAR(&DQOLIB)   VALUE(%SST(&REQUEST 43 10))
  CHGVAR VAR(&DQOROP)   VALUE(%SST(&REQUEST 53 2))
  CHGVAR VAR(&DQOLEN)   VALUE(%SST(&REQUEST 55 4))
  CHGVAR VAR(&DQOKEY)   VALUE(%SST(&REQUEST 59 256))

```

```

/* SERWER CENTRALNY */
  CHGVAR VAR(&CSFMT)    VALUE(%SST(&REQUEST 21 8))
  CHGVAR VAR(&CSFID)    VALUE(%SST(&REQUEST 29 4))

```

```

/* JEŚLI FORMAT JEST ZSCL0100 */
IF COND(&CSFMT *EQ 'ZSCL0100') THEN(DO)
  CHGVAR VAR(&CSCNAM)   VALUE(%SST(&REQUEST 33 255))
  CHGVAR VAR(&CSLUSR)   VALUE(%SST(&REQUEST 288 8))
  CHGVAR VAR(&CSPID)    VALUE(%SST(&REQUEST 296 7))
  CHGVAR VAR(&CSFID)    VALUE(%SST(&REQUEST 303 4))
  CHGVAR VAR(&CSRID)    VALUE(%SST(&REQUEST 307 6))
  CHGVAR VAR(&CSTYPE)   VALUE(%SST(&REQUEST 313 2))
ENDDO

```

```

/* JEŚLI FORMAT JEST ZSCS0100 */
IF COND(&CSFMT *EQ 'ZSCS0100') THEN(DO)
  CHGVAR VAR(&CSCNAM)   VALUE(%SST(&REQUEST 33 255))
  CHGVAR VAR(&CSCMTY)   VALUE(%SST(&REQUEST 288 255))
  CHGVAR VAR(&CSNODE)   VALUE(%SST(&REQUEST 543 1))
  CHGVAR VAR(&CSNNAM)   VALUE(%SST(&REQUEST 544 255))
ENDDO

```

```

/* JEŚLI FORMAT JEST ZSCN0100 */
IF COND(&CSFMT *EQ 'ZSCN0100') THEN(DO)
  CHGVAR VAR(&CSFROM)   VALUE(%SST(&REQUEST 33 4))
  CHGVAR VAR(&CSTO)     VALUE(%SST(&REQUEST 37 4))
  CHGVAR VAR(&CSCSTYP)  VALUE(%SST(&REQUEST 41 2))
ENDDO

```

```

/* SERWER BAZ DANYCH */
  CHGVAR VAR(&DBFMT)    VALUE(%SST(&REQUEST 21 8))
  CHGVAR VAR(&DBFID)    VALUE(%SST(&REQUEST 29 4))

```

```

/* JEŚLI FORMAT JEST ZDAD0100 */
IF COND(&CSFMT *EQ 'ZDAD0100') THEN(DO)
  CHGVAR VAR(&DBFILE) VALUE(%SST(&REQUEST 33 128))
  CHGVAR VAR(&DBLIB) VALUE(%SST(&REQUEST 161 10))
  CHGVAR VAR(&DBMBR) VALUE(%SST(&REQUEST 171 10))
  CHGVAR VAR(&DBAUT) VALUE(%SST(&REQUEST 181 10))
  CHGVAR VAR(&DBBFIL) VALUE(%SST(&REQUEST 191 128))
  CHGVAR VAR(&DBBLIB) VALUE(%SST(&REQUEST 319 10))
  CHGVAR VAR(&DBOFIL) VALUE(%SST(&REQUEST 329 10))
  CHGVAR VAR(&DBOLIB) VALUE(%SST(&REQUEST 339 10))
  CHGVAR VAR(&DBOMBR) VALUE(%SST(&REQUEST 349 10))
ENDDO

/* JEŚLI FORMAT JEST ZDAD0200 */
IF COND(&CSFMT *EQ 'ZDAD0200') THEN(DO)
  CHGVAR VAR(&DBNUM) VALUE(%SST(&REQUEST 33 4))
  CHGVAR VAR(&DBLIB2) VALUE(%SST(&REQUEST 37 10))
ENDDO

/* JEŚLI FORMAT JEST ZDAQ0100 */
IF COND(&CSFMT *EQ 'ZDAQ0100') THEN(DO)
  CHGVAR VAR(&DBSTMT) VALUE(%SST(&REQUEST 33 18))
  CHGVAR VAR(&DBCRSR) VALUE(%SST(&REQUEST 51 18))
  CHGVAR VAR(&DBOPT) VALUE(%SST(&REQUEST 69 2))
  CHGVAR VAR(&DBATTR) VALUE(%SST(&REQUEST 71 2))
  CHGVAR VAR(&DBPKG) VALUE(%SST(&REQUEST 73 10))
  CHGVAR VAR(&DBPLIB) VALUE(%SST(&REQUEST 83 10))
  CHGVAR VAR(&DBDRDA) VALUE(%SST(&REQUEST 93 2))
  CHGVAR VAR(&DBCMT) VALUE(%SST(&REQUEST 95 1))
  CHGVAR VAR(&DBTEXT) VALUE(%SST(&REQUEST 96 512))
ENDDO

/* JEŚLI FORMAT JEST ZDAR0100 */
IF COND(&CSFMT *EQ 'ZDAR0100') THEN(DO)
  CHGVAR VAR(&DBLIBR) VALUE(%SST(&REQUEST 33 20))
  CHGVAR VAR(&DBRDBN) VALUE(%SST(&REQUEST 53 36))
  CHGVAR VAR(&DBPKGR) VALUE(%SST(&REQUEST 69 20))
  CHGVAR VAR(&DBATTR) VALUE(%SST(&REQUEST 89 20))
  CHGVAR VAR(&DBFILR) VALUE(%SST(&REQUEST 109 256))
  CHGVAR VAR(&DBMBRR) VALUE(%SST(&REQUEST 365 20))
  CHGVAR VAR(&DBFFMT) VALUE(%SST(&REQUEST 385 20))
ENDDO

/* DODATKOWE PARAMETRY DLA FORMATU ZDAR0200 */
/* JEŚLI FORMAT JEST ZDAR0200 */
IF COND(&CSFMT *EQ 'ZDAR0200') THEN(DO)
  CHGVAR VAR(&DBPLIB) VALUE(%SST(&REQUEST 33 10))
  CHGVAR VAR(&DBPTBL) VALUE(%SST(&REQUEST 43 128))
  CHGVAR VAR(&DBFLIB) VALUE(%SST(&REQUEST 171 10))
  CHGVAR VAR(&DBFTBL) VALUE(%SST(&REQUEST 181 128))
ENDDO

/* SERWER KOMEND ZDALNYCH */
CHGVAR VAR(&RCFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&RCFID) VALUE(%SST(&REQUEST 29 4))

```

```

CHGVAR VAR(&RCPGM) VALUE(%SST(&REQUEST 33 10))
CHGVAR VAR(&RCLIB) VALUE(%SST(&REQUEST 43 10))
CHGVAR VAR(&RCNUM) VALUE(%SST(&REQUEST 53 4))
CHGVAR VAR(&RCDATA) VALUE(%SST(&REQUEST 57 6000))

/* DEKLARACJE SERWERA WPISYWANIA SIĘ DO SYSTEMU */
CHGVAR VAR(&SOFMT) VALUE(%SST(&REQUEST 21 8))
CHGVAR VAR(&SOFID) VALUE(%SST(&REQUEST 29 4))

/*****
/*
/* POCZĄTEK PROGRAMU GŁÓWNEGO
/*
/*
*****/

CHGVAR VAR(&STATUS) VALUE('1')
/* INICJOWANIE WARTOŚCI
POWROTU DO ZAAKCEPTOWANIA ZGŁOSZENIA */

/* DODANIE WSPÓLNEJ LOGIKI DO WSZYSTKICH SERWERÓW */

/* PRZETWARZANIE W OPARCIU O ID SERWERA*/
IF COND(&APPLIC *EQ '*VPRT') THEN(GOTO CMDLBL(VPRT)) /* JEŚLI DRUKARKA WIRT*/
IF COND(&APPLIC *EQ '*TFRFCL') THEN(GOTO CMDLBL(TFR)) /* JEŚLI FUNK.TRANSFER*/
IF COND(&APPLIC *EQ '*FILESRV') THEN(GOTO CMDLBL(FLR)) /* JEŚLI SERWERY PLIK.*/
IF COND(&APPLIC *EQ '*MSGFCL') THEN(GOTO CMDLBL(MSG)) /* JEŚLI F.PRZES.KOMUN*/
IF COND(&APPLIC *EQ '*DQSRV') THEN(GOTO CMDLBL(DATAQ)) /* JEŚLI KOLEJKI DANYC*/
IF COND(&APPLIC *EQ '*RQSRV') THEN(GOTO CMDLBL(RSQL)) /* JEŚLI ZDALNE SQL
IF COND(&APPLIC *EQ '*SQL') THEN(GOTO CMDLBL(SQLINIT)) /* JEŚLI SQL
IF COND(&APPLIC *EQ '*NDB') THEN(GOTO CMDLBL(NDB)) /* JEŚLI RODZIMA B.DAN*/
IF COND(&APPLIC *EQ '*SQLSRV') THEN(GOTO CMDLBL(SQLSRV)) /* JEŚLI SQL
IF COND(&APPLIC *EQ '*RTVOBJINF') THEN(GOTO CMDLBL(RTVOBJ)) /* JEŚLI POBR.OBI*/
IF COND(&APPLIC *EQ '*DATAQSRV') THEN(GOTO CMDLBL(ODATAQ)) /* JEŚLI D
IF COND(&APPLIC *EQ '*QNPSERV') THEN(GOTO CMDLBL(NETPRT)) /* JEŚLI WYDR.SIEC*/
IF COND(&APPLIC *EQ '*CNTRLSRV') THEN(GOTO CMDLBL(CENTRAL)) /* JEŚLI S.CENTRA*/
IF COND(&APPLIC *EQ '*RMTSRV') THEN(GOTO CMDLBL(RMTCMD)) /* JEŚLI RMTCMD/D*/
IF COND(&APPLIC *EQ '*SIGNON') THEN(GOTO CMDLBL(SIGNON)) /* JEŚLI WPIS.DO SY*/

GOTO EXIT

/* * * * * *
/* PODPROGRAMY
/*
/*
/* * * * * *

/* DRUKARKA WIRTUALNA */
VPRT:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

GOTO EXIT
/* FUNKCJA PRZESYŁANIA */
TFR:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

GOTO EXIT

/* SERWER PLIKÓW */
FLR:

```

```

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT
/* FUNKCJA PRZESYŁANIA */
MSG:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT
/* KOLEJKI DANYCH */
DATAQ:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT

/* ZDALNE SQL */
RSQL:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT
/* INICJACJA BAZY DANYCH */
SQLINIT:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT

/* RODZIMA BAZA DANYCH */
NDB:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT
/* BAZA DANYCH SQL */
SQLSRV:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT

/* POBRANIE INFORMACJI O OBIEKCIE */
RTVOBJ:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT

/* SERWER KOLEJKI DANYCH */
ODATAQ:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT
/* SIECIOWY SERWER WYDRUKÓW */
NETPRT:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT
/* SERWER CENTRALNY */
CENTRAL:

```

```

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

    GOTO EXIT
/* KOMENDA ZDALNA I ROZPROSZONE WYWOŁANIE PROGRAMU */
RMTCMD:

/* W TYM PRZYPADKU, JEŚLI UŻYTKOWNIK PRÓBUJE WYKONAĆ KOMENDĘ ZDALNĄ */
/* ROZPROSZONE WYWOŁANIE PROGRAMU I JEGO USERID MA WARTOŚĆ userid, NIE */
/* BĘDZIE MÓGŁ KONTYNUOWAĆ */
IF COND(&USER *EQ 'userid') THEN(CHGVAR VAR(&STATUS) VALUE('0'))

    GOTO EXIT
/* SERWER WPISYWANIA SIĘ DO SYSTEMU */
SIGNON:

/* TU ZNAJDUJE SIĘ OKREŚLONA LOGIKA */

GOTO EXIT

EXIT:
ENDPGM

```

Administrowanie iSeries NetServer



iSeries^(TM) Access for Windows^(R) wykorzystuje funkcję systemu operacyjnego IBM^(R) Operating System/400^(R) (OS/400^(R)) o nazwie IBM iSeries Support for Windows Network Neighborhood (iSeries NetServer), umożliwiającą udostępnianie plików i drukarek.

Kompletną dokumentację na temat konfigurowania, administrowania i wykorzystania iSeries NetServer zawiera sekcja iSeries NetServer.

Konfigurowanie ograniczeń użytkowników za pomocą strategii i funkcji Administrowania aplikacjami

iSeries^(TM) Access for Windows^(R) obsługuje dwie podstawowe metody wprowadzania kontroli administratora w sieci: Administrowanie aplikacjami i strategię. Usługa Administracja aplikacji wykorzystuje możliwość wprowadzania ograniczeń w profilu użytkownika iSeries i jest zarządzana za pomocą programu iSeries Navigator. Strategie ustalają ustawienia konfiguracyjne i ograniczenia oraz mogą dotyczyć zarówno określonych komputerów PC, jak i poszczególnych profili użytkowników Windows. W ten sposób oferują większą dokładność zabezpieczeń niż Administracja aplikacjami, ale znacznie trudniej je konfigurować i nimi zarządzać. Aby używać strategii, należy pobrać “Edytor strategii systemowych Microsoft” na stronie 94 i skonfigurować komputery PC oraz serwer iSeries tak, aby mogły przechowywać, pobierać i stosować wprowadzone strategie. Zaleca się użycie Administracji aplikacji w przypadku, gdy wszystkie funkcje, które będą ograniczone, są dostępne dla Administracji aplikacji i jeśli używana wersja OS/400^(R) obsługuje to oprogramowanie.

W wersji V5R2, w Administracji aplikacji dodano obsługę Ustawień centralnych. Obsługa Ustawień centralnych umożliwia zarządzanie elementami sterującymi większości funkcji aplikacji iSeries Access for Windows za pomocą następujących szablonów strategii:

- Ograniczenia wykonawcze (caerestr.adm)
- Obowiązkowe właściwości połączenia (config.adm)
- Strategie konfiguracyjne (caecfg.adm)

Więcej informacji o Administracji aplikacji zawiera dokument Administracja aplikacji.

Więcej informacji o strategiach zawierają następujące tematy:

- “Przegląd strategii iSeries Access for Windows”
- “Konfigurowanie systemu w celu korzystania ze strategii” na stronie 93
- “Lista strategii iSeries Access for Windows” na stronie 95

Przegląd strategii iSeries Access for Windows

Strategii iSeries^(TM) Access for Windows^(R) można używać w celu uniemożliwienia użytkownikom wykonywania określonych działań lub wskazania, bądź wymuszenia pewnych elementów konfiguracyjnych. Strategie systemu mogą dotyczyć profili indywidualnych użytkowników Windows i konkretnych komputerów PC. Nie zapewniają jednak kontroli nad zasobami serwera iSeries i nie zastępują ochrony iSeries. Opis funkcji pełnionych przez strategie serwera zawiera dokument “Typy i zasięgi strategii” na stronie 92.

Wykorzystanie strategii grupy do sterowania użyciem i konfigurowaniem iSeries Access for Windows nie zostało do końca przetestowane i z tego powodu wyników nie da się przewidzieć. Dodatkowe informacje dotyczące strategii grupy zawiera dokumentacja firmy Microsoft^(R). Pozostała część dokumentu opisuje wykorzystanie przetestowanych strategii iSeries Access for Windows.

Obsługa strategii w sieci

Strategie znajdują się na serwerze zbiorów. Za każdym razem, kiedy użytkownik wpisuje się do stacji roboczej Windows, stacja ta ładuje wszystkie strategie, które odnoszą się do profilu użytkownika. Komputer PC użytkownika wpisuje strategie do rejestru zanim użytkownik wykona jakiegokolwiek czynności na stacji roboczej. Wszystkie systemy operacyjne Windows dostarczane są z kodem wymaganym do pobrania strategii.

Aby w pełni wykorzystać możliwości strategii, potrzebne są:

- serwer podstawowy,
- serwer strategii.

Jako serwera strategii można użyć IBM^(R) iSeries Support for Windows Network Neighborhood (iSeries NetServer). Serwerami mogą być systemy Windows NT/2000, jak i Novell Netware.

Dodatkowe informacje zawiera sekcja “Konfigurowanie systemu w celu korzystania ze strategii” na stronie 93.

Pliki strategii

Definicje strategii znajdują się w szablonach strategii, które organizują je w kategorie. iSeries Access for Windows zawiera pięć szablonów strategii, po jednym dla każdej z następujących funkcji:

- ograniczanie funkcji iSeries Access for Windows dla danego systemu (sysname.adm),
- ograniczanie konkretnej funkcji iSeries Access for Windows w czasie jej wykonywania (caerestr.adm),
- ograniczanie komponentów, które użytkownicy mogą zainstalować lub deinstalować (caeinrst.adm),
- narzucanie lub sugerowanie ustawień dla konkretnych środowisk, znajdujących się w nich systemów i wartości konfiguracyjnych tych systemów (config.adm),
- sugerowanie lub narzucanie globalnych wartości konfiguracyjnych (caecfg.adm).

Szablony strategii należy utworzyć za pomocą narzędzia CWBADGEN przed utworzeniem lub zmianą konkretnych strategii. Następnie, za pomocą narzędzia “Edytor strategii systemowych Microsoft” na stronie 94 należy uaktywnić szablony i skonfigurować ich strategie. Po skonfigurowaniu strategii należy zapisać zmiany w pliku strategii, na przykład (nt)config.pol.

Uwaga: Strategie dla różnych systemów operacyjnych Windows muszą być tworzone i obsługiwane oddzielnie. Szczegóły na ten temat zawiera dokumentacja firmy Microsoft.

Więcej informacji można znaleźć w sekcji “Tworzenie plików strategii” na stronie 93.

Typy i zasięgi strategii

Każda strategia, którą udostępnia program iSeries^(TM) Access for Windows^(R), jest albo strategią ograniczenia, albo konfiguracji i może obejmować jeden lub wiele zasięgów.

Strategie ograniczeń

Strategie ograniczeń można konfigurować dla dowolnego zasięgu i używać do następujących celów:

- Ograniczanie użycia funkcji lub czynności programu iSeries Access for Windows albo zezwolenie na jej użycie.
- Ograniczanie instalowania lub deinstalowania komponentów, pakietów serwisowych, aktualizacji lub całych produktów.
- Włączenie kilku innych ograniczeń. Na przykład za pomocą strategii Blokowanie wszystkich transmisji danych do serwera iSeries można ograniczyć pewien typ transmisji danych lub jednocześnie wszystkie typy transmisji danych.
- Ukrycie lub wyłączenie zazwyczaj dostępnych kontrolerek lub opcji.
- Poinformowanie użytkownika, zwykle w formie komunikatu na konsoli lub w oknie, kiedy próbuje on zastosować funkcję, która jest chroniona przez strategię ograniczenia.

Strategie konfiguracyjne

Strategie konfiguracyjne można określać tylko dla użytkowników i używać w następujących celach:

- Wstępne konfigurowanie tego, co użytkownik końcowy zazwyczaj konfiguruje samodzielnie.
- Konfigurowanie wartości i opcji, które normalnie użytkownicy mogą włączać lub wyłączać, oraz wskazywanie środowiska i połączenia.
- Blokowanie możliwości zmiany wartości. Jeśli strategia konfiguracyjna narzuca jakąś wartość, pole wejściowe dla tej wartości nie będzie akceptowało zmian.

Strategie konfiguracyjne mogą być zalecane lub wymagane.

- Zalecane: zostanie użyta udostępniona wartość, chyba że zostanie ona zmieniona przez użytkownika lub ustawiona przez aplikację. W ten sposób można zastąpić wartość domyślną, jakiej użyłby program iSeries Access for Windows, gdyby użytkownik nie podał innej wartości. Jednak użycie wartości zalecanej nie jest konieczne - użytkownik może podać inną wartość.
- Wymagane: udostępniona wartość będzie używana - ani użytkownik, ani aplikacja nie może jej zmienić.

Zasięgi strategii

Strategie można konfigurować dla trzech zasięgów: użytkownika, maszyny i połączenia iSeries. Dla niektórych strategii można ustawić więcej niż jeden zasięg.

Zasięg	Opis
Zasięg działania komputera	Strategię z ustawionym tego rodzaju zasięgiem stosuje się do wszystkich użytkowników komputerów PC. Jedyne wyjątek to sytuacja, gdy ta sama strategia ustawiona jest dla konkretnego użytkownika w celu przesłonięcia ustawień zasięgu działania komputera.
Zasięg działania użytkownika	Strategię z ustawionym tego rodzaju zasięgiem stosuje się do konkretnych użytkowników komputerów PC. Może ona być ustawiona dla niektórych użytkowników. Może być również ustawiona dla "użytkownika domyślnego" (dowolnego użytkownika bez indywidualnej konfiguracji strategii). Niektóre strategie o zasięgu działania użytkownika udostępniają ustawienia pozwalające na stosowanie funkcji bez względu na ustawienia zasięgu działania komputera. Jeśli są one używane, ustawienia zasięgu działania komputera są ignorowane.

Zasięg	Opis
Zasięg połączenia systemu iSeries (lub określonego systemu)	<p>Niektóre strategie, których zasięg może być ustawiony jako zasięg działania użytkownika lub zasięg działania komputera, mogą mieć zasięg zdefiniowany jako zasięg połączenia z systemem iSeries w obrębie zasięgu użytkownika lub komputera. Ustawienia strategii dla zasięgu połączenia z systemem iSeries działają tylko podczas pracy z systemem typu iSeries. Na przykład, jeśli strategia ograniczenia ma zasięg połączenia systemu iSeries w obrębie zasięgu działania użytkownika, a system iSeries nazwany jest SYS1 i użytkownik nazwany jest USER1, dana funkcja jest ograniczona, jeśli USER1 pracuje w SYS1.</p> <p>Uwaga: Jeśli zasięg strategii ustalony jest jako zasięg połączenia systemu iSeries, ustawienie to ma priorytet wyższy niż zasięg działania użytkownika i komputera. Na przykład, jeśli domyślny tryb użytkownika jest narzucony dla USER1 jako "Użycie domyślnego identyfikatora użytkownika", ale ustawieniem dla systemu SYS1 jest "Użycie identyfikatora oraz hasła użytkownika Windows", to gdy USER1 połączy się z SYS1 użyte zostanie jego hasło i identyfikator dla systemu Windows. Jeśli USER1 połączy się z dowolnym innym systemem, zostanie użyty określony domyślny identyfikator użytkownika.</p> <p>Uwaga: Aby umożliwić ustawienie tego zasięgu strategii, trzeba wygenerować i wykorzystać następujące szablony strategii:</p> <ul style="list-style-type: none"> • config.adm - Szablon konfigurowania środowisk i połączeń • sysname.adm - Szablon konkretnego systemu (według nazwy systemu iSeries)

Konfigurowanie systemu w celu korzystania ze strategii

Aby korzystać ze strategii iSeries^(TM) Access for Windows^(R), należy wykonać poniższe czynności:

1. "Konfigurowanie serwera iSeries^(TM) pod kątem strategii"
2. "Konfigurowanie klientów PC pod kątem strategii"
3. "Tworzenie plików strategii".

Konfigurowanie serwera iSeries^(TM) pod kątem strategii

Aby skonfigurować system AS/400 pod kątem obsługi strategii, wykonaj poniższe czynności. Przyjęto założenie, że na komputerach PC znajdujących się w sieci zainstalowany jest system Windows^(R).

- Skonfiguruj system AS/400 jako iSeries NetServer (chyba, że już jest skonfigurowany).
- Utwórz folder zintegrowanego systemu plików, w którym będą przechowywane pliki strategii.


Konfigurowanie klientów PC pod kątem strategii

Aby komputery PC w sieci mogły pobrać strategię z systemu iSeries^(TM), należy je odpowiednio skonfigurować.

➤ Każda stacja robocza Windows^(R) w sieci musi pobrać utworzony plik strategii. Istnieje możliwość pobrania narzędzi, które wykonują tę czynność automatycznie. Pobierz cwbpoluz ze strony

www.as400.ibm.com/clientaccess/cadownld.htm  . <<

Jeśli plik strategii zostanie umieszczony w zasobie współużytkowanym NETLOGON serwera wpisywania się iSeries, komputery PC użytkowników pobiorą go automatycznie, gdy użytkownicy wpiszą się do domeny iSeries.

Konfigurowanie Windows pod kątem strategii: Każda stacja robocza Windows^(R) w sieci musi pobrać utworzony plik strategii. Istnieje możliwość pobrania narzędzi, które wykonują tę czynność automatycznie. Pobierz cwbpoluz ze strony www.as400.ibm.com/clientaccess/cadownld.htm .

Tworzenie plików strategii

Aby utworzyć lub zmodyfikować określone strategię, należy pobrać edytor strategii firmy Microsoft^(R), wygenerować szablony strategii i utworzyć lub zmodyfikować plik strategii.

1. "Edytor strategii systemowych Microsoft" na stronie 94.

2. "Tworzenie szablonów strategii dla iSeries Access for Windows".
3. "Tworzenie i aktualizacja plików strategii".

Uwaga: Strategie dla różnych systemów operacyjnych Windows muszą być tworzone i obsługiwane oddzielnie. Szczegóły na ten temat zawiera dokumentacja firmy Microsoft.

Edytor strategii systemowych Microsoft: Aby utworzyć własne pliki strategii, potrzebny jest edytor strategii dostarczany przez Microsoft^(R). Bieżąca wersja edytora strategii jest dostarczana z Windows NT^(R) Server, Windows NT Workstation Resource Kit i Office 97 Resource Kit. Jest ona również dostępna na stronie WWW firmy Microsoft. System Windows 2000^(R) wymaga własnej wersji edytora strategii, która jest dostarczana z systemem Windows 2000 Server.

www.microsoft.com 

Poszukaj hasła **policy editor**. Starsza wersja edytora strategii jest dostarczana na dysku instalacyjnym Windows 95. Nie należy jej używać. Wersja ta umożliwia stosowanie tylko jednego szablonu strategii w danym momencie.

Do edytora strategii dołączone są wskazówki dotyczące rozpakowania plików instalacyjnych oraz instalacji edytora i szablonów strategii.

Tworzenie szablonów strategii dla iSeries Access for Windows: iSeries^(TM) Access for Windows^(R) zawiera program, który tworzy szablony używane do sterowania strategią.

1. Otwórz okno wiersza komend.
2. Przejdź do katalogu iSeries Access for Windows, zwykle jest to katalog:
[C:]Program Files\IBM\Client Access\
3. Aby utworzyć szablon strategii, która ma zostać skonfigurowana, wpisz odpowiednią komendę i parametry.

Komendy szablonów strategii

Komenda cwbadgen z parametrami	Opis
cwbadgen /ps S1034345 (gdzie S1034345 jest nazwą systemu)	Generuje szablon ułatwiający konfigurowanie strategii charakterystycznych dla systemu, S1034345.adm.
cwbadgen /std	Generuje szablony: caecfg.adm (dotyczy konfiguracji globalnej), caeinrst.adm (dotyczy ograniczeń instalacji) i caerestr.adm (dotyczy ograniczeń wykonawczych).
cwbadgen /cfg config.adm	Generuje szablon config.adm (strategia konfiguracji oparta na konfiguracjach systemu istniejących na komputerze PC, z którego komenda została uruchomiona). Po opcji /cfg podaj nazwę pliku szablonu. W przykładzie nazwą pliku szablonu jest config.adm.

Tworzenie i aktualizacja plików strategii: Utwórz pliki strategii sterujące czynnościami domyślnymi komputera i użytkownika.

➤ **Uwaga:** Poniższe instrukcje nie obejmują zastosowania Strategii grupy. Aby administrować funkcjami programu iSeries^(TM) Access for Windows^(R) używając Strategii grupy, należy zapoznać się z dokumentacją firmy Microsoft^(R) dotyczącą użycia Strategii grupy. ⏪

1. Uruchom edytor strategii dwukrotnie klikając program **poledit.exe**.
2. Przejdź do **Opcje** → **Szablon strategii** → **Dodaj**.
3. Przejdź do katalogu, w którym zapisano pliki .adm, powstałe podczas tworzenia szablonów strategii.
4. Wybierz odpowiednie pliki .adm, które chcesz dodać, i kliknij **Dodaj**. Powtarzaj tę czynność, aż zostaną dodane wszystkie pliki .adm, których chcesz użyć. Następnie kliknij **OK**.

5. Przejdź do **Plik** —> **Nowa strategia**.
6. Skonfiguruj strategię i zapisz plik strategii:

\\QYOURSYS\POLICIES\ntconfig.pol

gdzie:

- QYOURSYS jest nazwą programu iSeries NetServer.
- POLICIES jest nazwą współużytkowanego folderu plików w programie iSeries NetServer.
- config.pol jest nazwą pliku strategii.

Aby zaktualizować plik strategii, otwórz go w edytorze strategii, wprowadź zmiany i zachowaj plik.

Uwaga: Strategie dla różnych systemów operacyjnych Windows muszą być tworzone i obsługiwane oddzielnie. Szczegóły na ten temat zawiera dokumentacja firmy Microsoft.

Listy strategii iSeries Access for Windows

iSeries^(TM) Access for Windows^(R) obsługuje systemowe strategie firmy Microsoft^(R). Administratorzy mogą używać strategii do kontrolowania, jakie funkcje i ustawienia są dostępne dla każdego użytkownika. W sekcji tej opisano wszystkie strategie, które udostępnia program Client Access Express i opisano efekty oraz zasięg każdej z nich.

» Zestawy strategii są zdefiniowane w plikach szablonów. Za pomocą komendy **cwbadgen** można generować szablony strategii (pliki .adm) dla iSeries Access for Windows na komputerze PC, na którym jest on zainstalowany. Szczegóły na ten temat zawiera sekcja “Tworzenie szablonów strategii dla iSeries Access for Windows” na stronie 94. Lista istniejących strategii jest dostępna po wybraniu jednej z opcji: <<

- “Strategie dla poszczególnych funkcji”
Dokument zawierający listę strategii według funkcji, na jakie mają wpływ.
- “Strategie według szablonu” na stronie 98
Dokument zawierający szablony i przypisane do nich strategie.





Ogólny opis strategii w iSeries Access for Windows zawiera dokument “Przegląd strategii iSeries Access for Windows” na stronie 91.

Strategie dla poszczególnych funkcji

Poniższa tabela przedstawia strategie iSeries^(TM) Access for Windows^(R) według funkcji, na które oddziałują.

Funkcja	Strategie pokrewne
.NET Data provider	Blokowanie używania dostawcy OLE NET
Obiekty automatyzacji ActiveX	<ul style="list-style-type: none"> • Blokowanie obiektu automatyzacji przesyłania danych • Blokowanie obiektu automatyzacji pobierania danych • Blokowanie obiektu automatyzacji komendy zdalnej • Blokowanie obiektu automatyzacji zdalnego programu • Blokowanie obiektu automatyzacji kolejek danych

Funkcja	Strategie pokrewne
Komunikacja	<ul style="list-style-type: none"> • Tryb użytkownika domyślnego • Tryb sprawdzania adresu TCP/IP • Tryb sprawdzania portu • Wymagany protokół SSL • Blokowanie zmian aktywnego środowiska • Blokowanie zmian listy środowisk • Blokowanie połączeń z niezdefiniowanymi wcześniej systemami • Blokowanie używania nieuprawnionych środowisk • Limit czasu połączenia
Przesyłanie danych: pobieranie	<ul style="list-style-type: none"> • Blokowanie przesyłania danych do serwera iSeries • Blokowanie dopisywania i zastępowania zbiorów hosta • Blokowanie ładowania danych do hosta z poziomu interfejsu GUI • Blokowanie użycia komendy RFROMPCB • Blokowanie ładowania danych do hosta podczas autostartu • Blokowanie pobierania dodatków programu Excel
Przesyłanie danych: pobieranie	<ul style="list-style-type: none"> • Blokowanie pobierania danych z serwera iSeries • Blokowanie ładowania danych z hosta z poziomu GUI • Blokowanie użycia komendy RTOPCB • ,Blokowanie pobierania danych podczas autostartu • ,Blokowanie pobierania danych przez dodatki programu Excel .
Przesyłanie danych: tworzenie zbioru na serwerze iSeries	<ul style="list-style-type: none"> • Blokowanie tworzenia zbiorów na komputerze hosta • Blokowanie tworzenia zbiorów na serwerze iSeries przez kreatora • Blokowanie tworzenia zbiorów na serwerze iSeries bez użycia kreatora
Aktualizacja katalogu	Blokowanie aktualizacji katalogu
Przychodząca komenda zdalna	<ul style="list-style-type: none"> • Uruchom jako system • Tryb komendy • Ochrona pamięci podręcznej • Dopuszczalna ochrona ogólna • Ochrona ogólna uruchamia komendy jak zalogowany użytkownik

Funkcja	Strategie pokrewne
Instalowanie	 <ul style="list-style-type: none"> • Katalog źródeł instalacji selektywnej • Blokowanie instalacji • Blokowanie instalacji selektywnej • Blokowanie deinstalacji • Blokowanie sprawdzania poziomu pakietu serwisowego • Blokowanie instalacji pakietu serwisowego • Blokowanie aktualizacji • Blokowanie instalacji pojedynczych komponentów • Blokowanie instalacji dodatków 
Zarządzanie licencjami	Opóźnienie zwolnienia licencji
Obsługa języków międzynarodowych	<ul style="list-style-type: none"> • Strona kodowa ANSI • Strona kodowa OEM • Strona kodowa EBCDIC • Transformacja dwukierunkowa
Interfejs ODBC	<ul style="list-style-type: none"> • Nazwane źródła danych • Blokowanie używania źródeł danych wygenerowanych przez program
Dostawca OLE DB	Blokowanie używania Dostawcy OLE DB
Aplikacja iSeries Navigator	Blokowanie użycia iSeries Navigator
Hasła	 <ul style="list-style-type: none"> • Ostrzeżenie użytkownika o wygaśnięciu hasła iSeries • Blokowanie zmian haseł iSeries Access for Windows 

Funkcja	Strategie pokrewne
Emulacja terminalu PC5250	<ul style="list-style-type: none"> • Blokowanie konfiguracji sesji terminalu • Blokowanie konfiguracji sesji drukarki • Blokowanie używania emulatora PC5250 • Maksymalna liczba sesji PC5250 • Blokowanie zmian profili .WS • Blokowanie konfiguracji menu • Blokowanie konfiguracji paska narzędzi • Blokowanie konfiguracji sesji • Blokowanie konfiguracji klawiatury • Blokowanie konfiguracji myszy • Blokowanie wykonywania apletów Java^(TM) • Blokowanie dostępu do makr • Blokowanie importowania profili w menedżerze sesji emulatora • Blokowanie usuwania profili w menedżerze sesji emulatora • Blokowanie zmian katalogów w menedżerze sesji emulatora
Komendy PC	<ul style="list-style-type: none"> • cwblogon • cwbcfg • cwback • cwrest • cwbenv • cwbundbs • wrksplf • wrkmsg • wrkppt • wrkustj
Usługa	<ul style="list-style-type: none"> • Kiedy sprawdzać • Czas opóźnienia • Częstotliwość • Kopiowanie obrazu do komputera PC • Ciche uruchamianie • Ścieżka obsługi • Automatyczne uruchamianie usługi drugoplanowej
Interfejs użytkownika	Blokowanie tworzenia ikon pulpitu

Strategie według szablonu

Pliki szablonów służą do sterowania strategiami. Więcej informacji na ten temat zawiera sekcja “Tworzenie szablonów strategii dla iSeries Access for Windows” na stronie 94.

Plik szablonu	Opis
caecfg.adm	Strategie, które sugerują lub narzucają określone wartości ustawiane podczas konfigurowania. Aby je wygenerować, należy uruchomić program cwbadgen z opcją /std.

Plik szablonu	Opis
caerestr.adm	Strategie, które ograniczają określone funkcje programu iSeries ^(TM) Access for Windows. Aby je wygenerować, należy uruchomić program cwbadgen z opcją /std.
config.adm	Strategie, które narzucają ustawienia konfiguracji dla określonych środowisk, systemów w obrębie tych środowisk oraz niektóre konfigurowalne wartości dla tych systemów. Aby je wygenerować, należy uruchomić program cwbadgen z opcją /cfg.
caeinrst.adm	Strategie, które ograniczają zakres wykonywanych przez użytkownika instalacji i deinstalacji. Ograniczają również inne funkcje związane z instalacją. Aby je wygenerować, należy uruchomić program cwbadgen z opcją /std.
SYSNAME.adm	Strategie, które ograniczają określone funkcje programu iSeries Access for Windows dla danego systemu. Aby je wygenerować, należy uruchomić program cwbadgen z opcją /ps.

Administrowanie Secure Sockets Layer (SSL)

Secure Sockets Layer (SSL) jest popularnym mechanizmem ochrony, umożliwiającym klientowi PC uwierzytelnianie serwera i szyfrowanie wszystkich danych i żądań. Warto go użyć podczas przesyłania ważnych danych pomiędzy klientami a serwerami. Przykładem typowych transakcji klient/serwer korzystających z zalet SSL może być przekazywanie danych o kartach kredytowych i operacjach bankowych. Ze względu na dodatkowy proces szyfrowania i deszyfrowania użycie SSL powoduje wzrost kosztów. ➤

iSeries^(TM) Access for Windows^(R) zawiera opcjonalnie instalowalną obsługę warstwy SSL (Secure Sockets Layer) i możliwość zarządzania bazami danych kluczy za pomocą **IBM^(R) Key Management**. Wszystkie funkcje iSeries Access for Windows, z wyjątkiem Przychodzącej zdalnej komendy (Incoming Remote Command) mogą komunikować się przez warstwę SSL. iSeries Access for Windows umożliwia komunikację z serwerem iSeries na poziomie szyfrowania 128-bitowego. ⏪

Uwierzytelnianie klienta jest dostępne dla PC5250.

Dodatek. Uwagi

Informacja ta przeznaczona jest dla produktów i usług oferowanych w USA.

Firma IBM może nie oferować w innych krajach produktów, usług lub opcji omawianych w tej publikacji. Informacje o produktach i usługach dostępnych w danym kraju można uzyskać od lokalnego przedstawiciela IBM. Odwołanie do produktu, programu lub usługi IBM nie oznacza, że można użyć wyłącznie tego produktu, programu lub usługi. Zamiast nich można zastosować ich odpowiednik funkcjonalny, pod warunkiem, że nie narusza to praw własności intelektualnej firmy IBM. Jednakże cała odpowiedzialność za ocenę przydatności i sprawdzenie działania produktu, programu lub usługi, pochodzących od producenta innego niż IBM spoczywa na użytkowniku.

Firma IBM może być w posiadaniu patentów lub wniosków oczekujących na ich otrzymanie, a dotyczących tematów opisanych w tym dokumencie. Przedstawienie tej publikacji nie daje żadnych uprawnień licencyjnych do tychże patentów. Pisemne zapytania w sprawie licencji można przesyłać na adres:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594-1785
U.S.A.

Zapytania o licencję dotyczącą zestawu DBCS można wyjaśnić w Departamencie Własności Intelektualnych firmy IBM we własnym kraju lub przesłać je w formie pisemnej na adres:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

Poniższy akapit nie obowiązuje w Wielkiej Brytanii, a także w innych krajach, w których jego treść pozostaje w sprzeczności z przepisami prawa miejscowego: INTERNATIONAL BUSINESS MACHINES CORPORATION DOSTARCZA TĘ PUBLIKACJĘ W TAKIM STANIE, W JAKIM SIĘ ("AS IS"), BEZ JAKIKOLWIEK GWARANCJI (W TYM TAKŻE RĘKOJMI), WYRAŻNYCH LUB DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ, PRZYDATNOŚCI DO OKREŚLONEGO CELU ORAZ GWARANCJI, ŻE PUBLIKACJA TA NIE NARUSZA PRAW OSÓB TRZECICH. Ustawodawstwa niektórych krajów nie dopuszczają zastrzeżeń dotyczących gwarancji wyraźnych i domniemanych w odniesieniu od pewnych transakcji; w takiej sytuacji powyższe zdanie nie ma zastosowania.

Informacje zawarte w niniejszej publikacji mogą zawierać nieścisłości techniczne lub błędy drukarskie. Informacje te są okresowo aktualizowane, a zmiany te zostaną uwzględnione w kolejnych wydaniach tej publikacji. Firma IBM zastrzega sobie prawo do wprowadzania ulepszeń i/lub zmian w produktach i/lub programach opisanych w tej publikacji w dowolnym czasie, bez wcześniejszego powiadomienia.

Wszelkie wzmianki w tej publikacji na temat stron internetowych innych firm zostały wprowadzone wyłącznie dla wygody użytkownika i w żadnym wypadku nie stanowią zachęty do ich odwiedzania. Materiały dostępne na tych stronach nie są częścią materiałów opracowanych do tego produktu firmy IBM, a użytkownik korzysta z nich na własną odpowiedzialność.

Firma IBM może używać i dystrybuować każdą informację utrzymaną od użytkownika w dowolny sposób, który uważa za odpowiedni bez zaciągania jakichkolwiek zobowiązań względem użytkownika.

Licencjobiorcy tego programu, którzy chcieliby uzyskać informacje na temat programu w celu: (i) wdrożenia wymiany informacji między niezależnie utworzonymi programami i innymi programami (łącznie z tym opisywanym) oraz (ii) wspólnego wykorzystywania wymienianych informacji, powinni skontaktować się z:

IBM Corporation
Software Interoperability Coordinator, Department 49XA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Informacje takie mogą być udostępnione, o ile spełnione zostaną odpowiednie warunki, w tym, w niektórych przypadkach, uiszczenie odpowiedniej opłaty.

Licencjonowany program opisany w tej publikacji i wszystkie inne licencjonowane materiały dostępne dla tego programu są dostarczane przez IBM na warunkach określonych w Umowie IBM z Klientem, Międzynarodowej Umowie Licencyjnej IBM na Program lub w innych podobnych umowach zawartych między IBM i użytkownikami.

Informacje dotyczące produktów firm innych niż IBM pochodzą od dostawców tych produktów, z opublikowanych przez nich zapowiedzi lub innych powszechnie dostępnych źródeł. Firma IBM nie przetestowała i nie może potwierdzić dokładności pomiarów wydajności, kompatybilności ani żadnych innych danych związanych z tymi produktami. Pytania dotyczące produktów firm innych niż IBM należy kierować do dostawców tych produktów.

LICENCJA NA PRAWA AUTORSKIE:

Publikacja ta zawiera przykładowe aplikacje w kodzie źródłowym, które ilustrują techniki programowania na różnych platformach systemowych. Aplikacje te można bezpłatnie kopiować, modyfikować i rozpowszechniać w dowolnej formie w celu tworzenia, używania lub rozpowszechniania aplikacji przeznaczonych dla interfejsu programowania aplikacji systemu operacyjnego, dla którego zostały napisane. Programy przykładowe nie zostały gruntownie przetestowane. IBM nie może zatem gwarantować lub sugerować niezawodności, użyteczności i funkcjonalności tych programów. Użytkownik może kopiować, modyfikować i rozpowszechniać te programy przykładowe w dowolnej formie bez uiszczania opłat w celu rozbudowy, używania, handlowym lub w celu rozpowszechniania aplikacji zgodnych z aplikacyjnym interfejsem programowym IBM.

Każda kopia programu przykładowego lub jakiegokolwiek jego fragment, jak też jakiegokolwiek prace pochodne muszą zawierać następujące uwagi dotyczące praw autorskich:

(C) IBM Corp. 2004. Fragmenty tego kodu pochodzą z Programy przykładowe. (C) Copyright IBM Corp. 1999-2004. Wszelkie prawa zastrzeżone.

Przy przeglądaniu tych informacji w formie elektronicznej, fotografie i ilustracje kolorowe mogą się nie pojawić.

Znaki handlowe

Następujące znaki są znakami handlowymi firmy IBM w USA lub innych krajach:

Advanced Function Presentation
AFP
Application System/400
AS/400
DB2
DB2 Universal Database
Distributed Relational Database Architecture
DRDA
e (logo)
IBM
iSeries
Operating System/400
OS/2
OS/400

Lotus i 1-2-3 są znakami towarowymi International Business Machines Corporation i Lotus Development Corporation w Stanach Zjednoczonych i/lub w innych krajach.

Microsoft, Windows, Windows NT, oraz logo Windows są znakami handlowymi firmy Microsoft Corporation w USA i innych krajach.

Java i wszystkie oparte na Java znaki handlowe są znakami handlowymi firmy Sun Microsystems, Inc. w USA i innych krajach.

UNIX jest zarejestrowanym znakiem handlowym firmy The Open Group w USA i innych krajach.

Nazwy innych firm, produktów lub usług mogą być znakami towarowymi lub znakami usług innych podmiotów gospodarczych.

Warunki pobierania i drukowania publikacji

Pozwolenie użycia wybranych publikacji do pobierania nadane zostaje użytkownikowi na następujących warunkach i ich akceptacji.

Użytek własny: Można powielać publikacje na własny, niekomercyjny użytek, pod warunkiem, że zachowane zostaną wszystkie uwagi odnoszące się do własności. Użytkownik nie ma prawa dystrybuować ani wyświetlać tych publikacji czy ich części, ani też wykonywać z nich prac pochodnych bez wyraźnej zgody IBM^(R).

Użytek komercyjny: Publikacje te mogą być powielane, rozpowszechniane i wyświetlane jedynie na obszarze firmy użytkownika pod warunkiem, że zachowane zostaną wszystkie uwagi odnoszące się do własności. Poza firmą użytkownika żadna obróbka publikacji, ich powielanie, rozpowszechnianie lub wyświetlanie w całości lub części nie jest możliwe bez uzyskania zgody IBM.

Z wyjątkiem zezwoleń wyraźnie udzielonych w niniejszym dokumencie, nie udziela się jakichkolwiek innych zezwoleń, licencji ani praw, wyraźnych czy domniemanych, odnoszących się do tych publikacji czy jakichkolwiek informacji, danych, oprogramowania lub innej własności intelektualnej, o których mowa w niniejszym dokumencie.

IBM zastrzega sobie prawo do anulowania zezwolenia przyznanego w niniejszym dokumencie w każdej sytuacji, gdy, według uznania IBM, korzystanie z tych publikacji jest szkodliwe dla IBM lub jeśli IBM uzna, że warunki niniejszego dokumentu nie są przestrzegane.

Użytkownik ma prawo pobierać, eksportować lub reeksportować niniejsze informacje pod warunkiem zachowania bezwzględnej i pełnej zgodności z obowiązującym prawem i przepisami, w tym ze wszelkimi prawami i przepisami eksportowymi Stanów Zjednoczonych. IBM NIE UDZIELA JAKICHKOLWIEK GWARANCJI, W TYM TAKŻE RĘKOJMI, DOTYCZĄCYCH TREŚCI TYCH PUBLIKACJI. PUBLIKACJE TE SĄ DOSTARCZANE W STANIE, W JAKIM SIĘ ZNAJDUJĄ ("AS-IS") BEZ UDZIELANIA JAKICHKOLWIEK GWARANCJI, W TYM TAKŻE RĘKOJMI, WYRAŹNYCH CZY DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ CZY PRZYDATNOŚCI DO OKREŚLONEGO CELU.

Wszelkie materiały są chronione prawem autorskim IBM Corporation.

Poprzez pobieranie lub drukowanie publikacji z tej strony, użytkownik wyraża swą zgodę z powyższymi warunkami.

Informacje dotyczące kodu

Niniejszy dokument zawiera przykładowe kody programów.

IBM^(R) udziela niewyłącznej licencji na prawa autorskie, stosowanej przy używaniu wszelkich przykładowych kodów programów, na podstawie których można wygenerować podobne funkcje dostosowane do indywidualnych wymagań.

Cały kod przykładowy jest udostępniany przez IBM jedynie do celów ilustracyjnych. Programy przykładowe nie zostały gruntownie przetestowane. IBM nie może zatem gwarantować lub sugerować niezawodności, użyteczności i funkcjonalności tych programów.

Wszelkie zawarte tutaj programy są dostarczane w stanie, w jakim się znajdują ("AS IS") bez udzielania jakichkolwiek gwarancji. Nie udziela się domniemanych gwarancji nienaruszania praw osób trzecich, gwarancji przydatności handlowej oraz przydatności do określonego celu.

IBM