



Systemy IBM - iSeries

# IBM Content Manager OnDemand for iSeries Common Server ODWEK: Instalowanie i konfigurowanie

*Wersja 5 Wydanie 4*

SC85-0120-03







Systemy IBM - iSeries

IBM Content Manager OnDemand for iSeries  
Common Server ODWEK: Instalowanie i konfigurowanie

*Wersja 5 Wydanie 4*

SC85-0120-03

**Uwaga**

Przed wykorzystaniem informacji znajdujących się w podręczniku oraz produktu, którego one dotyczą należy zapoznać się z informacjami, które zawiera “Uwagi” na stronie 145.

**| Wydanie Dziesiąte (Luty 2006)**

| Niniejsze wydanie dotyczy produktu IBM Content Manager OnDemand for iSeries Common Server, Wersja 5 Wydanie 4 oraz  
| wszystkich nowszych wydań i modyfikacji, o ile w nowych wydaniach nie podano inaczej. To wydanie zastępuje publikację  
| SC27-1163-02.

© Copyright International Business Machines Corporation 2001, 2006. Wszelkie prawa zastrzeżone.

# Spis treści

## Informacje o publikacji IBM OnDemand for iSeries Common Server Web Enablement Kit: Podręcznik instalowania i konfigurowania . . . . . v

Dla kogo przeznaczony jest ten podręcznik . . . . .	v
Sposób organizacji tego podręcznika. . . . .	v
Wymagania wstępne i informacje pokrewne . . . . .	v
Inne informacje dostępne w sieci WWW . . . . .	vi
Centrum informacyjne OnDemand . . . . .	vi
Informacje o dostępności OnDemand . . . . .	vi
iSeries Navigator . . . . .	vi
Jak wysyłać uwagi . . . . .	vii

## Podsumowanie zmian . . . . . ix

## Rozdział 1. Przegląd . . . . . 1

Informacje o interfejsach programistycznych . . . . .	2
Informacje o przeglądarkach . . . . .	4
Korzystanie z pakietu ODWEK . . . . .	5
Funkcje produktu . . . . .	5
Dodanie adnotacji . . . . .	6
Zmiana hasła. . . . .	6
Lista trafień dokumentów . . . . .	6
Wylogowanie się . . . . .	6
Zalogowanie się. . . . .	6
Wczytanie dokumentu . . . . .	6
Kryteria wyszukiwania. . . . .	6
Serwerowy wydruk dokumentu . . . . .	7
Aktualizowanie dokumentu . . . . .	7
Podgląd adnotacji . . . . .	7
Ochrona serwera i danych . . . . .	7

## Rozdział 2. Lista kontrolna instalacji . . . . . 9

## Rozdział 3. Instalowanie i konfigurowanie serwera HTTP. . . . . 11

Wymagania instalacyjne . . . . .	11
Inne wymagania . . . . .	11
Instalowanie w systemie i5/OS . . . . .	12
Dalsze kroki . . . . .	13
Określanie pliku ARSWWW.INI . . . . .	13
[@SRV@_DEFAULT] . . . . .	13
[@SRV@_serwer]. . . . .	14
[CONFIGURATION]. . . . .	15
[SECURITY] . . . . .	21
[AFP2HTML] . . . . .	22
[AFP2PDF]. . . . .	24
[MIMETYPES] . . . . .	25
[ATTACHMENT IMAGES]. . . . .	29
[NO HTML] . . . . .	30
[DEFAULT BROWSER]. . . . .	31
[przeglądarka] . . . . .	38
[DEBUG] . . . . .	38
Przykład pliku ARSWWW.INI . . . . .	39

Dalsze kroki . . . . .	41
------------------------	----

## Rozdział 4. Konfigurowanie przykładowych aplikacji . . . . . 43

## Rozdział 5. Instalowanie przeglądarek WWW . . . . . 45

Przegląd. . . . .	45
Wymagania. . . . .	46
Instalowanie . . . . .	46
Dystrybucja plików zdefiniowanych przez użytkownika . . . . .	47
Instalowanie plików przeglądarki AFP Web Viewer . . . . .	48
Dodawanie podkatalogów . . . . .	49
Przechowywanie plików zdefiniowanych przez użytkownika . . . . .	49
Konfigurowanie plików czcionek . . . . .	49
Tworzenie pliku instalacyjnego przeglądarki AFP Web Viewer . . . . .	50
Instalowanie przeglądarki AFP Web Viewer na stacji roboczej użytkownika. . . . .	51
Odwzorowywanie czcionek AFP . . . . .	51
Przeglądarka AFP Web viewer . . . . .	51
Przeglądarka Image Web viewer . . . . .	52
Przeglądarka danych dla drukarki wierszowej w języku Java . . . . .	52
Dalsze kroki . . . . .	57

## Dodatek A. Funkcje API CGI. . . . . 59

Dodanie adnotacji . . . . .	60
Zmiana hasła . . . . .	63
Lista trafień dokumentów . . . . .	65
Wylogowanie się . . . . .	69
Zalogowanie się . . . . .	71
Drukowanie dokumentu (serwer) . . . . .	73
Wczytanie dokumentu . . . . .	77
Kryteria wyszukiwania . . . . .	80
Aktualizowanie dokumentu . . . . .	82
Podgląd adnotacji . . . . .	84

## Dodatek B. Serwlety w języku Java. . . . . 87

## Dodatek C. Funkcje API języka Java . . . . . 89

## Dodatek D. Podręcznik programowania funkcji API w języku Java . . . . . 91

Architektura klient/serwer . . . . .	91
Tworzenie pakietów w środowisku języka Java . . . . .	91
Wskazówki dotyczące programowania. . . . .	92
Konfigurowanie środowiska systemowego . . . . .	92
Ustawianie zmiennych środowiskowych . . . . .	92
Informacje diagnostyczne i związane ze śledzeniem. . . . .	94
Śledzenie . . . . .	94
Obsługa wyjątków. . . . .	94
Stałe . . . . .	95

Uruchamianie aplikacji ODWEK . . . . .	95
Nawiązywanie połączenia z serwerem OnDemand . . . . .	95
Nawiązywanie połączenia . . . . .	95
Ustawianie i uzyskiwanie haseł . . . . .	96
Praca z serwerem OnDemand . . . . .	96
Wyświetlanie grup aplikacji w folderze . . . . .	98
Wyszukiwanie w folderze . . . . .	100
Wyszukiwanie w folderze za pomocą łańcucha SQL . . . . .	104
Anulowanie wyszukiwania . . . . .	106
Wyświetlanie kryteriów wyszukiwania . . . . .	108
Wyświetlanie folderów i informacji o nich . . . . .	112
Wyświetlanie listy dokumentów . . . . .	113
Wczytywanie dokumentu . . . . .	115
Drukowanie dokumentu. . . . .	118
Wyświetlanie informacji o uwagach . . . . .	120
Dodawanie uwagi . . . . .	122
Aktualizowanie dokumentu. . . . .	124
Zmiana hasła . . . . .	126

**Dodatek E. Transformowanie AFP do HTML . . . . . 129**

Format pliku AFP2HTML.INI . . . . .	129
Opcje usługi AFP2WEB Transform . . . . .	130
Wyświetlanie przekształconych dokumentów . . . . .	131

**Dodatek F. Transformowanie AFP do PDF . . . . . 133**

Określanie pliku AFP2PDF.INI . . . . .	133
Wyświetlanie przekształconych dokumentów . . . . .	134

**Dodatek G. Pliki konfiguracyjne serwera HTTP . . . . . 135**

Serwer HTTP Apache . . . . .	135
WebSphere Application Server . . . . .	136

**Dodatek H. Brak danych wyjściowych HTML . . . . . 137**

Dane wyjściowe ASCII oddzielone separatorami . . . . .	137
Zalogowanie się . . . . .	137
Uwagi . . . . .	138
Kryteria wyszukiwania . . . . .	138
Uwagi . . . . .	138
Lista trafień dokumentów . . . . .	139
Uwagi . . . . .	139
Podgląd adnotacji . . . . .	140
Komunikat o błędzie. . . . .	140
Uwagi . . . . .	140

**Dodatek I. Obsługa języków narodowych . . . . . 141**

**Dodatek J. Narzędzia do określania problemu . . . . . 143**

<b>Uwagi.</b> . . . . .	145
Znaki towarowe i znaki usług . . . . .	146

**Indeks . . . . . 149**

---

# Informacje o publikacji IBM OnDemand for iSeries Common Server Web Enablement Kit: Podręcznik instalowania i konfigurowania

W tej publikacji znajdują się informacje, których można użyć do planowania, instalowania, konfigurowania i korzystania z produktu IBM Content Manager OnDemand for iSeries Wersja 5 Wydanie 4 Common Server (OnDemand) Web Enablement Kit.

---

## Dla kogo przeznaczony jest ten podręcznik

Ten podręcznik jest przeznaczony przede wszystkim dla administratorów systemów, których zadaniem jest implementowanie, instalowanie i obsługa oprogramowania oraz aplikacji OnDemand Web Enablement Kit (ODWEK). Mogą go także używać programiści, którzy chcą zintegrować OnDemand z aplikacjami WWW.

---

## Sposób organizacji tego podręcznika

Podręcznik ten zawiera informacje potrzebne do zainstalowania i skonfigurowania produktu ODWEK oraz zaplanowania dostępu użytkowników do danych systemu IBM Content Manager OnDemand for iSeries Common Server za pomocą przeglądarki WWW. Publikacja ta składa się z następujących sekcji:

- Rozdział 1, “Przegląd”, na stronie 1
- Rozdział 3, “Instalowanie i konfigurowanie serwera HTTP”, na stronie 11
- Rozdział 4, “Konfigurowanie przykładowych aplikacji”, na stronie 43
- Rozdział 5, “Instalowanie przeglądarek WWW”, na stronie 45
- Dodatek A, “Funkcje API CGI”, na stronie 59
- Dodatek B, “Serwlety w języku Java”, na stronie 87
- Dodatek C, “Funkcje API języka Java”, na stronie 89
- Dodatek D, “Podręcznik programowania funkcji API w języku Java”, na stronie 91
- Dodatek E, “Transformowanie AFP do HTML”, na stronie 129
- Dodatek F, “Transformowanie AFP do PDF”, na stronie 133
- Dodatek G, “Pliki konfiguracyjne serwera HTTP”, na stronie 135
- Dodatek H, “Brak danych wyjściowych HTML”, na stronie 137
- Dodatek I, “Obsługa języków narodowych”, na stronie 141
- Dodatek J, “Narzędzia do określania problemu”, na stronie 143

---

## Wymagania wstępne i informacje pokrewne

Jako punkt wyjścia do wyszukiwania informacji technicznych dotyczących systemów iSeries może służyć Centrum informacyjne IBM iSeries.

Dostęp do Centrum informacyjnego można uzyskać na dwa sposoby:

- Za pomocą serwisu WWW: <http://www.ibm.com/eserver/iseries/infocenter>.
- Za pomocą dysków CD-ROM dostarczonych wraz systemem operacyjnym i5/OS:  
*Centrum informacyjne iSeries*, SK3T-5495-04. W pakiecie znajdują się również wersje podręczników do systemu iSeries w formacie PDF, *Centrum informacyjne iSeries: Podręczniki uzupełniające*, SK3T-5496-01, które zastępują bibliotekę elektroniczną na dysku CD-ROM.

Centrum informacyjne zawiera doradców i ważne tematy dotyczące takich zagadnień, jak język Java, protokół TCP/IP, serwery WWW, sieci chronione, środowiska klastrów, komendy języka CL oraz systemowe aplikacyjne interfejsy programistyczne (API). Zawiera ono także odsyłacze do pokrewnej dokumentacji technicznej (IBM Redbooks) oraz do innych serwisów WWW firmy IBM, takich jak strona główna IBM.

## Inne informacje dostępne w sieci WWW

W sieci WWW dostępnych jest więcej informacji dotyczących systemów iSeries. Dostęp do informacji ogólnych można uzyskać ze strony głównej poświęconej systemom iSeries, która znajduje się pod adresem: <http://www-1.ibm.com/servers/eserver/series/>

Dostęp do informacji dotyczących zaawansowanych funkcji systemów iSeries można uzyskać przy użyciu serwisu Technical Studio, pod adresem: <http://www.iseries.ibm.com/tstudio/>

Użytkownicy z całego świata mogą uzyskać informacje na temat poprawek PTF dotyczących systemów iSeries, zamawiać je i uzyskiwać przez Internet. Internetowe poprawki PTF dla systemów iSeries (jako pakiety do pobrania) oraz informacje PSP (Preventive Service Planning) są dostępne w następującym serwisie internetowym: <http://as400service.ibm.com>

Dokumentacja produktu, która została przeniesiona ze strony bibliotek na stronę wsparcia w serwisie WWW produktu CM OnDemand for iSeries. Aby wyświetlić listę dokumentacji dostępnej dla produktu OnDemand for iSeries, należy wybrać adres <http://www.ibm.com/software/data/ondemand/400/support.html>. Znajduje się ona w lewej kolumnie, w sekcji "Self help", pod nagłówkiem "Learn".

## Centrum informacyjne OnDemand

Poza Centrum informacyjnym iSeries (wymienionym wcześniej) zaleca się odwiedzenie Centrum informacyjnego OnDemand, zawierającego informacje dotyczące CM OnDemand. Centrum informacyjne OnDemand zapewnia szybki, scentralizowany dostęp elektroniczny do informacji o produkcie. Jest to repozytorium dokumentacji stworzone w oparciu o zadania pozwalające na przeszukiwanie całej bazy danych produktów w celu odnalezienia komend, kodów błędów i innych tematów. Użytkownik może tworzyć zakładki do interesujących i często przeglądanych stron, aby uprościć wyszukiwanie informacji w przyszłości.

Centrum informacyjne OnDemand jest dostępne pod adresem WWW <http://publib.boulder.ibm.com/infocenter/cmod83/index.jsp>

## Informacje o dostępności OnDemand

Kompletne informacje dotyczące opcji dostępności obsługiwanych przez ten produkt znajdują się w przewodniku *IBM Content Manager OnDemand for iSeries Common Server Administration Guide*

## iSeries Navigator

IBM iSeries Navigator to rozbudowany graficzny interfejs do zarządzania serwerami iSeries. Funkcje programu iSeries Navigator obejmują poruszanie się po systemie, jego konfigurowanie, planowanie i pomoc elektroniczną z instrukcjami dotyczącymi poszczególnych zadań. Program iSeries Navigator ułatwia obsługę i administrowanie serwerem oraz zwiększa jego efektywność. Jest to także jedyny interfejs użytkownika z dostępem do nowych, zaawansowanych funkcji systemu i5/OS. Częścią programu jest Centrum Zarządzania, które służy do zarządzania wieloma serwerami z systemu centralnego.

Więcej informacji na temat programu iSeries Navigator można znaleźć w Centrum informacyjnym IBM iSeries oraz w następującym serwisie WWW: <http://www.ibm.com/eserver/series/navigator/>



---

## Jak wysyłać uwagi

Opinie użytkowników są niezbędne w celu udostępnienia jak najdokładniejszej, wysokiej jakości dokumentacji technicznej. Mile widziane są komentarze dotyczące tej publikacji, jak również innych publikacji dotyczących produktu OnDemand. W celu wysłania uwag można użyć jednej z następujących metod:

- Przesłać komentarze faksem (numer faksu w Stanach Zjednoczonych, Kanadzie i Puerto Rico: 1-800-937-3430, z innych państw: 1-507-253-5192).
- Przesłać komentarze pocztą elektroniczną pod adresem: [RCHCLERK@us.ibm.com](mailto:RCHCLERK@us.ibm.com); komentarze dotyczące Centrum informacyjnego należy przesyłać pod adresem: [RCHINFO@us.ibm.com](mailto:RCHINFO@us.ibm.com)

W komentarzu należy zamieścić:

- tytuł podręcznika lub temat Centrum informacyjnego iSeries,
- numer publikacji w przypadku książki (znajduje się w prawym dolnym rogu pierwszej strony okładki),
- numer strony lub tytuł tematu, którego dotyczy komentarz.



---

## Podsumowanie zmian

Niniejsze wydanie podręcznika *IBM Content Manager OnDemand for iSeries Common Server OnDemand Web Enablement Kit: Instalowanie i konfigurowanie* zawiera nowe informacje techniczne. Niekiedy może się zdarzyć, że zmienione informacje nie zostały na stronach oznaczone pionowymi kreskami. Poważne zmiany, na które należy zwrócić uwagę, to:

- Te publikacje dotyczące produktu Content Manager OnDemand for iSeries, które mają zastosowanie do produktów Spool File Archive, Object Archive, Record Archive i AnyStore, nie zostały zaktualizowane do Wersji 5 Wydania 4. Oprogramowanie to jest obecnie bardzo stabilne i nie zawiera większych zmian w porównaniu z Wersją 5 Wydaniem 3. W razie potrzeby zaleca się skorzystanie z publikacji *Content Manager OnDemand for iSeries Installation Guide* i *Content Manager OnDemand for iSeries Administration Guide* dla Wersji 5 Wydania 3. Wszystkie publikacje dotyczące produktu Common Server zostały zaktualizowane dla Wersji 5 Wydania 4.
- W znacznym stopniu rozszerzono dokumentację dotyczącą narzędzi migracji ze środowiska Spool File Archive do środowiska Common Server w Dodatku A publikacji *Content Manager OnDemand for iSeries Common Server: Podręcznik planowania i instalowania*.
- Nową możliwością narzędzi migracji ze środowiska Spool File Archive do środowiska Common Server jest redukcja liczby protokołów zadania generowanych podczas pomyślnej pracy programu migracji. Więcej informacji znajduje się w Dodatku A publikacji *Content Manager OnDemand for iSeries Common Server: Podręcznik planowania i instalowania*.
- Dodano szczegóły dotyczące aplikacji Disk Storage Manager (DSM) i Archive Storage Manager (ASM). Większość dodanych informacji znajduje się w publikacji *Content Manager OnDemand for iSeries Common Server Administration Guide*.

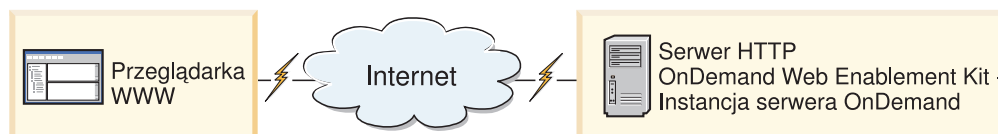


---

## Rozdział 1. Przegląd

Aplikacja ODWEK daje użytkownikom dostęp do danych przechowywanych w serwerze IBM Content Manager OnDemand za pomocą przeglądarki WWW lub programu napisanego przez użytkownika. Pozwala to na przykład udostępnić niektórym osobom adres URL strony WWW, która pozwoli im zalogować się do serwera OnDemand; innym osobom można udostępnić adres URL strony WWW, która pozwoli im wyszukać określony folder. ODWEK sprawdza, czy informacje o użytkowniku, takie jak uprawnienia dostępu do serwera i danych przechowywanych w grupie aplikacji, są poprawne na serwerze OnDemand. Po wprowadzeniu przez użytkownika żądania wyszukania pakiet ODWEK wyświetli stronę WWW z listą dokumentów odpowiadających zapytaniu. Użytkownik wybiera z tej listy dokument do wyświetlenia, a pakiet ODWEK wysyła ten dokument do przeglądarki.

Rys. 1 przedstawia stację roboczą z przeglądarką WWW, z której uzyskiwany jest dostęp do danych na serwerze OnDemand.



Rysunek 1. Dostęp do danych przechowywanych na serwerze OnDemand za pomocą aplikacji ODWEK

ODWEK może wyszukiwać dokumenty na serwerach OnDemand z uruchomionym serwerem IBM Content Manager OnDemand for iSeries Common Server wersja 5, IBM Content Manager OnDemand for Multiplatforms wersja 7 oraz IBM Content Manager OnDemand for z/OS and OS/390 wersja 7.1 i pobierać je z nich.

Pakiet ODWEK zawiera kilka komponentów:

- Interfejs programistyczny OnDemand. Korzysta on ze standardowych interfejsów i protokołów systemu OnDemand w celu uzyskania dostępu do danych przechowywanych na serwerze OnDemand. Obsługa pakietu ODWEK nie wymaga instalowania na serwerze OnDemand żadnego dodatkowego kodu. Do sterownia aplikacją ODWEK może służyć jeden z następujących interfejsów programistycznych:
  - Program Common Gateway Interface (CGI). Daje on dostęp do danych OnDemand z przeglądarki WWW. Działa w systemie, na którym uruchomiony jest serwer HTTP (Hypertext Transfer Protocol), na przykład IBM HTTP Server.
  - Serwlet w języku Java. Daje on dostęp do danych OnDemand z przeglądarki WWW. Serwlet działa na serwerze HTTP obsługującym język Java, na którym uruchomiono serwer aplikacji Java, na przykład IBM WebSphere Application Server.
  - Funkcje API języka Java. Zapewniają one dostęp do danych OnDemand z poziomu programu napisanego przez użytkownika. Funkcje API języka Java wymagają zainstalowania wersji 1.2.2 języka Java lub nowszej.
- Przeglądarka IBM OnDemand Advanced Function Presentation (AFP) Web Viewer. Przeglądarka AFP Web Viewer umożliwia użytkownikom wyszukiwanie dokumentów AFP, wczytywanie ich, wyświetlanie i nawigację w tych dokumentach przy użyciu przeglądarki WWW.

- Przeglądarka IBM OnDemand Image Web Viewer. Przeglądarka Image Web Viewer umożliwia użytkownikom wyszukiwanie dokumentów w formatach BMP, GIF, JPEG, PCX i TIFF, wczytywanie ich, wyświetlanie, drukowanie i nawigację w tych dokumentach przy użyciu przeglądarki WWW.
- Aplet Line Data w języku Java. Aplet Line Data umożliwia użytkownikom wyświetlanie danych dla drukarki wierszowej w oknie przeglądarki WWW. Administrator umożliwia wykorzystanie tego apletu, konfigurując plik ARSWWW.INI.
- Aplet AFP2HTML w języku Java. Aplet AFP2HTML umożliwia użytkownikom wyświetlanie danych wyjściowych wygenerowanych przez usługę IBM AFP2WEB Transform. Usługa AFP2WEB Transform przekształca dokumenty i zasoby AFP w pliki HTML, które można wyświetlić przy użyciu apletu AFP2HTML. Po zainstalowaniu i skonfigurowaniu apletu AFP2WEB Transform administrator umożliwia jego użycie, konfigurując plik ARSWWW.INI.

**Uwaga:** Aby wyświetlać inne rodzaje dokumentów przechowywanych w systemie OnDemand, należy uzyskać i zainstalować odpowiednie przeglądarki. Na przykład do wyświetlania dokumentów w formacie Adobe Portable Data Format (PDF) IBM zaleca użycie modułu dodatkowego Adobe Acrobat Viewer do przeglądarki WWW używanej w danej organizacji.

---

## Informacje o interfejsach programistycznych

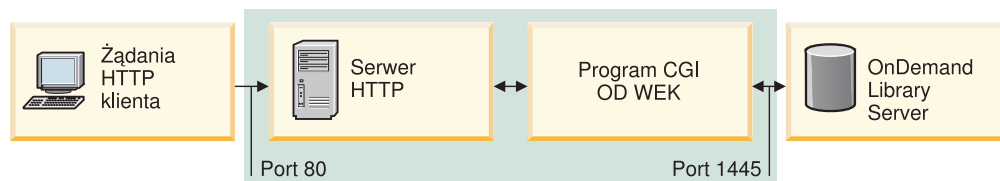
*Instancja* aplikacji ODWEK jest kodem tej aplikacji, który uzyskuje dostęp do danych na serwerze OnDemand. Instancja określa, co można zrobić z danymi, i zarządza zasobami systemowymi przypisanymi do tych danych. Każda instancja jest całym środowiskiem. Ma ona własny plik ASWWW.INI oraz interfejs programistyczny ODWEK, do którego inne instancje nie mają dostępu. Istnieją trzy interfejsy programistyczne ODWEK:

- Program CGI będący interfejsem pomiędzy przeglądarką WWW a serwerem OnDemand.
- Serwlet w języku Java będący interfejsem pomiędzy przeglądarką WWW a serwerem OnDemand.
- Funkcje API języka Java, zbiór metod, których można użyć do korzystania z danych OnDemand za pomocą programu napisanego przez użytkownika.

Należy mieć świadomość, że instancja może korzystać tylko z jednego interfejsu programistycznego. Interfejsy programistyczne wykluczają się wzajemnie. Jedna instancja nie może używać kilku z nich jednocześnie. Można jednak uruchomić wiele instancji aplikacji ODWEK na jednej maszynie, tak aby każda z nich używała innego interfejsu programistycznego. Należy w tym celu skonfigurować poszczególne instancje tak, aby używały innego numeru portu.

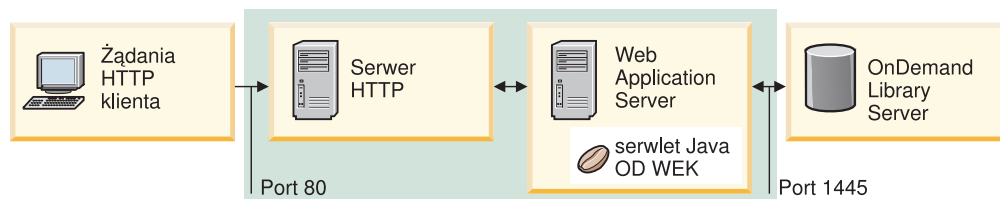
Najczęściej stosowaną implementacją pakietu ODWEK jest jedna instancja w systemie. Konfiguracja z pojedynczą instancją jest typowa dla przetwarzania programistycznego lub autonomicznego środowiska produkcyjnego, w których jedna instancja serwera aplikacji działa niezależnie od innych aplikacji.

Na Rys. 2 na stronie 3 przedstawiono przykład pojedynczej instancji używającej interfejsu CGI.



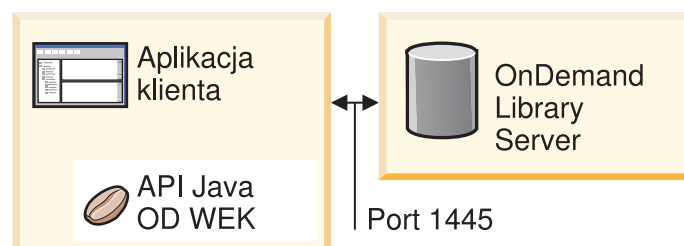
Rysunek 2. Pojedyncza instancja używająca interfejsu CGI

Na Rys. 3 przedstawiono przykład pojedynczej instancji używającej interfejsu serwletu w języku Java.



Rysunek 3. Pojedyncza instancja używająca interfejsu serwletu w języku Java

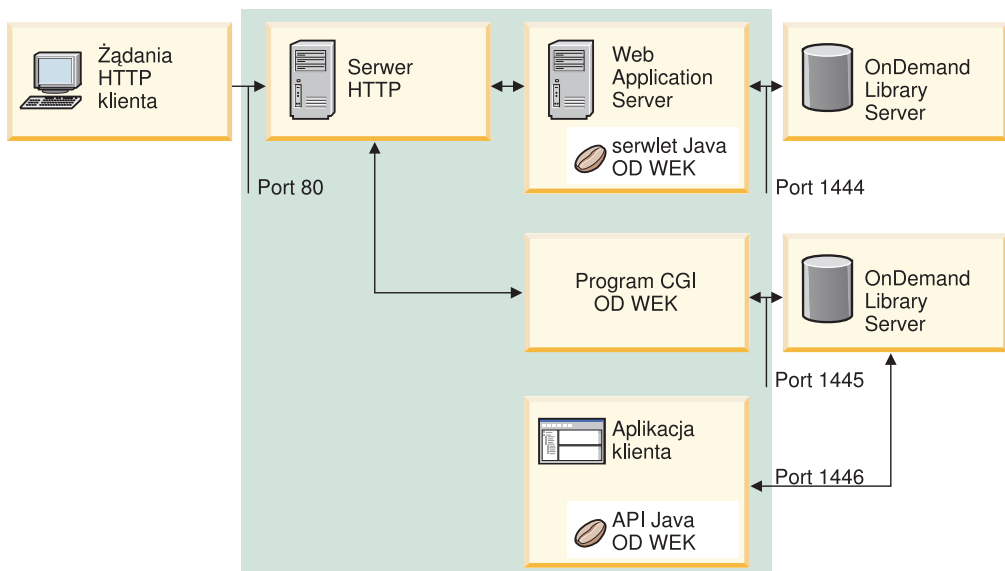
Na Rys. 4 przedstawiono przykład pojedynczej instancji używającej interfejsu API w języku Java.



Rysunek 4. Pojedyncza instancja używająca interfejsu API w języku Java

W danym systemie można skonfigurować wiele instancji pakietu ODWEK. Każda instancja wymaga odrębnego interfejsu programistycznego i pliku ARSWWW.INI określającego unikalny numer portu, przez który ma odbywać się komunikacja pomiędzy interfejsem a serwerem OnDemand. Każda instancja wymaga także oddzielnej pamięci i ochrony. Konfiguracja z wieloma instancjami jest zazwyczaj używana przez klientów, którzy w jednym systemie muszą uruchamiać wiele aplikacji w celach programistycznych, testowych lub produkcyjnych. Instancje działają niezależnie od siebie.

Na Rys. 5 na stronie 4 przedstawiono przykład topologii z wieloma instancjami.



Rysunek 5. Topologia z wieloma instancjami

## Informacje o przeglądarkach

Pakiet ODWEK udostępnia następujące przeglądarki:

- AFP Web Viewer
- Image Web Viewer
- Aplet Line Data w języku Java
- Aplet AFP2HTML w języku Java

Przeglądarki AFP Web Viewer i Image Web Viewer są programami rozszerzającymi możliwości przeglądarki WWW w określony sposób. Przeglądarka AFP Web Viewer umożliwia wyświetlanie dokumentów AFP. Przeglądarka Image Viewer umożliwia wyświetlanie dokumentów BMP, GIF, JPEG, PCX i TIFF. Wszystkie przeglądarki dają możliwość wyświetlania dokumentów w oknie przeglądarki WWW. Każda z nich dodaje pasek narzędzi u góry okna. Może on być dodany do paska narzędzi przeglądarki WWW. Pasek narzędzi modułu dodatkowego udostępnia elementy sterujące, które pomagają w pracy z dokumentami. Użytkownicy, którzy zamierzają używać przeglądarek WWW do wyświetlania tych dokumentów, muszą zainstalować na swoich stacjach roboczych odpowiednie przeglądarki.

**Uwaga:** Program instalacyjny instaluje przeglądarki jako moduły dodatkowe lub kontrolki ActiveX. Jeśli na stacji roboczej zainstalowana jest przeglądarka Internet Explorer, to program instalacyjny zainstaluje kontrolki ActiveX, natomiast jeśli zainstalowana jest przeglądarka Netscape - moduły dodatkowe do niej. Jeśli na stacji roboczej zainstalowane są obie przeglądarki, program instalacyjny zainstaluje kontrolki ActiveX dla programu Internet Explorer i moduły dodatkowe do programu Netscape.

Aplet Line Data umożliwia wyświetlanie dokumentów SCS i danych dla drukarki wierszowej przechowywanych w serwerze OnDemand. Aplet Line Data wyświetla dane dla drukarki wierszowej w oknie przeglądarki i u góry okna dodaje pasek narzędzi. Pasek narzędzi apletu Line Data udostępnia elementy sterujące, które pomagają w pracy z dokumentami. Administrator umożliwia wykorzystanie tego apletu, konfigurując plik ARSWWW.INI.



Aplet AFP2HTML umożliwia użytkownikom wyświetlanie danych wyjściowych wygenerowanych przez usługę IBM AFP2WEB Transform. Usługa AFP2WEB Transform przekształca dokumenty i zasoby AFP w dokumenty HTML. Po zainstalowaniu i skonfigurowaniu apletu AFP2WEB Transform administrator umożliwia jego użycie, konfigurując plik ARSWWW.INI. Aplet AFP2HTML udostępnia pasek narzędzi z elementami sterującymi, które umożliwiają pracę z dokumentami, między innymi obsługę dużych obiektów.

Zaletą apletów jest to, że w przeciwieństwie do przeglądarek użytkownicy, aby ich używać, nie muszą instalować ani aktualizować oprogramowania na stacji roboczej. Ponadto, jeśli IBM udostępni nową wersję przeglądarki, konieczna jest jej dystrybucja wśród użytkowników.

Dzięki użyciu apletów i przeglądarek dostarczonych przez IBM dokumenty wczytywane z serwera OnDemand pozostają skompresowane, dopóki nie dotrą do klienta. Klient dekompresuje je i wyświetla ich strony w oknie przeglądarki WWW. Jeśli dokument był przechowywany w serwerze OnDemand jako duży obiekt, to w miarę, jak użytkownik porusza się po stronach dokumentu, klient wczytuje i dekompresuje jego segmenty.

---

## Korzystanie z pakietu ODWEK

Najbardziej typowym sposobem korzystania z pakietu ODWEK jest dostosowywanie przykładowych aplikacji HTML dostarczanych razem z nim. Przykładowa aplikacja LOGON.HTM obsługuje użytkowników, którzy mają dostęp do określonych folderów. Stronę LOGON.HTM należy najpierw zmodyfikować, aby zawierała informacje o serwerze OnDemand. Następnie należy opublikować adres URL pliku LOGON.HTM. Wtedy użytkownicy będą mogli korzystać z tego adresu w celu zalogowania się do określonego serwera. Pakiet ODWEK automatycznie wyświetla serię stron WWW, na których użytkownicy mogą wyszukiwać, wczytywać i wyświetlać dokumenty OnDemand. Przykładowa aplikacja CREDIT.HTM obsługuje doraźne korzystanie z serwera OnDemand, wyświetlając stronę WWW zawierającą kryteria wyszukiwania w określonym folderze. Po dostosowaniu tej aplikacji użytkownik wpisuje adres URL, wypełnia kryteria wyszukiwania i naciska przycisk Wyślij. Aplikacja ODWEK wyświetla stronę WWW, która zawiera listę dokumentów pasujących do zapytania.

**Ważne:** ODWEK wymaga, aby na komputerze klienta możliwe było zapisywanie danych cookie. Należy upewnić się, że użytkownicy skonfigurowali swoje przeglądarki tak, aby akceptowały informacje cookie.

Większość klientów definiuje jeden identyfikator użytkownika OnDemand w celu korzystania z serwera z aplikacją ODWEK. Ten sposób najczęściej jest stosowany w środowiskach z wieloma przypadkowymi użytkownikami serwera OnDemand, którzy korzystają z tego samego folderu. Można także nadać każdemu użytkownikowi jego własny identyfikator użytkownika na serwerze OnDemand. Niezależnie od wybranego sposobu dostępu do serwera OnDemand z ODWEK identyfikatorami należy zarządzać na serwerze OnDemand: trzeba dodać użytkowników do serwera i określić dla nich uprawnienia do grupy aplikacji i folderów.

---

## Funkcje produktu

Pakiet ODWEK obsługuje opisane poniżej funkcje serwera OnDemand. Zazwyczaj wywołuje się je, tworząc strony WWW z odsyłaczami do programu serwera ODWEK. Każdy odsyłacz wywołuje określoną funkcję. Dane wyjściowe tej funkcji są kolejną stroną WWW, która zawiera odsyłacze kierujące użytkownika do następnej logicznie funkcji. Na przykład początkowa strona WWW może wywoływać funkcję zalogowania się. Funkcja zalogowania

się generuje stronę WWW z odsyłaczem do funkcji określającej kryteria wyszukiwania. Każdą z tych funkcji można wywołać za pomocą funkcji API. Szczegóły zawiera Dodatek A, "Funkcje API CGI", na stronie 59.

## Dodanie adnotacji

Funkcja Dodanie adnotacji umożliwia dodawanie adnotacji do określonego dokumentu. Wyświetlona później adnotacja zawierać będzie, poza samym tekstem, dane na temat czasu i daty utworzenia oraz użytkownika, który utworzył adnotację. W tym celu użytkownik musi mieć uprawnienie do dodawania adnotacji dla każdej grupy aplikacji zawierającej dokumenty, do których ma zostać dodana adnotacja. (Uprawnienie dostępu do grupy aplikacji umożliwia dodawanie adnotacji).

## Zmiana hasła

Funkcja Zmiana hasła umożliwia użytkownikom zmienianie swojego hasła do serwera OnDemand.

## Lista trafień dokumentów

Funkcja Lista trafień dokumentów tworzy listę pozycji, które pasują do kryteriów wyszukiwania. Jest ona wyświetlana w postaci tabeli HTML. Każda pozycja odpowiadająca kryteriom wyszukiwania jest umieszczana w komórce tabeli i zawiera odsyłacz do funkcji Wczytanie dokumentu.

**Ważna uwaga dla klientów, w których systemie działają środowiska OnDemand Spool File Archive i Common Server i którzy używają pozycji ARS\_MIGR\_SERVER w pliku ARS.CFG w celu łączenia folderów Spool File Archive i Common Server w jednej liście wyboru folderu:** Dla użytkowników pakietu ODWEK foldery Spool File Archive będą wyświetlane na liście folderów ODWEK i możliwe będzie wyszukiwanie w nich. Jednakże próba wczytania dokumentu nie powiedzie się.

## Wylogowanie się

Funkcja wylogowania się umożliwia użytkownikom wylogowanie się z serwera OnDemand.

## Zalogowanie się

Funkcja zalogowania się umożliwia użytkownikom zalogowanie się do serwera OnDemand. Jeśli zalogowanie powiedzie się, zostanie wyświetlona strona WWW zawierająca listę folderów, do otwarcia których użytkownik ma uprawnienia.

## Wczytanie dokumentu

Funkcja Wczytanie dokumentu wczytuje dokument z serwera OnDemand. Zwrócony przez serwer strumień danych zawiera dokument i, w zależności od typu danych, zasoby wymagane do jego wyświetlenia. Strumień danych nie może zostać w żaden sposób zmodyfikowany. Przeglądarka interpretuje i dekoduje strumień danych, a następnie wyświetla dokument. Jeśli dokument jest przechowywany w serwerze OnDemand jako duży obiekt, to serwer zwraca tylko pierwszy segment dokumentu. Kolejne segmenty są wczytywane i wyświetlane w miarę potrzeby.

## Kryteria wyszukiwania

Po udanym zalogowaniu się wyświetlana jest lista folderów, które użytkownik ma prawo otworzyć. Następnie użytkownik wybiera folder do otwarcia. Po otwarciu folderu wyświetlana jest strona WWW zawierająca pola wyszukiwania w folderze. Użytkownik może

zaakceptować domyślne kryteria wyszukiwania lub wprowadzić nowe, aby wyszukać określony dokument. Po naciśnięciu przycisku Wyślij żądanie wyszukiwania jest wysyłane do serwera OnDemand.

## Serwerowy wydruk dokumentu

Funkcja Serwerowy wydruk dokumentu wysyła kopie dokumentu do drukarki serwera OnDemand. Aby użyć wydruku serwerowego, użytkownik musi mieć uprawnienie do drukowania dokumentu dla każdej grupy aplikacji, w której znajduje się żądany dokument. (Uprawnienie dostępu do grupy aplikacji umożliwia drukowanie dokumentów). Na serwerze OnDemand musi być zdefiniowana przynajmniej jedna drukarka serwerowa.

## Aktualizowanie dokumentu

Funkcja Aktualizowanie dokumentu umożliwia użytkownikom aktualizowanie bazy danych. Powoduje ona aktualizację jednego lub wielu pól bazy danych dla podanego dokumentu.

## Podgląd adnotacji

Funkcja Podgląd adnotacji umożliwia wyświetlanie adnotacji powiązanych z określonym dokumentem. Aby wyświetlić adnotacje, użytkownik musi mieć uprawnienie do podglądu adnotacji dla każdej grupy aplikacji, w której znajduje się dokument zawierający adnotacje, które użytkownik chce wyświetlić. (Uprawnienie dostępu do grupy aplikacji umożliwia wyświetlanie adnotacji).

---

## Ochrona serwera i danych

Aby korzystać z pakietu ODWEK, należy mieć na uwadze dwa poziomy ochrony:

- Kto może mieć dostęp do programów ODWEK i stron WWW.
- Kto może mieć dostęp do danych na serwerze OnDemand.

Każdy użytkownik, który ma dostęp do serwera HTTP oraz programów i stron WWW, które stanowią frontową część pakietu ODWEK, może potencjalnie dostać się do danych przechowywanych w serwerze OnDemand. IBM zdecydowanie zaleca ograniczenie dostępu do programów i stron WWW. Istnieje wiele sposobów na ograniczenie dostępu do programów i stron WWW na serwerze serwerze HTTP. Na przykład wiele serwerów HTTP zapewnia system ochrony newralgicznych stron WWW, umożliwiając ograniczenie dostępu do katalogów. Można użyć pliku haseł do serwera HTTP, który wymaga, aby użytkownicy w celu wyświetlenia stron WWW podali identyfikator użytkownika i hasło. Mimo podobieństwa identyfikatorów użytkowników i haseł serwera HTTP do identyfikatorów i haseł użytkowników systemu, nie ma pomiędzy nimi żadnej zależności. Nie ma także zależności pomiędzy identyfikatorami i hasłami użytkowników serwera HTTP a identyfikatorami i hasłami użytkowników serwera OnDemand.

ODWEK zapewnia dostęp do serwerów i danych OnDemand za pomocą standardowych funkcji API serwera OnDemand. Funkcje te sprawdzają, czy dany użytkownik serwera OnDemand ma dostęp do tego serwera i żądanych przez tego użytkownika danych. Wyznaczona osoba w organizacji musi administrować ochroną użytkowników i danych na serwerze OnDemand.

Istnieje jeszcze jeden szczegół związany z ochroną, który należy wziąć pod uwagę: metoda użyta do przesyłania parametrów i wartości formularza pomiędzy klientem i serwerem. Formularze dostarczane w pakiecie ODWEK używają metody POST do przesyłania parametrów i wartości w treści żądania HTTP. Metoda POST nie powoduje wyświetlenia parametrów i wartości w polu adresu w przeglądarce. Na przykład typowe wywołanie funkcji wygląda następująco:

```
http://www.firma.com/cgi-bin/arswww.cgi
```

Jednakże, jeśli użytkownik podczas tworzenia formularza nie określi metody, to domyślną metodą będzie GET, która powoduje przekazanie parametrów i wartości w samym adresie URL. W przypadku metody GET typowe wywołanie funkcji wygląda następująco:

```
http://www.firma.com/cgi-bin/arswww.cgi?_function=logon  
&_user=jan&_password=tajemnica
```

W polu adresu w przeglądarce WWW wyświetlane są wtedy parametry i ich wartości jako zwykły tekst. IBM zaleca, aby tworząc własne formularze używać metody POST. Aby zmienić domyślną metodę z GET na POST, należy określić atrybut METHOD znacznika FORM.

**Uwaga:** Używając metody GET, można zaszyfrować parametry i ich wartości za pomocą parametru ENCRYPTURL w pliku ARSWWW.INI. Więcej informacji znajduje się w sekcji “ENCRYPTURL” na stronie 34.

---

## Rozdział 2. Lista kontrolna instalacji

Konfigurowanie zestawu ODWEK wymaga zwykle wykonania następujących czynności:

1. Skontaktuj się z centrum wsparcia IBM, aby uzyskać najnowsze poprawki PTF dla serwera OnDemand. Lista najnowszych poprawek PTC znajduje się w raporcie APAR informacyjnym nr II14053.
2. Skontaktuj się z centrum wsparcia IBM, aby uzyskać najnowsze poprawki PTF dla grupy serwerów. Numer produktu serwera HTTP to 5722-DG1. Aby wyświetlić listę najnowszych poprawek PTF serwera HTTP, należy wybrać stronę [http://www-912.ibm.com/s\\_dir/sline003.NSF/GroupPTFs?OpenView&view](http://www-912.ibm.com/s_dir/sline003.NSF/GroupPTFs?OpenView&view) i kliknąć odpowiednią grupę PTF.
3. Skontaktuj się z centrum wsparcia IBM, aby uzyskać najnowsze poprawki PTF dla grupy baz danych. Aby wyświetlić listę najnowszych poprawek PTF dla DB2, należy wybrać stronę [http://www-912.ibm.com/s\\_dir/sline003.NSF/GroupPTFs?OpenView&view](http://www-912.ibm.com/s_dir/sline003.NSF/GroupPTFs?OpenView&view) i kliknąć odpowiednią grupę PTF.
4. Najnowsza wersja dokumentu *Przeczytaj zanim zaczniesz*, dotyczącego systemu OnDemand, znajduje się pod adresem <http://www.ibm.com/software/data/ondemand/400/support.html>. Zamieszczona jest pod nagłówkiem 'Learn' wraz z pozostałą dokumentacją Wersji 5 Wydania 4. Przed rozpoczęciem instalacji należy wydrukować i zapoznać się z całą zawartością zbioru.
5. Sprawdź wymagania wstępne ODWEK. Patrz Rozdział 3, "Instalowanie i konfigurowanie serwera HTTP", na stronie 11
6. Zainstaluj oprogramowanie OnDemand na serwerze iSeries. Patrz "Instalowanie w systemie i5/OS" na stronie 12
7. Skonfiguruj zbiór ARSWWW.INI. Patrz "Określanie pliku ARSWWW.INI" na stronie 13
8. Skonfiguruj serwer HTTP Apache. Dodatek G dokumentu Dodatek G, "Pliki konfiguracyjne serwera HTTP", na stronie 135 zawiera przykład zbioru konfiguracyjnego serwera HTTP.
9. Jeśli to konieczne, zaktualizuj listy uwierzytelniania QONDADM i QRDARS400. Patrz rozdział 3, "Inne wymagania" na stronie 11
10. Skonfiguruj przeglądarkę na swojej stacji roboczej. Na etap ten składają się następujące czynności:
  - a. Pobierz i zainstaluj odpowiednie zbiory modułów dodatkowych przeglądarki. Patrz Rozdział 5, "Instalowanie przeglądarek WWW", na stronie 45
  - b. Aby móc korzystać z najnowszej wersji apletu wyświetlania danych drukarki, pobierz i zainstaluj moduł środowiska wykonawczego Java ze strony <http://www.java.com>.
  - c. Upewnij się, że przeglądarka akceptuje wszystkie informacje cookie. Wybierz Narzędzia > Opcje internetowe i zakładkę Prywatne.
  - d. Upewnij się, że dla przeglądarki Internet Explorer wybrana jest opcja UTF-8. Wybierz Narzędzia > Opcje internetowe i w zakładce Zaawansowane zaznacz opcję "Zawsze wysyłaj adresy URL jako UTF-8".
  - e. Upewnij się, że środowisko wykonawcze programów Java jest włączone. Wybierz Narzędzia > Opcje internetowe i w zakładce Zaawansowane przejdź do sekcji Java (Sun). Upewnij się, że zaznaczona jest opcja "Używaj Java n vx.y.x dla <applet> (wymaga restartu)". Wersja wyświetlona w panelu odzwierciedla wersję zainstalowaną obecnie dla przeglądarki.



---

## Rozdział 3. Instalowanie i konfigurowanie serwera HTTP

W tej sekcji określono wymagania oraz sposób instalowania oprogramowania ODWEK na serwerze HTTP oraz modyfikowania pliku konfiguracyjnego pakietu ODWEK.

Oprogramowanie ODWEK należy zainstalować w systemie iSeries z aktualną wersją serwera IBM HTTP Server. Ponadto, jeśli ma być używany serwlet w języku Java, należy upewnić się, że posiadana wersja serwera iSeries Web Application Server (WebSphere) jest aktualna oraz że działa prawidłowo.

Aplikacja ODWEK wyszukuje dokumenty na serwerach OnDemand z serwerem IBM Content Manager OnDemand for iSeries Wersja 5 Wydanie 4 Common Server i wczytuje je z nich.

---

### Wymagania instalacyjne

Pakiet ODWEK wymaga:

- Bieżącej wersji serwera IBM HTTP Apache (5722-DG1). Ponadto:
  - Jeśli ma być używany serwlet w języku Java, należy upewnić się, że posiadana wersja serwera iSeries Web Application Server (WebSphere) jest aktualna oraz że działa prawidłowo. Serwery muszą działać w systemie i5/OS Wersja 5 Wydanie 4 lub nowszym.
  - W konfiguracji serwera Apache HTTP należy dodać/zmienić następujące wiersze:
    - DefaultFsCCSID ffff , gdzie w ffff należy podać identyfikator CCSID serwera OnDemand
    - DefaultNetCCSID 01208
    - CGIConvMode EBCDIC dla identyfikatorów CCSID bez znaków DBCS \*\* LUB \*\*
    - CGIConvMode EBCDIC\_JCD dla identyfikatorów CCSID ze znakami DBCS
  - Po wprowadzeniu tych zmian serwer Apache, na którym uruchomiony jest ten program, należy zamknąć i uruchomić ponownie.
- Odpowiedniego typu nośnika instalacyjnego.
- Wystarczającej ilości miejsca na dysku dla plików instalacyjnych: około 30 MB na serwerze HTTP.
- Wystarczającej ilości miejsca na dysku dla dyskowej pamięci podręcznej: domyślnie 10 MB na serwerze HTTP. Więcej informacji znajduje się w sekcji “CACHESIZE” na stronie 17.

---

### Inne wymagania

Aplikacja ODWEK może przechowywać w *pamięci podręcznej* (tymczasowo) dokumenty na serwerze HTTP. Zwiększa to szybkość przesyłania użytkownikom wyświetlanych przez nich poprzednio dokumentów. Aby włączyć dyskową pamięć podręczną dla dokumentów, należy skonfigurować parametr CACHEDOCS w pliku ARSWWW.INI. Szczegóły znajdują się w sekcji “CACHEDOCS” na stronie 16.

Domyślnie aplikacja ODWEK przechowuje dane w katalogu /QIBM/UserData/OnDemand/WWW/CACHE. Można podać inny katalog pamięci podręcznej, modyfikując plik ARSWWW.INI. Szczegóły znajdują się w sekcji “CACHEDIR” na stronie 16.

Upewnij się, że procesy, które uruchamiają programy ODWEK, mogą odczytywać katalog zawierający programy i zapisywać pliki w katalogu, w którym znajduje się pamięć podręczna. Po zainstalowaniu pakietu ODWEK wszystkie obiekty są chronione za pomocą listy autoryzacji QONDADM, a profile użytkowników: QTMHHTTP, QTMHHTTP1 i QEJBSVR są dodawane do listy autoryzacji z uprawnieniem \*CHANGE. Ponadto profile użytkowników QTMHHTTP, QTMHHTTP1 i QEJBSVR muszą mieć uprawnienie \*USE do listy autoryzacji QRDARS400.

Aplikacja ODWEK wymaga, aby przeglądarka użytkownika końcowego akceptowała format UTF-8. W przeglądarce Internet Explorer firmy Microsoft, należy wybrać Narzędzia/Opcje internetowe, a następnie zakładkę Zaawansowane. W części Przeglądanie należy zaznaczyć opcję Zawsze wysyłać adresy URL jako UTF-8.

Planując korzystanie z apletu AFP2HTML, należy uzyskać usługę AFP2WEB Transform od IBM oraz zainstalować i skonfigurować ją na serwerze HTTP. Więcej informacji na temat oferty AFP2WEB Transform można uzyskać u przedstawiciela IBM. Należy także określić opcje konfiguracyjne dla dokumentów i zasobów AFP, które mają być przekształcane za pomocą usługi AFP2WEB Transform. Więcej informacji na temat pliku konfiguracyjnego zawiera Dodatek E, "Transformowanie AFP do HTML", na stronie 129.

Planując przekształcanie dokumentów AFP przechowywanych w systemie OnDemand w dokumenty PDF, które można wyświetlać w przeglądarce Adobe Acrobat, należy uzyskać usługę AFP2PDF Transform od IBM, zainstalować ją i skonfigurować na serwerze HTTP. Więcej informacji na temat oferty AFP2WEB Transform można uzyskać u przedstawiciela IBM. Należy także określić opcje konfiguracyjne dla dokumentów i zasobów AFP, które mają być przekształcane za pomocą usługi AFP2PDF Transform. Więcej informacji na temat pliku konfiguracyjnego zawiera Dodatek F, "Transformowanie AFP do PDF", na stronie 133. Aby wyświetlić przekształcone dokumenty, należy pobrać i zainstalować moduł dodatkowy Adobe Acrobat viewer dla przeglądarek WWW używanych w organizacji.

---

## Instalowanie w systemie i5/OS

Konfigurowanie produktu ODWEK wymaga zwykle wykonania następujących czynności:

1. Aby zainstalować pakiet ODWEK, należy wykonać instrukcje zawarte w publikacji *IBM Content Manager OnDemand for iSeries Common Server: Podręcznik planowania i instalowania* (SC85-0121). Numer programu licencjonowanego to 5722RD1, a numer opcji - 11.

**Uwaga:** Zalecanym sposobem zainstalowania pakietu ODWEK jest użycie opcji menu Instalowanie programów licencjonowanych (Install Licensed Programs) wybranej z menu Praca z programami licencjonowanymi (Work with Licensed Programs) (go licpgm). Na ekranie Instalowanie programów licencjonowanych (Install Licensed Programs) należy wybrać 1, aby dodać opcję, jako Program licencjonowany wpisać 5722RD1, a jako Opcję produktu - 11, bądź też przewinąć listę programów licencjonowanych i opcji produktów, aż zostanie wyświetlony pakiet ODWEK, i obok jego nazwy wpisać 1. **W przypadku zainstalowania serwera OnDemand w inny sposób podczas próby jego użycia mogą wystąpić błędy.**

2. IBM zaleca zamówienie, załadowanie i zastosowanie wszystkich poprawek PTF dostępnych dla serwera OnDemand po udanym zainstalowaniu programu licencjonowanego. Pełna lista poprawek PTF dla serwera OnDemand Wersja 5 Wydanie 4 znajduje się w dokumencie APAR II13680. Powyższy APAR informacyjny można zamówić elektronicznie przy użyciu komendy SNDPTFORD, podając jako numer poprawki II14053. Ważne jest przeczytanie listów przewodnich dotyczących poprawek PTF i wykonanie wszystkich określonych w nich instrukcji specjalnych.



3. Istotne jest również zainstalowanie wszystkich dostępnych poprawek PTF dla produktu IBM 5722-DG1 (IBM HTTP Server). Bieżące poprawki PTF dla tego produktu są konieczne dla prawidłowego funkcjonowania aplikacji ODWEK.

---

## Dalsze kroki

Należy upewnić się, że w systemie iSeries działa aktualna wersja serwera IBM HTTP server. Serwer HTTP należy skonfigurować. Dodatek G, "Pliki konfiguracyjne serwera HTTP", na stronie 135 zawiera przykład pliku konfiguracyjnego serwera HTTP.

Jeśli ma być używany serwlet w języku Java, należy upewnić się, że posiadana wersja serwera iSeries Web Application Server (WebSphere) jest aktualna oraz że działa prawidłowo. Konieczne będzie skonfigurowanie serwera WebSphere. Odpowiednie instrukcje znajdują się w Centrum dokumentacji serwera IBM WebSphere Application Server w sieci WWW pod adresem [www.ibm.com/servers/eserver/series/software/websphere/wsappserver/](http://www.ibm.com/servers/eserver/series/software/websphere/wsappserver/). W serwisie tym należy kliknąć odsyłacze Installation and Initial Configuration (Instalowanie i wstępne konfigurowanie) dla odpowiedniej wersji serwera WebSphere.

Po zainstalowaniu oprogramowania ODWEK, skonfigurowaniu serwera HTTP i (opcjonalnie) skonfigurowaniu serwera WebSphere można skonfigurować plik inicjowania ODWEK dla używanego środowiska operacyjnego. Patrz sekcja "Określanie pliku ARSWWW.INI".

---

## Określanie pliku ARSWWW.INI

Plik ARSWWW.INI jest plikiem tekstowym ASCII zawierającym parametry odczytywane przez programy ODWEK (na przykład program CGI lub serwlet w języku Java). Każdy parametr podaje się w osobnym wierszu w następującym formacie: **PARAMETR=wartość**. Na przykład:

```
AFPVIEWING=plugin
CACHEDIR=/tmp/cache
LANGUAGE=ENU
```

Parametry w pliku ARSWWW.INI są pogrupowane w sekcje. Początek sekcji określa się za pomocą nagłówka sekcji w następującym formacie: **[nagłówek\_sekcji]**. Parametry w sekcji podaje się po nagłówku. Na przykład:

```
[@SRV@_gunnar]
HOST=gunnar
PORT=1446
PROTOCOL=0
```

Przykładowy plik ARSWWW.INI jest dostarczany razem z produktem. Zawiera on najczęściej używane wartości. Jest on przedstawiony w sekcji "Przykład pliku ARSWWW.INI" na stronie 39.

Sekcje i parametry pliku ARSWWW.INI są następujące:

### **[@SRV@\_DEFAULT]**

Sekcja serwera domyślnego. Służy ona do określania parametrów, które są wspólne dla serwerów OnDemand, z którymi komunikować się będzie aplikacja ODWEK. Podane w niej parametry i wartości będą stosowane o ile nie zostaną określone w sekcji dotyczącej konkretnego serwera.

Sekcja ta ma zasięg globalny obejmujący wszystkie serwery i określa się ją w pliku ARSWWW.INI tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

## **PORT**

Numer portu TCP/IP, którego serwery OnDemand używają do komunikowania się z aplikacją ODWEK. Jeśli parametr PORT nie zostanie określony, serwer użyje numeru portu podanego dla serwera OnDemand w tabeli usług (WRKSRVTBLE). W przypadku, gdy parametr PORT nie zostanie określony, a serwera OnDemand nie ma na liście w tabeli usług, serwery będą próbowały użyć portu 1445. Aby serwery użyły portu podanego w tabeli usług dla serwera OnDemand, należy wpisać tu wartość 0 (zero).

Parametr ten można określić tylko raz w sekcji zawierającej wartości domyślne. Używając funkcji API Logon, można przesłonić numer portu za pomocą parametru `_port`.

Parametr opcjonalny.

Przykład:

```
[@SRV@_DEFAULT]  
PORT=0
```

## **PROTOCOL**

Protokół sieciowy, którego serwery OnDemand używają do komunikowania się z aplikacją ODWEK. Dla protokołu TCP/IP należy podać 0 (zero).

Parametr ten trzeba określić jeden raz w sekcji zawierającej wartości domyślne.

Parametr opcjonalny. Jeśli nie zostanie określony, użyta zostanie wartość 0 (zero).

Przykład:

```
[@SRV@_DEFAULT]  
PROTOCOL=0
```

## **[@SRV@\_serwer]**

Sekcja serwera. Dla każdego serwera OnDemand, z którym będzie komunikować się aplikacja ODWEK, należy określić jedną sekcję serwera. Znajdować się w niej będą parametry i wartości dotyczące konkretnego serwera. Jej nagłówek musi zawierać łańcuch znaków identyfikujący serwer. Parametry podane w sekcji serwera przesłaniają parametry znajdujące się w sekcji serwera domyślnego.

Dla każdego serwera należy określić jedną sekcję.

Sekcja ta jest wymagana.

Może ona zawierać następujące parametry:

## **HOST**

Nazwa serwera OnDemand. Można podać adres TCP/IP, alias nazwy hosta lub pełną nazwę hosta serwera.

Parametr ten trzeba określić jeden raz w sekcji serwera.

Jest to parametr wymagany.

Przykład:

```
[@SRV@_gunnar]  
HOST=gunnar
```

## PORT

Numer portu TCP/IP, którego serwer OnDemand używa do komunikowania się z aplikacją ODWEK. Jeśli nie zostanie określony, serwer użyje numeru portu, który został podany w sekcji serwera domyślnego, bądź domyślnego numeru portu.

Parametr ten można określić jeden raz w sekcji serwera. Używając funkcji API Logon, można przesłonić numer portu za pomocą parametru `_port`.

Parametr opcjonalny.

Przykład:

```
[@SRV@_gunnar]  
PORT=0
```

Numer portu musi odpowiadać numerowi portu określone w pliku `ars.ini` dla danej instancji serwera.

## PROTOCOL

Protokół sieciowy, którego serwer OnDemand używa do komunikowania się z aplikacją ODWEK. Dla protokołu TCP/IP należy podać 0 (zero).

Parametr ten można określić jeden raz w sekcji serwera.

Parametr opcjonalny. Jeśli nie zostanie podany, zostanie użyta wartość określona w sekcji serwera domyślnego.

Przykład:

```
[@SRV@_gunnar]  
PROTOCOL=0
```

## [CONFIGURATION]

Sekcja CONFIGURATION zawiera parametry używane przez aplikację ODWEK na serwerze HTTP.

Sekcja ta ma zasięg globalny i określa się ją w pliku `ARSWWW.INI` tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

### APPLETCACHEDIR

Określa katalog, w którym aplety Line Data i AFP2HTML tymczasowo przechowują dokumenty. Może to być lokalny katalog na stacji roboczej użytkownika lub znajdować się na dysku sieciowym. Wszyscy użytkownicy muszą mieć uprawnienie dostępu do tego katalogu.

Przykład:

```
[Configuration]  
APPLETCACHEDIR=/QIBM/UserData/OnDemand/www/cache
```

#### Uwagi:

1. Parametr APPLETCACHEDIR ma zasięg globalny.
2. Parametr APPLETCACHEDIR jest opcjonalny. Jeśli jednak nie zostanie podany, aplety będą próbować przechowywać dokumenty w katalogu roboczym języka Java.

3. Jeśli podany katalog nie istnieje, aplety również będą próbować przechowywać dokumenty w katalogu roboczym języka Java.
4. Aplet usuwa dokument z katalogu zawierającego pamięć podręczną w momencie, gdy użytkownik opuszcza aplet (na przykład zamykając dokument).

## APPLETDIR

Określa katalog, w którym znajdują się aplety Line Data i AFP2HTML.

### Uwagi:

1. Można podać nazwę katalogu lub wartość AliasMatch:
  - Jeśli zostanie podana nazwa katalogu, będzie ona traktowana jako względna w stosunku do katalogu /QIBM/UserData/OnDemand/WWW. Na przykład, jeśli zostanie określony parametr `appletdir=aplety`, to aplety muszą znajdować się w katalogu /QIBM/UserData/OnDemand/WWW/APLETY .
  - Jeśli zostanie podana wartość AliasMatch, musi ona być zdefiniowana w pliku konfiguracyjnym serwera HTTP. Na przykład, jeśli zostanie podana wartość `appletdir=/aplety/`, to plik konfiguracyjny serwera HTTP musi mieć wartość AliasMatch dla katalogu /aplety/. Zastępująca regułę AliasMatch ścieżka do pliku musi być pełną ścieżką do katalogu na serwerze. Na przykład:  

```
AliasMatch ^/aplety/com/ibm/edmslod/(.*)$ /QIBM/UserData/OnDemand/www/aplety/$1
```

```
AliasMatch ^/aplety/(.*)$ /QIBM/UserData/OnDemand/www/aplety/$1
```
2. Sprawdź uprawnienia do podanego katalogu. Procesy, które uruchamiają programy ODWEK, muszą mieć możliwość odczytywania katalogu z apletami.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Jest to parametr wymagany.

Przykład:

```
[CONFIGURATION]
APPLETDIR=aplety
```

## CACHEDIR

Służy do określania katalogu na serwerze HTTP, w którym aplikacja ODWEK tymczasowo przechowuje dokumenty (*pamięć podręczna*). Patrz sekcja “CACHEDOCS”. Domyślnie aplikacja ODWEK przechowuje dokumenty w katalogu /QIBM/UserData/OnDemand/WWW/CACHE.

**Uwaga:** Sprawdź uprawnienia do podanego katalogu. Procesy, które uruchamiają programy ODWEK, muszą mieć możliwość zapisu do katalogu pamięci podręcznej i odczytu z niego.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny.

Przykład:

```
[CONFIGURATION]
CACHEDIR=/QIBM/UserData/OnDemand/WWW/CACHE
```

## CACHEDOCS

Określa, czy aplikacja ODWEK tymczasowo przechowuje dokumenty w *pamięci podręcznej* serwera HTTP. Pamięć podręczna może przyspieszyć wczytywanie wyświetlanych wcześniej dokumentów z serwera. Wartością domyślną jest 0 (zero), co oznacza, że pamięć podręczna na dokumenty nie jest włączona. Aby ją włączyć, podaj wartość 1 (jeden). Włączając pamięć

podręczną dla dokumentów, sprawdź katalog, w którym aplikacja ODWEK będzie przechowywała dokumenty (patrz sekcja “CACHEDIR” na stronie 16) i ilość miejsca na dysku zarezerwowaną dla pamięci podręcznej (patrz sekcja “CACHESIZE”).

**Uwaga:** IBM zaleca włączenie pamięci podręcznej dla dokumentów zawsze wtedy, gdy używana jest przeglądarka Microsoft Internet Explorer z przeglądarką AFP Web Viewer lub Image Web Viewer.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny. W zasadzie jednak większość klientów powinna w każdej sytuacji skonfigurować pamięć podręczną dla dokumentów.

Przykład:

```
[CONFIGURATION]
CACHEDOCS=1
```

## CACHEMAXTHRESHOLD

Określa, kiedy aplikacja ODWEK powinna zacząć usuwać dane i dokumenty z pamięci podręcznej. Aplikacja ODWEK zaczyna usuwanie danych i dokumentów wtedy, gdy procent zajętego miejsca na dysku będzie większy lub równy podanej wartości. Wartością domyślną jest 80 (osiemdziesiąt procent). Aplikacja ODWEK usuwa najstarsze pozycje w pamięci podręcznej, aż zostanie osiągnięty podany próg (patrz sekcja “CACHEMINTHRESHOLD”).

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny.

Przykład:

```
[CONFIGURATION]
CACHEMAXTHRESHOLD=80
```

## CACHEMINTHRESHOLD

Określa, kiedy aplikacja ODWEK powinna zakończyć usuwanie danych i dokumentów z pamięci podręcznej. Aplikacja ODWEK zakończy usuwanie danych i dokumentów wtedy, gdy procent zajętego miejsca na dysku będzie mniejszy lub równy podanej wartości. Wartością domyślną jest 40 (czterdzieści procent). Aplikacja ODWEK zaczyna usuwanie od najstarszych pozycji w pamięci podręcznej w momencie, gdy zostanie osiągnięty podany próg (patrz sekcja “CACHEMAXTHRESHOLD”).

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny.

Przykład:

```
[CONFIGURATION]
CACHEMINTHRESHOLD=40
```

## CACHESIZE

Ilość miejsca na dysku, jakiej może użyć aplikacja ODWEK w celu tymczasowego przechowania danych i dokumentów w *pamięci podręcznej* na serwerze HTTP. Wartość tę należy podać w megabajtach. Wartością domyślną jest 10 (dziesięć megabajtów).

**Uwaga:** Informacje o tym, jak włączyć pamięć podręczną dla dokumentów, znajdują się w sekcji “CACHEDOCS” na stronie 16.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny. Jednak im więcej przestrzeni na dysku zostanie przydzielonej, tym więcej dokumentów aplikacja ODWEK będzie mogła przechować w serwerze HTTP. Ogólnie rzecz biorąc, zwiększy to szybkość przesyłania wyświetlonych wcześniej dokumentów do użytkowników przez aplikację ODWEK.

Przykład:

```
[CONFIGURATION]
CACHE SIZE=1024
```

## CACHEUSERIDS

Określa oddzielną przecinkami listę identyfikatorów użytkowników OnDemand, dla których aplikacja ODWEK będzie używać danych z dyskowej pamięci podręcznej w celu przeprowadzenia logowania. W przypadku podanych tu użytkowników wielokrotne zalogowanie się spowoduje pominięcie standardowego przetwarzania logowania OnDemand, chyba że w pamięci podręcznej nie będzie wymaganych danych lub zostanie osiągnięty limit czasu nieaktywności (patrz parametry systemowe w serwerze OnDemand). Identyfikatory użytkowników należy oddzielić przecinkami.

### Uwagi:

1. Jeśli na serwerze jest rozróżniana wielkość liter w identyfikatorze użytkownika (patrz parametry systemowe w serwerze OnDemand), to należy podać dokładnie taki identyfikator, jaki został zdefiniowany w serwerze OnDemand.
2. Użytkownicy określani na liście CACHEUSERIDS mają dostęp tylko do tych folderów, których nazwy i inne informacje znajdują się w pamięci podręcznej. Użytkownicy ci nie będą mieli dostępu do folderów utworzonych po zalogowaniu się do serwera OnDemand. Aby użytkownik znajdujący się na liście CACHEUSERIDS miał dostęp do nowego folderu, należy albo usunąć jego nazwę z listy CACHEUSERIDS, albo opróżnić pamięć podręczną.
3. Aby aplikacja ODWEK udostępniała dane z pamięci podręcznej wszystkim użytkownikom OnDemand, należy wpisać CACHEUSERIDS=\*

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny.

Przykład:

```
[CONFIGURATION]
CACHEUSERIDS=użytkownik1,użytkownik2,użytkownik3
```

## CODEPAGE

Określa stronę kodową bazy danych serwera OnDemand. Domyślnie aplikacja ODWEK używa strony kodowej serwera HTTP.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz. Używając funkcji API Logon można przesłonić stronę kodową za pomocą parametru `_codepage`.

Parametr opcjonalny. Jeśli jednak serwer HTTP ma inną stronę kodową niż baza danych, to jego określenie jest konieczne.

Przykład:

```
[CONFIGURATION]
CODEPAGE=37
```

## DOCSIZE

Podczas wczytywania dokumentów z serwera OnDemand parametr ten określa maksymalną wielkość (w bajtach) dokumentu, jaki może być bezpośrednio zapisany w pamięci zamiast uprzedniego zapisywania go na dysk. Każdy dokument, którego wielkość jest mniejsza lub równa podanej wartości, zostanie zapisany bezpośrednio w pamięci. Wszystkie dokumenty większe od podanej wartości zostaną przed wyświetleniem w przeglądarce najpierw zapisane na dysk, a następnie z niego odczytane i zapisane w pamięci. Mniejsza wartość zapewnia mniejsze wykorzystanie zasobów systemowych, natomiast większa zwiększy wydajność wyświetlania. Dopuszczalne wartości zawierają się w zakresie od 0 (zera) do  $n$  bajtów, gdzie  $n$  oznacza ilość dostępnej pamięci w systemie. Wartość zero powoduje użycie domyślnej wielkości 1 MB. Jeśli parametr nie zostanie określony lub jego wartość nie zostanie zdefiniowana lub rozpoznana, system przyjmie domyślną wielkość 1 MB.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny.

Przykład:

```
[CONFIGURATION]
DOCSIZE=524287
```

## IMAGEDIR

Określa katalog, w którym znajdują się pliki obrazów używane przez aplikację ODWEK.

**Uwagi:**

1. ODWEK konkatenuje podaną wartość z nazwami plików znalezionymi w znacznikach HTML, które odwołują się do obrazów. Na przykład po określeniu wartości:

```
imagedir=obrazy
```

Znacznik HTML określający obraz odwołujący do funkcji wyświetlenia dokumentu będzie miał następującą postać:

```
<IMG SRC="obrazy/odc_vd.gif">
```

2. Można podać nazwę katalogu lub wartość AliasMatch:

- Jeśli zostanie podana nazwa katalogu, będzie ona traktowana jako względna w stosunku do katalogu /QIBM/UserData/OnDemand/WWW. Na przykład, jeśli zostanie określony parametr imagedir=obrazy, to obrazy muszą znajdować się w katalogu /QIBM/UserData/OnDemand/WWW/OBRAZY.
- Jeśli zostanie podana reguła AliasMatch, musi ona być zdefiniowana w pliku konfiguracyjnym serwera HTTP. Na przykład, jeśli zostanie podana wartość imagedir=/obrazy/, to plik konfiguracyjny serwera HTTP musi mieć wartość AliasMatch dla katalogu /obrazy/. Reguła AliasMatch musi być pełną ścieżką do katalogu na serwerze. Na przykład:

```
AliasMatch ^/obrazy/(.*)$ /QIBM/UserData/OnDemand/WWW/OBRAZY/$1
```

3. Sprawdź uprawnienia do podanego katalogu. Procesy, które uruchamiają programy ODWEK, muszą mieć możliwość odczytywania katalogu z obrazami.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Jest to parametr wymagany.

Przykład:

```
[CONFIGURATION]
IMAGEDIR=obrazy
```

## LANGUAGE

Określa język, w którym ODWEK wyświetla komunikaty. Językiem domyślnym jest angielski (ENU). ODWEK obsługuje następujące języki:

Wartość	Terytorium
ARA	Egipt
CHS	Chiny
CHT	Tajwan
DAN	Dania
DEU	Niemcy
ENU	U.S.A. / angielski
ESP	Hiszpania
FIN	Finlandia
FRA	Francja
FRC	Kanada
ITA	Włochy
JPN	Japonia
KOR	Korea
NLD	Holandia
NOR	Norwegia
PTB	Brazylia
SVE	Szwecja

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny.

Przykład:

```
[CONFIGURATION]  
LANGUAGE=JPN
```

### ShowSearchString

Określa, czy funkcja automatycznego wyszukiwania jest aktywna, czy nie. Funkcja ta obsługuje wyszukiwanie transakcji i tekstów w dokumentach zawierających dane dla drukarki wierszowej z poziomu przeglądarki Line Data w języku Java. Funkcja automatycznego wyszukiwania automatycznie wyszukuje i wyróżnia te wiersze w dokumencie, które pasują do podanych przez użytkownika kryteriów wyszukiwania.

Jeśli funkcja automatycznego wyszukiwania jest aktywna i użytkownik uruchomi wyszukiwanie transakcji lub tekstów, a następnie z wyświetlonej listy dokumentów otworzy dokument, system automatycznie wyszuka w dokumencie tekst odpowiadający podanym kryteriom wyszukiwania. Jeśli zostanie on znaleziony, to wiersz odpowiadający kryteriom wyszukiwania zostanie wyróżniony; w przeciwnym razie wyświetlony zostanie odpowiedni komunikat. Gdy użytkownik otworzy inny dokument w celu wyświetlenia (lub ponownie otworzy przeglądany wcześniej dokument), wyszukiwanie jest wykonywane ponownie.

Aby aktywować funkcję automatycznego wyszukiwania, należy nadać parametrowi ShowSearchString wartość 1 (jeden). Aby wyłączyć funkcję automatycznego wyszukiwania, należy nadać parametrowi ShowSearchString wartość 0 (zero).



Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny. Jeśli nie zostanie określony, wartością domyślną jest 0 (zero; nieaktywne).

Przykład:

```
[CONFIGURATION]
ShowSearchString=1
```

## TEMPDIR

Służy do określania katalogu, w którym ODWEK przechowuje pliki tymczasowe.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny. Jeśli nie zostanie określony, ODWEK będzie przechowywać pliki tymczasowe w katalogu wykonawczym. Jeśli używany jest program CGI, katalogiem wykonawczym jest katalog, w którym zainstalowano program CGI. Jeśli używany jest serwet, katalogiem wykonawczym jest katalog, w którym znajduje się ten serwet; w niektórych przypadkach jest to katalog, w którym znajduje się plik java.exe, w innych - katalog, w którym znajdują się serwlety. Faktyczne położenie zależy jednak od serwera aplikacji Java.

Przykład:

```
[CONFIGURATION]
TEMPDIR=/QIBM/UserData/OnDemand/WWW/TMP
```

**Uwaga:** Sprawdź uprawnienia do podanego katalogu. Procesy, które uruchamiają programy ODWEK, muszą mieć możliwość zapisu w katalogu tymczasowym i odczytu z niego.

## TEMPLATEDIR

Określa katalog, w którym znajdują się pliki szablonów HTML. Produkt ODWEK używa tych plików w celu generowania stron WWW w odpowiedzi na różne swoje funkcje (na przykład Zalogowanie się, Wyszukiwanie, Wczytanie dokumentu itd.). Domyślnie aplikacja ODWEK wczytuje pliki szablonów z katalogu /QIBM/UserData/OnDemand/WWW/SAMPLES.

**Uwaga:** Sprawdź uprawnienia do podanego katalogu. Procesy, które uruchamiają programy ODWEK, muszą mieć możliwość odczytywania katalogu z szablonami.

Parametr ten ma zasięg globalny i określa się go w sekcji CONFIGURATION tylko raz.

Parametr opcjonalny.

Przykład:

```
[CONFIGURATION]
TEMPLATEDIR=/QIBM/UserData/OnDemand/WWW/SAMPLES
```

## [SECURITY]

Sekcja SECURITY zawiera parametry ochrony używane przez aplikację ODWEK na serwerze HTTP.

Sekcja ta ma zasięg globalny i określa się ją w pliku ARSWWW.INI tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

## REPORTSERVERTIMEOUT

Służy do określenia, że aplikacja ODWEK powinna używać parametru limitu czasu nieaktywności określonego w serwerze OnDemand. Parametr ten określa, kiedy serwer może przerwać sesję z nieaktywnym użytkownikiem. Aby określić, że aplikacja ODWEK ma używać tego parametru, należy parametrowi REPORTSERVERTIMEOUT nadać wartość 1 (jeden).

Parametr ten ma zasięg globalny i określa się go w sekcji SECURITY tylko raz.

Parametr opcjonalny. Jeśli parametr REPORTSERVERTIMEOUT nie zostanie określony, aplikacja ODWEK nie użyje parametru określającego limit czasu nieaktywności. Innymi słowy, aplikacja ODWEK nie przerwie sesji z nieaktywnym użytkownikiem. Więcej informacji dotyczących limitu czasu nieaktywności znajduje się w pomocy elektronicznej klienta administracyjnego.

Przykład:

```
[SECURITY]  
REPORTSERVERTIMEOUT=1
```

## SERVERACCESS

Określa oddzielną przecinkami listę serwerów OnDemand, do których aplikacja ODWEK ma dostęp. Jeśli parametr SERVERACCESS zostanie określony, to klienci używający aplikacji ODWEK i programów, które korzystają z jej funkcji API, mają dostęp tylko do podanych serwerów. Można podać adres TCP/IP, alias nazwy hosta lub pełną nazwę hosta serwera.

Parametr ten ma zasięg globalny i określa się go w sekcji SECURITY tylko raz.

Parametr opcjonalny.

Przykład:

```
[SECURITY]  
SERVERACCESS=dave,gunnar
```

## [AFP2HTML]

Sekcja AFP2HTML zawiera parametry używane przez usługę AFP2WEB Transform. Usługa AFP2WEB Transform przekształca dokumenty i zasoby AFP w dokumenty HTML, które można wyświetlić przy użyciu apletu AFP2HTML.

### Uwagi:

1. Aby przekształcić dokumenty AFP w dokumenty HTML, administrator musi nabyć usługę AFP2WEB Transform od IBM, a następnie zainstalować i skonfigurować ją na serwerze. Więcej informacji na temat oferty AFP2WEB Transform można uzyskać u przedstawiciela IBM. Należy także określić opcje konfiguracyjne usługi AFP2WEB Transform. Więcej informacji na temat pliku konfiguracyjnego zawiera Dodatek E, "Transformowanie AFP do HTML", na stronie 129.
2. Aby przekształcić dokumenty za pomocą usługi AFP2WEB Transform, należy określić parametr AFPVIEWING=HTML w sekcji DEFAULT BROWSER (lub sekcjach dotyczących innych przeglądarek). Szczegóły znajdują się w sekcji "AFPVIEWING" na stronie 32. (Planując użycie funkcji API Retrieve Document (Wczytanie dokumentu), należy określić parametr \_afp=HTML. Szczegóły znajdują się w sekcji "Wczytanie dokumentu" na stronie 77.)

3. Domyślnie aplikacja ODWEK do wyświetlania przekształconych dokumentów używa apletu AFP2HTML. Jeśli przekształcony dokument był przechowywany w serwerze OnDemand jako duży obiekt, to aplet AFP2HTML udostępnia elementy sterujące ułatwiające przechodzenie do dowolnej strony dokumentu.

Sekcja ta ma zasięg globalny i określa się ją w pliku ARSWWW.INI tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

## CONFIGFILE

Plik konfiguracyjny zawierający opcje używane przez usługę AFP2WEB Transform w celu przekształcenia dokumentów i zasobów AFP w dane HTML, czcionki i obrazy, które można wyświetlić za pomocą apletu AFP2HTML. Przykład pliku konfiguracyjnego dostarczanego z serwerem OnDemand zawiera Dodatek E, "Transformowanie AFP do HTML", na stronie 129. Szczegóły dotyczące opcji, jakie można określić w pliku konfiguracyjnym, znajdują się w dokumentacji usługi AFP2WEB Transform.

Parametr ten ma zasięg globalny i określa się go w sekcji AFP2HTML tylko raz.

Parametr opcjonalny.

Przykład:

```
[AFP2HTML]
CONFIGFILE=afp2html.ini
```

## INSTALLDIR

Katalog zawierający programy, pliki konfiguracyjne i pliki odwzorowania usługi AFP2WEB Transform. Należy określić pełną nazwę ścieżki lub katalogu na serwerze HTTP.

**Uwaga:** Sprawdź uprawnienia do podanego katalogu. Procesy, które uruchamiają programy ODWEK, muszą mieć możliwość odczytywania katalogu instalacyjnego.

Parametr ten ma zasięg globalny i określa się go w sekcji AFP2HTML tylko raz.

Parametr opcjonalny.

Przykład:

```
[AFP2HTML]
INSTALLDIR=/QIBM/UserData/OnDemand/www/bin
```

## USEEXECUTABLE

Określa, czy aplikacja ODWEK uruchamia usługę AFP2WEB Transform za pomocą biblioteki współużytkowanej (DLL) czy pliku wykonywalnego (EXE).

**Ważne:** Produkt ODWEK dla systemu iSeries musi używać pliku wykonywalnego. Dlatego parametr ten zawsze musi mieć wartość 1 (jeden).

Parametr ten ma zasięg globalny i określa się go w sekcji AFP2HTML tylko raz.

Parametr opcjonalny.

Przykład:

```
[AFP2HTML]
USEEXECUTABLE=1
```

## [AFP2PDF]

Sekcja AFP2PDF zawiera parametry używane przez usługę IBM AFP2PDF Transform. Usługa AFP2PDF Transform przekształca dokumenty i zasoby AFP w dokumenty PDF, które można wyświetlić przy użyciu przeglądarki Adobe Acrobat.

### Uwagi:

1. Aby przekształcić dokumenty AFP w dokumenty PDF, administrator musi nabyć usługę AFP2PDF Transform od IBM, a następnie zainstalować i skonfigurować ją na serwerze HTTP. Więcej informacji na temat oferty AFP2PDF Transform można uzyskać u przedstawiciela IBM. Należy także określić opcje konfiguracyjne usługi AFP2PDF Transform. Więcej informacji na temat pliku konfiguracyjnego zawiera Dodatek F, "Transformowanie AFP do PDF", na stronie 133.
2. Aby przekształcić dokumenty za pomocą usługi AFP2PDF Transform, należy określić parametr `AFPVIEWING=PDF` w sekcji `DEFAULT BROWSER` (lub sekcjach dotyczących innych przeglądarek). Szczegóły znajdują się w sekcji "AFPVIEWING" na stronie 32. (Planując użycie funkcji `API Retrieve Document` (Wczytanie dokumentu), należy określić parametr `_afp=PDF`. Szczegóły znajdują się w sekcji "Wczytanie dokumentu" na stronie 77.)
3. Domyślnie aplikacja ODWEK do wyświetlania przekształconych dokumentów używa przeglądarki Adobe Acrobat. Użytkownik musi uzyskać odpowiednią dla używanych przeglądarek WWW wersję tego programu.

Sekcja ta ma zasięg globalny i określa się ją w pliku `ARSWWW.INI` tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

### CONFIGFILE

Plik konfiguracyjny zawierający opcje używane przez usługę AFP2PDF Transform podczas konwersji dokumentów i zasobów AFP w dokumenty PDF, które można wyświetlić za pomocą przeglądarki Adobe Acrobat. Dodatek F, "Transformowanie AFP do PDF", na stronie 133 zawiera przykładowy plik konfiguracyjny dostarczany razem z serwerem OnDemand. Szczegóły dotyczące opcji, jakie można określić w pliku konfiguracyjnym, znajdują się w dokumentacji usługi AFP2PDF Transform.

Parametr ten ma zasięg globalny i określa się go w sekcji `AFP2PDF` tylko raz.

Parametr opcjonalny.

Przykład:

```
[AFP2PDF]
CONFIGFILE=afp2pdf.ini
```

### INSTALLDIR

Katalog zawierający programy, pliki konfiguracyjne i pliki odwzorowania usługi AFP2PDF Transform. Należy określić pełną nazwę ścieżki lub katalogu na serwerze HTTP.

**Uwaga:** Sprawdź uprawnienia do podanego katalogu. Procesy, które uruchamiają programy ODWEK, muszą mieć możliwość odczytywania katalogu instalacyjnego.

Parametr ten ma zasięg globalny i określa się go w sekcji `AFP2PDF` tylko raz.

Parametr opcjonalny.

Przykład:

```
[AFP2PDF]
INSTALLDIR=/QIBM/UserData/OnDemand/www/bin
```

## USEEXECUTABLE

Określa, czy aplikacja ODWEK uruchamia usługę AFP2WEB Transform za pomocą biblioteki współużytkowanej (DLL) czy pliku wykonywalnego (EXE).

**Ważne:** Produkt ODWEK dla systemu iSeries musi używać pliku wykonywalnego. Dlatego parametr ten zawsze musi mieć wartość 1 (jeden).

Parametr ten ma zasięg globalny i określa się go w sekcji AFP2PDF tylko raz.

Parametr opcjonalny.

Przykład:

```
[AFP2PDF]
USEEXECUTABLE=1
```

## [MIMETYPES]

Sekcja MIMETYPES określa typ zawartości MIME (Multipurpose Internet Mail Extension) dla dokumentów wczytywanych z serwera OnDemand. Przeglądarka używa typu zawartości MIME w celu sformatowania i wyświetlenia dokumentu, wybrania odpowiedniego apletu lub przeglądarki do jego otwarcia lub do uruchomienia w tym celu programu zdefiniowanego przez użytkownika.

### Uwagi:

1. Sekcja MIMETYPES powinna zawierać parę parametr=*wartość* dla każdego rodzaju dokumentu, który może być wczytany z serwera OnDemand. Część parametr określa typ danych dokumentu na serwerze OnDemand. (Jest to typ danych przypisany do aplikacji OnDemand na stronie Wyświetlenie informacji). Część *wartość* określa program, który należy uruchomić w celu otwarcia dokumentu. Wielkość liter w części *wartość* jest rozróżniana.
2. W przykładowym pliku ARSWWW.INI (patrz "Przykład pliku ARSWWW.INI" na stronie 39) sekcja MIMETYPES zawiera jeden parametr dla każdego ze standardowych typów danych obsługiwanych przez serwer OnDemand (AFP, BMP, EMAIL, GIF, JFIF, LINE, PCX, PDF i TIFF).
3. Oprócz standardowych typów danych OnDemand obsługuje także typy zdefiniowane przez użytkownika. Zdefiniowany przez użytkownika typ danych może określać dowolny inny niż standardowy rodzaj danych przechowywanych w systemie. Aby użytkownicy mogli wyświetlać dokumenty zawierające typ danych zdefiniowany przez użytkownika, należy dodać odpowiedni parametr do sekcji MIMETYPE. Parametr ten musi określać typ zawartości MIME oraz rozszerzenie pliku podane dla aplikacji OnDemand na stronie Wyświetlenie informacji. Rozszerzenie pliku musi także być zarejestrowane w systemie operacyjnym klienta. Przypuśćmy na przykład, że użytkownik definiuje aplikację do przechowywania dokumentów Lotus WordPro w serwerze OnDemand. W tym celu na stronie Wyświetlenie informacji podaje rozszerzenie pliku LWP. Aby system rozpoznawał dokumenty wczytane z tej aplikacji, do pliku ARSWWW.INI należy dodać następujący parametr:

```
[MIMETYPES]
LWP=application/vnd.lotus-wordpro
```

Następnie, gdy użytkownik będzie wczytywał dokument z aplikacji, pakiet ODWEK ustawi typ zawartości MIME na `application/vnd.lotus-wordpro`, a system uruchomi

program Lotus WordPro w celu otwarcia dokumentu. W przypadku przeglądarki Netscape typ zawartości MIME należy zdefiniować za pomocą opcji konfiguracyjnej Preferences->Navigator->Applications.

Tabela 1 zawiera listę typów zawartości MIME dla niektórych aplikacji na komputery PC:

Tabela 1. Typy zawartości MIME dla niektórych aplikacji na komputery PC:

Aplikacja	Typy zawartości MIME
Aplikacje Lotus	WK1=application/vnd.lotus-1-2-3 WK3=application/vnd.lotus-1-2-3 WK4=application/vnd.lotus-1-2-3 123=application/vnd.lotus-1-2-3 APR=application/vnd.lotus-approach VEW=application/vnd.lotus-approach LWP=application/vnd.lotus-wordpro SAM=application/vnd.lotus-wordpro MWP=application/vnd.lotus-wordpro SMM=application/vnd.lotus-wordpro PRE=application/vnd.lotus-freelance PRZ=application/vnd.lotus-freelance
Aplikacje Microsoft	DOC=application/msword XLS=application/vnd.ms-excel PPS=application/vnd.ms-powerpoint PPT=application/vnd.ms-powerpoint MPD=application/vnd.ms-project MPP=application/vnd.ms-project MPT=application/vnd.ms-project MPD=application/vnd.ms-project
Aplikacje HTML	HTML=application/html HTM=application/htm

Sekcja ta ma zasięg globalny i określa się ją w pliku ARSWWW.INI tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

### AFP

Typ zawartości MIME dla dokumentów AFP, gdy w sekcji [DEFAULT BROWSER] określono parametr AFPVIEWING=NATIVE. Więcej informacji znajduje się w sekcji "AFPVIEWING" na stronie 32. Definiuje on typ MIME dokumentu, którego następnie przeglądarka używa w celu określenia, za pomocą jakiego programu dokument powinien być przetwarzany.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny.

Przykład:

```
[MIMETYPES]
AFP=application/afp
```

## **BMP**

Typ zawartości MIME dla dokumentów BMP. Domyślnie dokumenty BMP są wyświetlane przy użyciu przeglądarki Image Web Viewer.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny. Jeśli jednak nie zostanie podany, to aplikacja ODWEK ustawi typ zawartości MIME na `image/bmp` i uruchomi program powiązany z typem plików BMP w systemie operacyjnym klienta.

Przykład:

```
[MIMETYPES]
BMP=image/IBM-OnDemand
```

## **GIF**

Typ zawartości MIME dla dokumentów GIF. Domyślnie dokumenty GIF są wyświetlane przy użyciu przeglądarki Image Web Viewer.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny. Jeśli jednak nie zostanie określony, aplikacja ODWEK ustawi typ zawartości MIME na `image/gif` i do wyświetlenia dokumentów GIF użyje wbudowanych funkcji przeglądarki WWW.

Przykład:

```
[MIMETYPES]
GIF=image/IBM-OnDemand
```

## **EMAIL**

Typ zawartości MIME dla dokumentów EMAIL. Więcej informacji dotyczących przetwarzania dokumentów EMAIL przed wysłaniem do klienta znajduje się w sekcji “EMAILVIEWING” na stronie 33.

### **Uwagi:**

1. Przekształcając dokumenty EMAIL do formatu HTML, aplikacja ODWEK ustawia typ zawartości MIME na `text/html`. Jeśli w takiej sytuacji parametr EMAIL ma wartość, aplikacja ODWEK ignoruje ją.
2. Jeśli dokumenty EMAIL są wyodrębniane i dekompresowane z serwera OnDemand, aplikacja ODWEK użyje wartości parametru EMAIL w celu określenia programu, za pomocą którego należy otworzyć dokument.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny.

Przykład:

```
[MIMETYPES]
EMAIL=text/plain
```

## **JFIF**

Typ zawartości MIME dla dokumentów JFIF (JPEG). Domyślnie dokumenty JFIF są wyświetlane przy użyciu przeglądarki Image Web Viewer.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny. Jeśli jednak nie zostanie podany, to aplikacja ODWEK ustawi typ zawartości MIME na `image/jpeg` i uruchomi program powiązany z typem plików JPEG w systemie operacyjnym klienta.

Przykład:

```
[MIMETYPES]
JFIF=image/IBM-OnDemand
```

## LINE

Typ zawartości MIME dla dokumentów zawierających dane dla drukarki wierszowej. Więcej informacji dotyczących przetwarzania dokumentów, które zawierają dane dla drukarki wierszowej, przed wysłaniem do klienta znajduje się w sekcji “LINEVIEWING” na stronie 35.

Parametr ten jest używany w sytuacjach, gdy w sekcji [DEFAULT BROWSER] podano parametr LINEVIEWING=NATIVE. Jeśli dokumenty zawierające dane dla drukarki wierszowej są wyodrębniane i dekompresowane z serwera OnDemand, aplikacja ODWEK użyje wartości parametru LINE w celu określenia programu, który należy uruchomić w celu otwarcia dokumentu.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny.

Przykład:

```
[MIMETYPES]
LINE=text/html
```

## PCX

Typ zawartości MIME dla dokumentów PCX. Domyślnie dokumenty PCX są wyświetlane przy użyciu przeglądarki Image Web Viewer.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny. Jeśli jednak nie zostanie podany, to aplikacja ODWEK ustawi typ zawartości MIME na `image/pcx` i uruchomi program powiązany z typem plików PCX w systemie operacyjnym klienta.

Przykład:

```
[MIMETYPES]
PCX=image/IBM-OnDemand
```

## PDF

Typ zawartości MIME dla dokumentów PDF.

### Uwagi:

1. ODWEK korzysta z wartości parametru PDF do określenia, jaki program należy uruchamiać w celu otwierania dokumentów PDF. Domyślnie dokumenty PDF są otwierane za pomocą przeglądarki Adobe Acrobat.
2. Aby wyświetlić dokumenty PDF, należy pobrać i zainstalować moduł dodatkowy Adobe Acrobat viewer dla przeglądarek WWW używanych w organizacji.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny.



Przykład:

```
[MIMETYPES]
PDF=application/pdf
```

## TIFF

Typ zawartości MIME dla dokumentów TIFF. Domyślnie dokumenty TIFF są wyświetlane przy użyciu przeglądarki Image Web Viewer.

Parametr ten ma zasięg globalny i określa się go w sekcji MIMETYPES tylko raz.

Parametr opcjonalny. Jeśli jednak nie zostanie podany, to aplikacja ODWEK ustawi typ zawartości MIME na image/tiff i uruchomi program powiązany z typem plików TIFF w systemie operacyjnym klienta.

Przykład:

```
[MIMETYPES]
TIFF=image/IBM-OnDemand
```

## [ATTACHMENT IMAGES]

Sekcja ATTACHMENT IMAGES określa pliki obrazów, których aplikacja ODWEK używa do wyświetlania załączników do dokumentu. Każdy plik obrazu powinien zawierać ikonę odpowiadającą konkretnemu typowi załącznika. Można na przykład określić plik obrazu zawierającego ikony dla załączników tekstowych, map bitowych itd.

### Uwagi:

1. Każdy podany parametr musi określać typ pliku, któremu system operacyjny przypisuje typ załącznika. Typ pliku określa program, który system operacyjny uruchamia w celu przetworzenia załącznika. Na przykład, jeśli system operacyjny przypisze typ pliku TXT do załączników będących plikami tekstowymi, do sekcji ATTACHMENT IMAGES należy dodać parametr `TXT=wartość`. Jako *wartość* należy podać nazwę pliku zawierającego ikonę, która ma oznaczać tekstowy załącznik do dokumentu. Gdy użytkownik kliknie ikonę, system operacyjny uruchamia program, który został zarejestrowany do otwierania dokumentów typu TXT.
2. Domyślnie wszystkie załączniki do dokumentów są oznaczane plikiem `odc_att.gif` (który znajduje się w katalogu określonym przez parametr `IMAGEDIR` w sekcji `CONFIGURATION`). Pliku `odc_att.gif` OnDemand używa w odniesieniu do typów plików, dla których nie podano parametru w sekcji ATTACHMENT IMAGES.

Sekcja ta ma zasięg globalny i określa się ją w pliku ARSWWW.INI tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

## BMP

Parametr określający typ danych będących mapą bitową. Jego wartość określa plik zawierający ikonę, która oznacza mapę bitową załączoną do dokumentu.

Parametr ten ma zasięg globalny i określa się go w sekcji ATTACHMENT IMAGES tylko raz.

Parametr opcjonalny.

Przykład:

```
[ATTACHMENT IMAGES]
BMP=userBitMap.gif
```

## **GIF**

Parametr określający typ danych GIF. Jego wartość określa plik zawierający ikonę, która oznacza obraz GIF załączony do dokumentu.

Parametr ten ma zasięg globalny i określa się go w sekcji ATTACHMENT IMAGES tylko raz.

Parametr opcjonalny.

Przykład:

```
[ATTACHMENT IMAGES]
GIF=userGIF.gif
```

## **TXT**

Parametr określający typ danych TXT. Jego wartość określa plik zawierający ikonę, która oznacza plik tekstowy załączony do dokumentu.

Parametr ten ma zasięg globalny i określa się go w sekcji ATTACHMENT IMAGES tylko raz.

Parametr opcjonalny.

Przykład:

```
[ATTACHMENT IMAGES]
TXT=userText.gif
```

## **[NO HTML]**

Sekcja NO HTML zawiera parametry służące do przesłonięcia domyślnych znaków, które oddzielają łańcuchy znaków i listy wartości w danych wyjściowych ASCII oddzielonych separatorami. Funkcja generuje dane wyjściowe ASCII oddzielone separatorami, gdy ustawi się jej parametr `_nohtml` na 1 (jeden). Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.

Sekcja ta ma zasięg globalny i określa się ją w pliku ARSWWW.INI tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

### **BEGIN**

Znak używany przez aplikację ODWEK w celu określenia początku łańcucha znaków lub wartości. Separator BEGIN należy zmienić, jeśli łańcuch zawiera znak domyślny (znak `[]`).

Parametr ten ma zasięg globalny i określa się go w sekcji NO HTML tylko raz.

Parametr opcjonalny.

Przykład:

```
[NO HTML]
BEGIN=<
```

### **END**

Znak używany przez aplikację ODWEK w celu określenia końca łańcucha znaków lub wartości. Separator END należy zmienić, jeśli łańcuch zawiera znak domyślny (znak `[]`).

Parametr ten ma zasięg globalny i określa się go w sekcji NO HTML tylko raz.

Parametr opcjonalny.

Przykład:

```
[NO HTML]
END=>
```

## SEPARATOR

Znak, którego aplikacja ODWEK używa do oddzielania wartości w łańcuchu. Separator SEPARATOR należy zmienić, jeśli łańcuch zawiera znak domyślny (znak ^).

Parametr ten ma zasięg globalny i określa się go w sekcji NO HTML tylko raz.

Parametr opcjonalny.

Przykład:

```
[NO HTML]
SEPARATOR=;
```

## [DEFAULT BROWSER]

Sekcji DEFAULT BROWSER używa się do określania parametrów dla używanych przeglądarek. Podane tu parametry zostaną użyte wtedy, gdy nie będą podane w sekcji dotyczącej konkretnej przeglądarki opisanej w sekcji “[przełęczarka]” na stronie 38. (Parametry podane w sekcji dotyczącej konkretnej przeglądarki przesłaniają parametry, które znalazły się w sekcji DEFAULT BROWSER).

Sekcja ta ma zasięg globalny obejmujący wszystkie przeglądarki i określa się ją w pliku ARSWWW.INI tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

## ADDEXTENSION

Określa, czy trzysznakowe rozszerzenie pliku dokumentu jest dodawane do dodatkowych informacji o ścieżce w adresie URL zwracanym do przeglądarki. Dodanie rozszerzenia pliku do adresu URL ułatwia przeglądarkom określenie, jaki program należy uruchomić w celu otwarcia dokumentu. Wartością domyślną jest 0 (zero); oznacza ona, że rozszerzenie nie będzie dodawane do adresu URL.

**Uwaga:** Jeśli używana jest przeglądarka Microsoft Internet Explorer, IBM zaleca określenie parametru ADDEXTENSION=1, aby rozszerzenie pliku było dodawane do adresu URL.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
ADDEXTENSION=1
```

## ADDFIELDSTODOCID

Określa, czy wartości pól są dodawane do identyfikatorów dokumentów. (Identyfikatory dokumentów są zwracane przez funkcję tworzącą listę trafień). Wartością domyślną jest 0

(zero); oznacza ona, że wartości pól nie będą dodawane do identyfikatorów dokumentów. Jeśli aplikacja ODWEK ma do identyfikatorów dokumentów dodawać wartości pól, pojawia się one również w protokole systemowym pod warunkiem, że system skonfigurowano tak, aby zapisywał w nim komunikaty dla grupy aplikacji.

**Uwagi:**

1. Jeśli używana jest funkcja aktualizacji dokumentu, podanie wartości ADDFIELDSTODOCID=1 jest obowiązkowe.
2. Jeśli pole Flagi adnotacji w polu dokumentu w tabeli bazy danych ma wartość Tak, to określenie wartości ADDFIELDSTODOCID=1 jest **obowiązkowe**. Pole Flagi adnotacji w polu dokumentu w tabeli bazy danych można ustawić w oknie dialogowym Informacje o bazie danych na stronie Ogólne w definicjach grupy aplikacji w produkcie OnDemand. (Aby otworzyć okno dialogowe Informacje o bazie danych, należy kliknąć przycisk Zaawansowane).

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
ADDFIELDSTODOCID=1
```

## ADDNOTES

Określa, czy do dokumentów można dodawać adnotacje. Jeśli jest włączone, ODWEK umieszcza obok każdego dokumentu na liście element sterujący umożliwiający dodawanie adnotacji. Wartością domyślną jest 0 (zero); oznacza ona, że nie można dodawać adnotacji do dokumentów.

**Uwaga:** Uprawnienia poszczególnych użytkowników do dodawania adnotacji do dokumentów określa uprawnienie Annotation w grupie aplikacji OnDemand.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
ADDNOTES=1
```

## AFPVIEWING

Gdy użytkownik wczytuje dokument AFP z serwera OnDemand, wartość tego parametru określa, jakie czynności powinna wykonać aplikacja ODWEK przed wysłaniem dokumentu do klienta. Na przykład niektórzy użytkownicy przekształcają dokumenty AFP do formatu HTML za pomocą usługi AFP2WEB Transform i używają apletu AFP2HTML w celu wyświetlenia danych wyjściowych HTML. Powinni oni określić parametr AFPVIEWING=HTML, aby aplikacja ODWEK przekształcała dokument AFP przed wysłaniem go do klienta.

Parametrowi temu można nadać jedną z następujących wartości:

<b>ASCII</b>	ODWEK przekształca dokumenty AFP na tekst ASCII.
<b>HTML</b>	ODWEK przekształca dokumenty AFP w dokumenty HTML przy użyciu usługi AFP2WEB Transform.
<b>NATIVE</b>	ODWEK wyodrębnia i dekompresuje dokumenty AFP i ich zasoby z serwera OnDemand.

**Uwaga:** Określając parametr AFPVIEWING=NATIVE należy sprawdzić, czy typ zawartości MIME dla dokumentów AFP określa odpowiednią przeglądarkę do ich wyświetlania. Szczegóły znajdują się w sekcji “[MIMETYPES]” na stronie 25.

**PDF** ODWEK przekształca dokumenty AFP w dokumenty PDF przy użyciu usługi AFP2WEB Transform.

**Uwaga:** Określając parametr AFPVIEWING=PDF, należy sprawdzić, czy typ zawartości MIME dla dokumentów PDF określa odpowiednią przeglądarkę do ich wyświetlania. Szczegóły znajdują się w sekcji “[MIMETYPES]” na stronie 25.

**PLUGIN** ODWEK nie przekształca dokumentów AFP (jest to zachowanie domyślne).

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz. Używając funkcji Wczytanie dokumentu można przesłonić określone działanie za pomocą parametru `_afp`.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
AFPVIEWING=PLUGIN
```

## AUTODOCRETRIEVAL

Określa, czy klient automatycznie wyświetla dokument, gdy jeden i tylko jeden dokument pasuje do zapytania. Opcja ta oznacza, że w przypadku zapytań, które będą pasować tylko do jednego dokumentu, można skonfigurować system tak, aby pomijał stronę WWW z listą dokumentów i od razu wyświetlał znaleziony dokument. Wartością domyślną jest 0 (zero); oznacza ona, że aplikacja ODWEK wyświetli stronę WWW z listą dokumentów nawet wtedy, gdy tylko jeden dokument pasuje do zapytania.

**Ważne:** Automatycznego wczytywania dokumentu nie należy włączać, jeśli ma być używana przeglądarka Microsoft Internet Explorer. IBM zaleca określenie parametru AUTODOCRETRIEVAL=0 we wszystkich sekcjach, które są definiowane dla przeglądarki Internet Explorer.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
AUTODOCRETRIEVAL=1
```

## EMAILVIEWING

Gdy użytkownik wczytuje dokument EMAIL z serwera OnDemand, wartość tego parametru określa, jakie czynności powinna wykonać aplikacja ODWEK przed wysłaniem dokumentu do klienta.

Parametrowi temu można nadać jedną z następujących wartości:

**NATIVE** ODWEK wyodrębni i dekompresuje dokumenty EMAIL z serwera OnDemand.

**Uwaga:** Określając parametr EMAIL=NATIVE należy sprawdzić, czy typ zawartości MIME określa odpowiednią przeglądarkę do ich wyświetlania. Szczegóły znajdują się w sekcji “[MIMETYPES]” na stronie 25.

**HTML** ODWEK przekształca dokumenty EMAIL do formatu HTML. Jest to wartość domyślna.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz. Używając funkcji Wczytanie dokumentu można przesłonić określone działanie za pomocą parametru \_email.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
EMAILVIEWING=HTML
```

## ENCRYPTCOOKIES

Określa, czy aplikacja ODWEK szyfruje informacje cookie wysyłane do przeglądarki. Wartością domyślną jest 0 (zero); oznacza ona, że informacje cookie nie będą szyfrowane. Aby szyfrować wszystkie informacje cookie wysyłane do przeglądarki, należy podać wartość 1 (jeden).

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
ENCRYPTCOOKIES=1
```

## ENCRYPTURL

Określa, czy ODWEK ma szyfrować wartości serwer, ID\_użytkownika, hasło i ID\_dokumentu zawarte w adresie URL wysyłanym do przeglądarki. Wartością domyślną jest 0 (zero); oznacza ona, że wartości te nie będą szyfrowane. Aby umożliwić ich szyfrowanie, należy określić wartość 1 (jeden).

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny. Jeśli jednak do przesyłania parametrów i wartości z formularzy pomiędzy przeglądarką a serwerem HTTP użytkownik musi używać metody GET, to istnieje możliwość ich szyfrowania poprzez podanie parametru ENCRYPTURL=1. W sekcji “Ochrona serwera i danych” na stronie 7 znajduje się więcej informacji dotyczących atrybutu METHOD znacznika FORM.

Przykład:

```
[DEFAULT BROWSER]
ENCRYPTURL=1
```

## FOLDERDESC

Określa, czy opis folderu jest wyświetlany z prawej strony jego nazwy na stronie wyboru folderu. Wartością domyślną jest 0 (zero); oznacza ona, że opis folderu nie będzie wyświetlany. Aby opis folderu był wyświetlany, należy określić wartość 1 (jeden). Jeśli parametr nie zostanie określony lub jego wartość nie zostanie zdefiniowana lub rozpoznana, opis folderu nie będzie wyświetlany.

Parametr ten ma zasięg globalny, chyba że zostanie nadpisany w sekcji dotyczącej konkretnej przeglądarki (patrz sekcja “[przeglądarka]” na stronie 38). Parametr ten określa się tylko raz w sekcji DEFAULT BROWSER.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
FOLDERDESC=1
```

## LINEVIEWING

Gdy użytkownik wczytuje dokument zawierający dane dla drukarki wierszowej z serwera OnDemand, wartość tego parametru określa, jakie czynności powinna wykonać aplikacja ODWEK przed wysłaniem dokumentu do klienta.

Parametrowi temu można nadać jedną z następujących wartości:

<b>APPLET</b>	ODWEK przekształca dokumenty zawierające dane dla drukarki wierszowej za pomocą apletu Line Data (zachowanie domyślne).
<b>ASCII</b>	ODWEK przekształca dokumenty zawierające dane dla drukarki wierszowej w tekst ASCII.
<b>NATIVE</b>	ODWEK wyodrębni i dekompresuje dokumenty zawierające dane dla drukarki wierszowej z serwera OnDemand.

**Uwaga:** Określając parametr LINEVIEWING=NATIVE należy sprawdzić, czy typ zawartości MIME określa odpowiednią przeglądarkę do ich wyświetlania. Szczegóły znajdują się w sekcji “[MIMETYPES]” na stronie 25.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz. Używając funkcji Wczytanie dokumentu można przesłonić określone działanie za pomocą parametru `_line`.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
LINEVIEWING=APPLET
```

## MAXHITS

Maksymalna liczba pozycji zwrócona w liście dokumentów niezależnie od liczby pozycji pasujących do zapytania.

**Uwagi:**

1. Lista dokumentów składa się z pozycji pasujących do zapytania w kolejności załadowania tych pozycji do bazy danych.
2. ODWEK używa pierwszej z poniższych wartości w celu określenia liczby pozycji, jaką ma wyświetlić na liście dokumentów:
  - a. W przypadku funkcji Lista trafień dokumentów jest to wartość pola Maksymalna liczba trafień (podana na stronie Uprawnienia do folderu). Wartość ta zastępuje wszystkie inne wartości.
  - b. W przypadku funkcji Lista trafień dokumentów i Drukowanie dokumentu jest to wartość parametru `_max_hits` jeśli została określona dla funkcji. Wartość parametru `_max_hits` przesłania wartość parametru MAXHITS.
  - c. Wartość parametru MAXHITS, jeśli została podana.

- d. Jeśli nie została podana żadna z powyższych wartości, ODWEK wyświetla na liście dokumentów maksymalnie 200 pozycji.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
MAXHITS=200
```

## NOLINKS

Określa, czy lista dokumentów zawiera elementy sterujące służące do wyświetlania dokumentów. Jeśli tak, to ODWEK wyświetla element sterujący obok każdego dokumentu. Aby wyświetlić dokument, użytkownik musi użyć tego elementu. Wartością domyślną jest 0 (zero); oznacza ona, że w celu wyświetlenia dokumentu użytkownik musi użyć odsyłacza tekstowego.

**Ważne:** Jeśli używana jest przeglądarka Microsoft Internet Explorer, należy ustawić wartość NOLINKS=0. IBM zaleca określenie parametru NOLINKS=0 we wszystkich sekcjach, które są definiowane dla przeglądarki Internet Explorer.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
NOLINKS=1
```

## ODApplet.jre.path.IE

Patrz sekcja “Przeglądarka danych dla drukarki wierszowej w języku Java” na stronie 52.

## ODApplet.jre.path.NN

Patrz sekcja “Przeglądarka danych dla drukarki wierszowej w języku Java” na stronie 52.

## ODApplet.jre.version

Patrz sekcja “Przeglądarka danych dla drukarki wierszowej w języku Java” na stronie 52.

## ODApplet.version

Patrz sekcja “Przeglądarka danych dla drukarki wierszowej w języku Java” na stronie 52.

## SERVERPRINT

Określa, czy lista dokumentów zawiera elementy sterujące służące do wysyłania dokumentów do drukarki na serwerze. Jeśli tak, to ODWEK wyświetla element sterujący obok każdego dokumentu. Wartością domyślną jest 0 (zero); oznacza ona, że użytkownicy muszą otworzyć dokument przed wysłaniem go do drukarki na serwerze.

**Uwagi:**

1. Aby korzystać z wydruku serwerowego, w serwerze OnDemand musi być zdefiniowana przynajmniej jedna drukarka.
2. Uprawnienia poszczególnych użytkowników do drukowania dokumentów określa uprawnienie Drukowanie w grupie aplikacji OnDemand.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.



Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
SERVERPRINT=1
```

## SERVERPRINTERS

Służy do określania typu urządzeń drukujących na serwerze, które użytkownik może wybrać. Istnieją trzy rodzaje urządzeń obsługujących wydruki serwerowe:

- P** Drukarka na serwerze
- I** Drukarka na serwerze z informacjami
- F** Faks na serwerze

Można określić od zera do trzech rodzajów na liście oddzielonej przecinkami.

W poniższym przykładzie opisano, jak użytkownik może wybrać urządzenia drukujące i faksujące na serwerze:

```
[DEFAULT BROWSER]
SERVERPRINTERS=P,F
```

## SHOWDOCLOCATION

Podczas generowania danych wyjściowych ASCII oddzielonych separatorami zamiast danych wyjściowych HTML (patrz Dodatek H, “Brak danych wyjściowych HTML”, na stronie 137) określa, czy w danych wyjściowych znajdzie się położenie dokumentu w pamięci masowej. Szczegóły znajdują się w sekcji “Lista trafień dokumentów” na stronie 139. Wartością domyślną jest 0 (zero); oznacza ona, że położenie w pamięci masowej nie będzie uwzględnione w danych wyjściowych.

**Uwaga:** Aby wyświetlić położenie w pamięci masowej, należy także ustawić właściwość Wyświetlanie położenia dokumentu w folderze OnDemand.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
SHOWDOCLOCATION=1
```

## VIEWNOTES

Określa, czy można wyświetlić adnotacje do dokumentu. Jeśli jest włączony, ODWEK umieszcza obok każdego dokumentu na liście element sterujący umożliwiający wyświetlenie adnotacji. Wartością domyślną jest 0 (zero); oznacza ona, że nie można wyświetlać adnotacji.

**Uwaga:** Uprawnienia poszczególnych użytkowników do wyświetlania adnotacji do dokumentów określa uprawnienie Annotation w grupie aplikacji OnDemand.

Parametr ten ma zasięg globalny i określa się go w sekcji DEFAULT BROWSER tylko raz.

Parametr opcjonalny.

Przykład:

```
[DEFAULT BROWSER]
VIEWNOTES=1
```

## [przełładarka]

Można określić opcje dotyczące konkretnych przełładarek używanych w organizacji. Podane w tej sekcji parametry powodują przesłonięcie parametrów określonych w sekcji DEFAULT BROWSER pliku ARSWWW.INI. (Podane w sekcji DEFAULT BROWSER parametry będą stosowane, o ile nie zostaną określone w sekcji dotyczącej konkretnej przełładarki).

### Uwagi:

1. Nagłówek sekcji musi zawierać łańcuch określający przełładarkę, której mają dotyczyć podane opcje. ODWEK wyodrębnia tę wartość ze zmiennej środowiskowej HTTP\_USER\_AGENT, określając używaną przełładarkę. Następnie szuka w pliku ARSWWW.INI sekcji dotyczącej przełładarki, która odpowiada tej wartości. Jeśli nie zostanie znaleziona odpowiednia sekcja, ODWEK szuka w pliku ARSWWW.INI jednej z następujących sekcji:

[przełładarka wersja(główna.poboczna)/platforma]

[przełładarka wersja(główna.poboczna)]

[przełładarka wersja(główna)]

[przełładarka]

[DEFAULT BROWSER]

ODWEK zastosuje opcje z pierwszej sekcji, która będzie odpowiadać tej wartości.

2. Jako wartość przełładarka można określić IE lub Netscape.
3. Jako wartość platforma można określić WinNT lub Unix.

Sekcja dotycząca przełładarki ma zasięg globalny dla niej. Dla każdej przełładarki należy określić jedną sekcję. Należy podawać tylko te parametry, które mają zastąpić wartości podane w sekcji DEFAULT BROWSER.

Ta sekcja jest opcjonalna.

Może ona zawierać te same parametry, które zdefiniowano dla przełładarki domyślnej. Patrz sekcja “[DEFAULT BROWSER]” na stronie 31.

Przykłady:

```
[IE 5]
AUTODOCRETRIEVAL=0
NOLINKS=0
```

```
[Netscape 4.7]
AUTODOCRETRIEVAL=1    NOLINKS=1
```

## [DEBUG]

Sekcja DEBUG zawiera opcje, które mogą pomóc w rozwiązywaniu problemów związanych z korzystaniem z pakietu ODWEK. Musi ona być pierwszą instrukcją wykonywalną w pliku arswwww.ini.

Sekcja DEBUG ma zasięg globalny i określa się ją w pliku ARSWWW.INI tylko raz.

Ta sekcja jest opcjonalna.

Może ona zawierać następujące parametry:

## LOG

Umożliwia aplikacji ODWEK zapisywanie komunikatów i innych informacji do pliku protokołu. (Plik protokołu nosi nazwę ARSWWW.LOG).

Parametr ten ma zasięg globalny i określa się go w sekcji DEBUG tylko raz.

Parametr opcjonalny. Domyślnie ODWEK nie zapisuje komunikatów do pliku protokołu. Aby włączyć protokołowanie komunikatów, należy podać wartość 1 (jeden).

## LOGDIR

Określa katalog, w którym ODWEK zapisuje plik ARSWWW.LOG, jeśli w parametrze LOG włączono protokołowanie.

Parametr ten ma zasięg globalny i określa się go w sekcji DEBUG tylko raz.

Parametr opcjonalny. Domyślnie, jeśli protokołowanie jest włączone, ODWEK zapisuje plik protokołu w katalogu /QIBM/UserData/OnDemand/WWW/LOG.

Przykład:

```
[DEBUG]
LOGDIR=/QIBM/UserData/OnDemand/WWW/LOG
LOG=1
```

## Przykład pliku ARSWWW.INI

Poniżej przedstawiono przykładową konfigurację instancji dla domyślnej instancji QUSROND. Pogrubioną czcionką zaznaczono wiersze konfiguracji, które należy zmienić lub dodać. W niektórych przypadkach posiadają one również adnotacje, zapisane pogrubioną kursywą, w nawiasach. Wiersze komentarzy zaczynają się średnikiem. Należy upewnić się, że katalogi określone w tym zbiorze istnieją. Jeśli któryś z katalogów nie istnieje, aplikacja ODWEK nie powiedzie się.

```
=====
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Konfiguracja serwera   ;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
(Uwaga: Opcja DEBUG powinna być wyłączona, chyba że jest potrzebna do analizy błędów.
W znacznym stopniu wpływa ona na wydajność systemu i nie powinna być włączana,
jeśli nie jest to absolutnie konieczne. Aby włączyć opcję DEBUG, z następujących
trzech wierszy należy usunąć średniki. (Należy pamiętać o ponownym dodaniu
średników po zakończeniu korzystania z opcji DEBUG.)
;[debug]
;log=1
;logdir=/QIBM/UserData/OnDemand/www/logs

;[@SRV@_<host alias>]
;HOST=<host name>
;PORT=
;PROTOCOL=

[@SRV@ QUSROND]
HOST= (należy wpisać pełną nazwę systemu, na którym znajduje się instancja,
do której użytkownik chce uzyskać dostęp, lub jego adres TCP/IP)
PROTOCOL=0
PORT= (należy wpisać 4- lub 5-cyfrowy adres portu TCP/IP instancji, np. port 1450)

[configuration]
CodePage=37 (zmiana na stronę kodową instancji skonfigurowanej powyżej)
Language=ENU (zmiana na kod języka instancji skonfigurowanej powyżej)
TemplateDir=/QIBM/UserData/OnDemand/www/SAMPLES
ImageDir=/images
AppletDir=/applet
```

```

TempDir=/tmp
(Uwaga: Zaleca się korzystanie z buforowania w celu podwyższenia wydajności;
gdyby jednak grupa aplikacji lub dane uległy zmianie, zbiory buforowania
należy usunąć.)
CacheDir=/QIBM/UserData/OnDemand/www/CACHE
CacheSize=1024
CacheMinThreshold=40
CacheMaxThreshold=80
CacheDocs=1
CacheUserIDs=

[security]
ServerAccess=
ReportServerTimeout=0

;[afp2html]
;InstallDir=/QIBM/UserData/OnDemand/www/bin
;ConfigFile=/QIBM/UserData/OnDemand/www/bin/afp2html.ini

;[afp2pdf]
;InstallDir=/QIBM/UserData/OnDemand/www/bin
;ConfigFile=/QIBM/UserData/OnDemand/www/bin/afp2pdf.ini
;UseExecutable=0

[mimetypes]
BMP=image/IBM_OnDemand
GIF=image/IBM_OnDemand
JFIF=image/IBM_OnDemand
PCX=image/IBM_OnDemand
TIFF=image/IBM_OnDemand
PNG=image/IBM_OnDemand
PDF=application/pdf      AFP=application/afpLINE=application/line
LINE=application/line
EMAIL=text/html
META=application/unknown
DOC=application/msword
123=application/vnd.lotus-1-2-3
LWP=application/vnd.lotus-wordproSAM=application/vnd.lotus-wordpro
PRZ=application/vnd.lotus-freelance
XLS=application/vnd.ms-excel
PPS=application/vnd.ms-powerpoint
PPT=application/vnd.ms-powerpoint
HTML=application/html
HTM=application/htm
Uwaga: Możliwe, że konieczne będzie dodanie typów mime w celu obsługi innych
typów dokumentów. Przeglądarka musi znać aplikacje, które mają być wywoływane
do wyświetlania dokumentu )

[attachment images]
TXT=userText.gif      BMP=userBitMap.gif      GIF=userGIF.gif
[no html]
Begin=[
End=]
Separator=^

;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Przeglądarka domyślna ;;;
;;;;;;;;;;;;;;;;;;;;;;;;;

[default browser]
FOLDERDESC=1      (Należy ustawić na '1' jeśli ma być wyświetlany opis
folderu)
;AfpViewing=[ascii,html,native,pdf,plugin,xenos]
AfpViewing=plugin (Wymaga zainstalowania afpplgus.exe lub afpplgin.exe
na stacji roboczej)
;LineViewing=[ascii,applet,native]
LineViewing=applet (Wersja apletu danych dla drukarki, który ma zostać użyty,

```

wyświetlona jest poniżej, w opcji konfiguracyjnej  
*ODApplet.version*)

```
;EmailViewing=[html,native]
EmailViewing=html
;MetaViewing=[xenos,native]
MetaViewing=native
NoLinks=1
ViewNotes=1
AddNotes=1
ServerPrint=1
ServerPrinters=P
AutoDocRetrieval=1
MaxHits=200
ShowDocLocation=1
EncryptCookies=0
EncryptURL=0
ODApplet.version=2 (Druga wersja apletu przeglądarki danych drukarki)
ODApplet.jre.path.IE=http://www.java.com/en/download/windows_automatic.jsp
(Gdzie szukać środowiska wykonawczego programów Java (JRE),
jeśli nie jest ono jeszcze zainstalowane)
ODApplet.jre.version=1.4 (Minimalna wymagana wersja JRE)
```

(Poniższe sekcje umożliwiają nadpisanie domyślnych opcji konfiguracyjnych przedstawionych przeglądarek. W przypadku przeglądarki Netscape 3.01 wystarczy określenie Netscape.)

```
[IE]
NoLinks=0
AddExtension=1
AddFieldsToDocid=1 (możliwe, że konieczne będzie ustawienie na 0 w celu
wczytania dokumentów danych drukarki (*SCS))
AutoDocRetrieval=0
ViewNotes=1
AddNotes=1
```

**[Mozilla/4.0 (zgodne; MSIE 4.0; Windows 95)]**

**[Netscape 3.01]**

=====

## Dalsze kroki

Po zainstalowaniu oprogramowania ODWEK i skonfigurowaniu pliku ARSWWW.INI należy skonfigurować przykładowe aplikacje. Patrz sekcja Rozdział 4, “Konfigurowanie przykładowych aplikacji”, na stronie 43.



---

## Rozdział 4. Konfigurowanie przykładowych aplikacji

Konfigurowanie zestawu ODWEK wymaga zwykle wykonania następujących czynności:

1. Skontaktuj się z centrum wsparcia IBM, aby uzyskać najnowsze poprawki PTF dla serwera OnDemand. Lista najnowszych poprawek PTC znajduje się w raporcie APAR informacyjnym nr II14053.
2. Skontaktuj się z centrum wsparcia IBM, aby uzyskać najnowsze poprawki PTF dla grupy serwerów. Numer produktu serwera HTTP to 5722-DG1. Aby wyświetlić listę najnowszych poprawek PTF serwera HTTP, należy wybrać stronę [http://www-912.ibm.com/s\\_dir/sline003.NSF/GroupPTFs?OpenView&view](http://www-912.ibm.com/s_dir/sline003.NSF/GroupPTFs?OpenView&view) i kliknąć odpowiednią grupę PTF.
3. Skontaktuj się z centrum wsparcia IBM, aby uzyskać najnowsze poprawki PTF dla grupy baz danych. Aby wyświetlić listę najnowszych poprawek PTF dla DB2, należy wybrać stronę [http://www-912.ibm.com/s\\_dir/sline003.NSF/GroupPTFs?OpenView&view](http://www-912.ibm.com/s_dir/sline003.NSF/GroupPTFs?OpenView&view) i kliknąć odpowiednią grupę PTF.
4. Najnowsza wersja dokumentu *Przeczytaj zanim zaczniesz*, dotyczącego systemu OnDemand, znajduje się pod adresem <http://www.ibm.com/software/data/ondemand/400/support.html>. Zamieszczona jest pod nagłówkiem 'Learn' wraz z pozostałą dokumentacją Wersji 5 Wydania 4. Przed rozpoczęciem instalacji należy wydrukować i zapoznać się z całą zawartością zbioru.
5. Sprawdź wymagania wstępne ODWEK. Patrz Rozdział 3, "Instalowanie i konfigurowanie serwera HTTP", na stronie 11
6. Zainstaluj oprogramowanie OnDemand na serwerze iSeries. Patrz "Instalowanie w systemie i5/OS" na stronie 12
7. Skonfiguruj zbiór ARSWWW.INI. Patrz "Określanie pliku ARSWWW.INI" na stronie 13
8. Skonfiguruj serwer HTTP Apache. Dodatek G dokumentu Dodatek G, "Pliki konfiguracyjne serwera HTTP", na stronie 135 zawiera przykład zbioru konfiguracyjnego serwera HTTP.
9. Jeśli to konieczne, zaktualizuj listy uwierzytelniania QONDADM i QRDARS400. Patrz rozdział 3, "Inne wymagania" na stronie 11
10. Skonfiguruj przeglądarkę na swojej stacji roboczej. Na etap ten składają się następujące czynności:
  - a. Pobierz i zainstaluj odpowiednie zbiory modułów dodatkowych przeglądarki. Patrz Rozdział 5, "Instalowanie przeglądarek WWW", na stronie 45
  - b. Aby móc korzystać z najnowszej wersji apletu wyświetlania danych drukarki, pobierz i zainstaluj moduł środowiska wykonawczego Java ze strony <http://www.java.com>.
  - c. Upewnij się, że przeglądarka akceptuje wszystkie informacje cookie. Wybierz Narzędzia > Opcje internetowe i zakładkę Prywatne.
  - d. Upewnij się, że dla przeglądarki Internet Explorer wybrana jest opcja UTF-8. Wybierz Narzędzia > Opcje internetowe i w zakładce Zaawansowane zaznacz opcję "Zawsze wysyłaj adresy URL jako UTF-8".
  - e. Upewnij się, że środowisko wykonawcze programów Java jest włączone. Wybierz Narzędzia > Opcje internetowe i w zakładce Zaawansowane przejdź do sekcji Java (Sun). Upewnij się, że zaznaczona jest opcja "Używaj Java n vx.y.x dla <applet> (wymaga restartu)". Wersja wyświetlona w panelu odzwierciedla wersję zainstalowaną obecnie dla przeglądarki.





---

## Rozdział 5. Instalowanie przeglądarek WWW

---

### Przegląd

IBM dostarcza przeglądarki dla standardowych rodzajów dokumentów, które można wczytać z serwera OnDemand. Wymagania instalacyjne różnią się w zależności od tego, jakie przeglądarki są używane w organizacji.

- Do wyświetlania dokumentów zawierających dane dla drukarki wierszowej IBM zaleca użycie apletu Line Data. Jest on przechowywany na serwerze HTTP. Po udostępnieniu tego apletu, gdy użytkownik zechce wyświetlić dokument zawierający dane dla drukarki wierszowej, jest on automatycznie ładowany do pamięci na stacji roboczej. Należy sprawdzić, czy parametr LINEVIEWING w pliku ARSWWW.INI określa przeglądarkę, z której będą korzystać użytkownicy.
- Do wyświetlania dokumentów AFP można używać przeglądarki AFP Web Viewer dostarczanej z serwerem IBM OnDemand, apletu AFP2HTML lub przeglądarki Adobe Acrobat viewer.
  - Aby użytkownicy mogli wyświetlać dokumenty AFP za pomocą przeglądarki AFP Web Viewer dostarczanej z serwerem IBM OnDemand, muszą zainstalować ją na swoich stacjach roboczych.
  - Aby możliwe było wyświetlanie dokumentów AFP za pomocą apletu AFP2HTML, administrator musi zainstalować i skonfigurować usługę AFP2WEB Transform na serwerze HTTP oraz skonfigurować plik ARSWWW.INI. Aplet AFP2HTML jest przechowywany na serwerze HTTP. Po udostępnieniu tego apletu przez administratora, gdy użytkownik zechce wyświetlić dokument AFP, jest on automatycznie ładowany do pamięci na stacji roboczej.
  - Aby możliwe było wyświetlanie dokumentów AFP za pomocą przeglądarki Adobe Acrobat viewer, administrator musi zainstalować i skonfigurować usługę AFP2PDF Transform na serwerze HTTP oraz skonfigurować plik ARSWWW.INI. Po włączeniu funkcji transformowania domyślnie przeglądarka będzie próbowała uruchomić moduł dodatkowy Adobe Acrobat viewer za każdym razem, gdy użytkownik zechce wyświetlić dokument AFP. Użytkownik musi uzyskać i zainstalować na stacji roboczej przeglądarkę Adobe Acrobat viewer.

Należy sprawdzić, czy parametr AFPVIEWING w pliku ARSWWW.INI określa przeglądarkę, z której będą korzystać użytkownicy.

- Aby możliwe było wyświetlanie dokumentów BMP, GIF, JPEG, PCX i TIFF, IBM zaleca zainstalowanie przeglądarki IBM OnDemand Image Web Viewer na stacjach roboczych użytkowników. W przeciwnym razie powinni oni używać innej przeglądarki, która obsługuje te rodzaje dokumentów. (Na przykład większość przeglądarek WWW ma wbudowane funkcje wyświetlania obrazów GIF i JPEG). Jeśli użytkownicy mają używać innej przeglądarki, administrator powinien zmienić domyślny typ zawartości MIME dla tego rodzaju dokumentów. Należy sprawdzić, czy parametry w sekcji MIMETYPES pliku ARSWWW.INI określają przeglądarki, z których użytkownicy będą korzystać.

#### Uwagi:

1. Aby wyświetlić inne rodzaje danych, może być konieczne zainstalowanie innych przeglądarek. Na przykład do wyświetlania dokumentów PDF wczytanych z serwera OnDemand IBM zaleca zainstalowanie modułu dodatkowego Adobe Acrobat dla przeglądarek używanych w organizacji.
2. Aby było możliwe wyświetlanie dokumentów PDF, w katalogu modułów dodatkowych przeglądarki powinien znajdować się plik `nppdf32.dll`. W przypadku przeglądarki Internet Explorer jest to katalog `\Program Files\Internet Explorer\PLUGINS`. W

przypadku przeglądarki Netscape - katalog \Program Files\Netscape\Communicator\Program\Plugins. Jeśli w odpowiednim katalogu brak tego pliku, należy przeinstalować oprogramowanie firmy Adobe.

---

## Wymagania

Przeglądarki dostarczone przez IBM wymagają aplikacji Netscape Navigator 4.7 lub nowszej bądź Microsoft Internet Explorer 5.5 lub nowszej.

ODWEK wymaga, aby na komputerze PC możliwe było zapisywanie informacji cookie. Należy upewnić się, że użytkownicy skonfigurowali swoje przeglądarki tak, aby akceptowały informacje cookie.

W katalogu apletów IBM umieścił dwie wersje przeglądarki danych dla drukarki wierszowej w języku Java:

**ODLineDataViewer.jar** jest starą wersją przeglądarki danych dla drukarki wierszowej w języku Java, która wymaga obsługi języka Java w przeglądarce WWW. Obsługę języka Java najczęściej zapewnia wirtualna maszyna języka Java (JVM). **Uwaga:** W przypadku przeglądarki Microsoft Internet Explorer może zaistnieć konieczność zainstalowania modułu JVM za pomocą opcji niestandardowej instalacji przeglądarki.

**ODLineDataViewer2.jar** jest nową wersją przeglądarki, która wymaga modułu dodatkowego języka Java w wersji 1.4.1 lub nowszej. Nowa przeglądarka danych dla drukarki wierszowej w języku Java nie korzysta z obsługi języka Java w przeglądarce. Aby korzystać z nowej przeglądarki danych dla drukarki wierszowej w języku Java, użytkownik musi zainstalować moduł dodatkowy języka Java na komputerze PC. W sekcji "Przeglądarka danych dla drukarki wierszowej w języku Java" na stronie 52 znajdują się istotne informacje dotyczące konfiguracji.

Domyślny proces instalacji zainstaluje starą wersję przeglądarki Java Line Data Viewer.

Przeglądarka musi być zainstalowana w systemie Windows 2000, Windows XP lub Windows Server 2003 i do jej działania wymagany jest następujący sprzęt i oprogramowanie:

- fizyczne połączenie z siecią przy użyciu adaptera sieciowego, na przykład Token Ring lub Ethernet;
- obsługa protokołu TCP/IP w systemie Windows;
- co najmniej 256 MB pamięci RAM;
- procesor Pentium lub kompatybilny z Pentium o częstotliwości zegara 800 MHz lub większej;
- ekran i karta graficzna Super VGA o minimalnej rozdzielczości 800 x 600;
- co najmniej 20 MB wolnego miejsca na dysku do wyświetlania dokumentów;
- około 3 MB na każdej stacji roboczej dla przeglądarki IBM OnDemand AFP Web Viewer i 2 MB na każdej stacji roboczej dla przeglądarki IBM OnDemand Image Web Viewer.

---

## Instalowanie

**Uwaga:** Planując dystrybucję plików zdefiniowanych przez użytkownika razem z przeglądarką AFP Web Viewer, należy przed zainstalowaniem tej przeglądarki przez użytkowników skonfigurować jej plik instalacyjny, aby zawierał dodatkowe pliki zdefiniowane przez użytkownika. Więcej informacji znajduje się w sekcji "Dystrybucja plików zdefiniowanych przez użytkownika" na stronie 47.

Przeglądarki dostarczane przez IBM instaluje się z plików samorozpakowujących. W celu ich zainstalowania użytkownik powinien pobrać je do systemu Windows i uruchomić. Jeśli

podczas instalowania uruchomiona jest przeglądarka WWW, to aby można było korzystać z modułu przeglądarki dostarczonej przez IBM, należy zrestartować przeglądarkę WWW. W katalogu /QIBM/ProdData/OnDemand/www/plugins znajdują się następujące przeglądarki:

- afpplgus.exe - IBM OnDemand AFP Web Viewer - tylko w wersji angielskiej;
- afpplgin.exe - IBM OnDemand AFP Web Viewer - we wszystkich językach i z obsługą DBCS;
- afpplgin.zip - IBM OnDemand AFP Web Viewer - format ZIP we wszystkich językach i z obsługą DBCS;
- imgplgin.exe - IBM OnDemand Image Web Viewer - we wszystkich językach.

Proces instalacji kopiuje przeglądarkę i powiązane z nią pliki do wybranych przez użytkownika katalogów. Przeglądarka AFP Web Viewer wymaga około 3 MB wolnego miejsca na stacji roboczej. Przeglądarka Image Web Viewer wymaga około 2 MB wolnego miejsca na stacji roboczej. Należy przypomnieć użytkownikom o konieczności zrestartowania przeglądarki WWW, jeśli jest uruchomiona podczas instalacji.

**Uwaga:** Program instalacyjny instaluje przeglądarki jako moduły dodatkowe lub kontrolki ActiveX. Jeśli na stacji roboczej zainstalowana jest przeglądarka Internet Explorer, to program instalacyjny zainstaluje kontrolki ActiveX, natomiast jeśli zainstalowana jest przeglądarka Netscape - moduły dodatkowe do niej. Jeśli na stacji roboczej zainstalowane są obie przeglądarki, program instalacyjny zainstaluje kontrolki ActiveX dla programu Internet Explorer i moduły dodatkowe do programu Netscape.

---

## Dystrybucja plików zdefiniowanych przez użytkownika

Pliki zdefiniowane przez użytkownika można dystrybuować za pomocą oprogramowania IBM OnDemand AFP Web Viewer dostarczanego przez IBM. Przypuśćmy na przykład, że ktoś w organizacji utworzył pliki czcionek AFP dla dokumentów przechowywanych na serwerze OnDemand. Pliki te można rozpowszechnić za pomocą oprogramowania AFP Web Viewer. Dzięki temu, gdy użytkownik zechce przejrzeć dokument AFP, będzie on wyświetlony za pomocą poprawnych czcionek.

W celu rozpowszechnienia zdefiniowanych przez użytkownika plików za pomocą oprogramowania AFP Web Viewer należy zapisać je w pliku instalacyjnym i umieścić go we współużytkowanym miejscu. Po uruchomieniu pliku instalacyjnego przez użytkownika program ten automatycznie zainstaluje przeglądarkę AFP Web Viewer oraz niestandardowe pliki czcionek na stacji roboczej użytkownika.

Za pomocą przeglądarki AFP Web Viewer można rozpowszechniać następujące rodzaje plików zdefiniowanych przez użytkownika:

- Pliki czcionek AFP. Są one kopiowane do podkatalogu FONT w katalogu docelowym przeglądarki AFP Web Viewer na stacji roboczej.
- Pliki czcionek Adobe Type 1. Są one kopiowane do katalogu określonego przez użytkownika i instalowane w menedżerze ATM przez program instalacyjny.
- Pliki czcionek TrueType. Są one kopiowane do katalogu FONTS systemu Windows i instalowane w tym systemie przez program instalacyjny.
- Różne pliki zdefiniowane przez użytkownika. Są one kopiowane do katalogu docelowego przeglądarki AFP Web Viewer na stacji roboczej użytkownika.

**Uwaga:** Program instalacyjny kopiuje pliki zdefiniowane przez użytkownika na stację roboczą po plikach przeglądarki AFP Web Viewer dostarczonych przez IBM. Jeśli plik zdefiniowany przez użytkownika będzie miał tę samą nazwę, co plik dostarczony przez IBM, to plik dostarczony przez IBM zostanie zastąpiony plikiem

zdefiniowanym przez użytkownika. Można na przykład skorzystać z tego, aby przeprowadzić dystrybucję zaktualizowanego pliku FLDPORT2.INI lub zmodyfikowanych plików czcionek AFP IBM.

W poniższych tematach znajduje się więcej informacji dotyczących konfigurowania i dystrybucji przeglądarki AFP Web Viewer:

- Instalowanie plików przeglądarki AFP Web Viewer dostarczonych przez IBM
- Dodawanie podkatalogów dla plików zdefiniowanych przez użytkownika
- Umieszczanie plików zdefiniowanych przez użytkownika w podkatalogach
- Konfigurowanie plików czcionek
- Tworzenie pliku instalacyjnego przeglądarki AFP Web Viewer
- Instalowanie przeglądarki AFP Web Viewer na stacji roboczej użytkownika

## Instalowanie plików przeglądarki AFP Web Viewer

Większość klientów używa jednego lub dwóch sposobów dystrybucji plików przeglądarki z serwera w zależności od tego, czy razem z przeglądarką AFP Web Viewer zamierzają dostarczyć pliki zdefiniowane przez użytkownika:

- Instalacja standardowa. Służy do dystrybuowania plików przeglądarki AFP Web Viewer dostarczonych przez IBM oraz przygotowania do dystrybucji plików zdefiniowanych przez użytkownika razem z przeglądarką AFP Web Viewer. Podczas instalowania oprogramowania ODWEK przez administratora na serwerze HTTP pliki instalacyjne przeglądarek są umieszczane w odpowiednim katalogu na serwerze. Dla każdej przeglądarki powinien znaleźć się tam plik instalacyjny (EXE) oraz jeden plik archiwum ZIP dla przeglądarki AFP Web Viewer. Administrator zazwyczaj przenosi pliki instalacyjne do katalogu publicznego na serwerze i tworzy stronę WWW z odsyłaczami do nich. Użytkownik instaluje przeglądarkę, ładując stronę w przeglądarce WWW i klikając odsyłacz do odpowiedniego pliku instalacyjnego.
- Niestandardowa instalacja przeglądarki AFP Web Viewer. Służy do dystrybucji plików zdefiniowanych przez użytkownika razem z przeglądarką AFP Web Viewer.
  1. Skonfiguruj serwer do instalacji standardowej.
  2. Zanim użytkownicy zainstalują przeglądarkę uzyskaj kopię pliku archiwum ZIP przeglądarki AFP Web Viewer ZIP.
  3. Wyodrębnij pliki z archiwum ZIP do pustego katalogu roboczego.
  4. Dodaj do niego podkatalogi i umieść w nich pliki zdefiniowane przez użytkownika. Szczegóły znajdują się w sekcjach “Dodawanie podkatalogów” na stronie 49 i “Przechowywanie plików zdefiniowanych przez użytkownika” na stronie 49.
  5. W celu dystrybucji zdefiniowanych przez użytkownika plików czcionek Adobe Type 1 utwórz plik konfiguracyjny czcionek. Szczegóły znajdują się w sekcji “Konfigurowanie plików czcionek” na stronie 49.
  6. Po skonfigurowaniu wszystkich katalogów i plików utwórz samorozpakowujący się plik EXE w celu dystrybucji. Szczegóły znajdują się w sekcji “Tworzenie pliku instalacyjnego przeglądarki AFP Web Viewer” na stronie 50.
  7. Zastąp plik EXE dostarczony przez IBM w instalacji standardowej utworzonym przez siebie samorozpakowującym się plikiem EXE.
  8. Po zakończeniu przez administratora czynności opisanych w krokach od 1 do 7 użytkownicy mogą zainstalować przeglądarkę AFP Web Viewer i zdefiniowane przez użytkownika pliki, ładując w przeglądarkach WWW odpowiednią stronę i klikając odsyłacz do zaktualizowanego pliku instalacyjnego.

## Dodawanie podkatalogów

Przygotowane do dystrybucji pliki zdefiniowane przez użytkownika muszą być przechowywane w drzewie podkatalogów CUSTOM w głównym katalogu instalacyjnym klienta. Na przykład główny katalog instalacyjny klienta można nazwać \ONDEMAND\AFP32.

Aby skonfigurować główny katalog instalacyjny klienta i umieścić w nim pliki zdefiniowane przez użytkownika:

1. Utwórz katalog CUSTOM w głównym katalogu instalacyjnym klienta. Na przykład:

```
\ondemand\afp32\custom
```

**Uwaga:** W katalogu CUSTOM można umieścić inne<sup>1</sup> pliki zdefiniowane przez użytkownika, które mają być udostępnione użytkownikom. Program instalacyjny kopiuje pliki z tego katalogu do katalogu docelowego przeglądarki AFP Web Viewer na stacji roboczej.

2. Dodaj jeden lub wiele podkatalogów katalogu CUSTOM. Dodawane podkatalogi zależą od typu plików zdefiniowanych przez użytkownika, które mają być udostępnione użytkownikom.

- Utwórz podkatalog FONT katalogu CUSTOM, aby umieścić w nim pliki czcionek AFP (typy plików FNT i MAP). Na przykład:

```
\ondemand\afp32\custom\font
```

Program instalacyjny kopiuje te pliki do katalogu docelowego FONT przeglądarki AFP Web Viewer na stacji roboczej.

- Utwórz podkatalog TYPEONE katalogu CUSTOM, aby umieścić w nim pliki czcionek Adobe Type 1 (typy plików PFB i PFM) oraz plik konfiguracyjny czcionek. Na przykład:

```
\ondemand\afp32\custom\typeone
```

Program instalacyjny kopiuje te pliki do katalogu podanego przez użytkownika i instaluje czcionki w menedżerze ATM.

- Utwórz podkatalog TRUETYPE katalogu CUSTOM, aby umieścić w nim pliki czcionek Windows TrueType (typ plików TTF). Na przykład:

```
\ondemand\afp32\custom\truetype
```

Program instalacyjny kopiuje pliki z tego katalogu do katalogu FONT systemu Windows i instaluje czcionki w systemie Windows.

## Przechowywanie plików zdefiniowanych przez użytkownika

Po wyodrębnieniu dostarczonych przez IBM plików instalacyjnych do katalogu roboczego i utworzeniu katalogów CUSTOM można umieścić pliki zdefiniowane przez użytkownika w poszczególnych podkatalogach. Skopiuj na przykład pliki Adobe Type 1 (typy plików PFB i PFM), które chcesz udostępnić użytkownikom, do katalogu \ONDEMAND\AFP32\CUSTOM\TYPEONE.

## Konfigurowanie plików czcionek

Aby przeprowadzić dystrybucję zdefiniowanych przez użytkownika plików czcionek Adobe Type 1, wykonaj następujące czynności:

---

1. Pliki inne niż pliki czcionek AFP, Adobe Type 1 i Windows TrueType.

1. Umieść zdefiniowane przez użytkownika pliki czcionek Type 1 (typy plików PFB i PFM) w podkatalogu TYPEONE katalogu CUSTOM. Więcej informacji znajduje się w sekcji “Dodawanie podkatalogów” na stronie 49.
2. Utwórz plik konfiguracyjny czcionek Type 1. Poniżej opisano tworzenie pliku konfiguracyjnego czcionek Type 1.

Plik konfiguracyjny czcionek Type 1 musi mieć nazwę ATM\_INI.CFG i znajdować się w podkatalogu TYPEONE katalogu CUSTOM. Więcej informacji na temat katalogów dystrybucyjnych znajduje się w sekcji “Dodawanie podkatalogów” na stronie 49.

Każdy rekord (wiersz) w pliku konfiguracyjnym czcionek Type 1 określa jedną i tylko jedną zdefiniowaną przez użytkownika czcionkę Adobe Type 1, która będzie udostępniona użytkownikom. Format rekordu jest następujący:

```
nazwa_czcionki=nazwa_pliku.PFM,nazwa_pliku.PFB
```

Gdzie `nazwa_czcionki` jest nazwą czcionki Type 1 wyświetlaną na liście czcionek w panelu sterującym ATM, `nazwa_pliku.PFM` jest nazwą pliku PFM czcionki, a `nazwa_pliku.PFB` jest nazwą pliku PFB czcionki. Poniżej przedstawiono przykład pliku konfiguracyjnego czcionek Type 1 z dwoma rekordami:

```
Courier,BOLD=coub.pfm,coub.pfb
SonoranSansSerif_36,BOLDITALIC=c0a175z0.pfm,c0a175z0.pfb
```

Pierwszy rekord w pliku określa czcionkę o nazwie Courier,BOLD i jej plik PFM `coub.pfm` oraz plik PFB `coub.pfb`. Drugi rekord określa czcionkę o nazwie SonoranSansSerif\_36,BOLDITALIC i jej plik PFM `c0a175z0.pfm` oraz plik PFB `c0a175z0.pfb`.

Po uruchomieniu pliku instalacyjnego przeglądarki AFP Web Viewer, który zawiera zdefiniowane przez użytkownika pliki czcionek Adobe Type 1 program instalacyjny przetworzy pliki czcionek w następujący sposób:

1. Skopiuje zdefiniowane przez użytkownika pliki czcionek Adobe Type 1 (typy plików PFB i PFM) znalezione w katalogu TYPEONE do katalogu docelowego. Katalog docelowy jest określany przez użytkownika.
2. Sprawdzi, czy dla każdej czcionki zdefiniowanej w pliku konfiguracyjnym czcionek Type 1 (ATM\_INI.CFG) zostały skopiowane dwa pliki czcionek. Nazwy plików kopiowanych na stację roboczą muszą być takie same, jak nazwy określone w pliku konfiguracyjnym czcionek.

**Uwaga:** Jeśli nazwy plików czcionek określone w pliku konfiguracyjnym nie odpowiadają nazwom plików skopiowanych do stacji roboczej, program instalacyjny wyświetli komunikat ostrzegawczy i nie zainstaluje czcionki.

3. Doda informację o ścieżce do plików PFB i PFM, korzystając z podanego przez użytkownika katalogu docelowego.
4. Zainstaluje czcionki w menedżerze ATM.

## Tworzenie pliku instalacyjnego przeglądarki AFP Web Viewer

Po zakończeniu tworzenia katalogów i umieszczania plików w drzewie katalogów CUSTOM należy utworzyć plik instalacyjny zawierający pliki zdefiniowane przez użytkownika oraz pliki przeglądarki AFP Web Viewer dostarczone przez IBM. Plik instalacyjny zazwyczaj ma nazwę `Setup.exe`.

Wiele firm produkuje oprogramowanie do pakowania plików i aplikacji w jeden samorozpakowujący się plik wykonywalny przeglądarki AFP Web Viewer w celu dalszej dystrybucji. Na przykład firma InstallShield Software Corporation oferuje produkt o nazwie PackageForTheWeb.

**Uwaga:** Oprogramowanie innych firm nie jest obsługiwane przez IBM.

Mając oprogramowanie do pakowania, należy je uruchomić i postępować zgodnie z wyświetlanymi instrukcjami, aby utworzyć plik instalacyjny przeglądarki AFP Web Viewer, który zawiera pliki zdefiniowane przez użytkownika oraz pliki przeglądarki AFP Web Viewer dostarczone przez IBM.

## Instalowanie przeglądarki AFP Web Viewer na stacji roboczej użytkownika

Po skonfigurowaniu drzewa katalogów CUSTOM, utworzeniu pliku instalacyjnego przeglądarki AFP Web Viewer i zastąpieniu go na serwerze użytkownicy mogą zacząć instalować przeglądarkę AFP Web Viewer i pliki zdefiniowane przez użytkownika. Następnym razem, gdy użytkownik aktywuje odsyłacz do pliku instalacyjnego przeglądarki AFP Web Viewer na serwerze, program instalacyjny zainstaluje przeglądarkę AFP Web Viewer i skopiuje wszystkie zdefiniowane przez użytkownika pliki umieszczone w pliku instalacyjnym przeglądarki AFP Web Viewer na stacji roboczej użytkownika.

---

## Odwzorowywanie czcionek AFP

Czcionki AFP, z którymi utworzono dokument, muszą być odwzorowane na czcionki, które nadają się do wyświetlania za pomocą modułu dodatkowego AFP. Aplikacja ODWEK zawiera pliki definicji czcionek, które umożliwiają odwzorowanie czcionek IBM Core Interchange (tylko skrypt łaciński) i zapewniających kompatybilność z czcionkami TrueType. Pliki definicji i odwzorowania czcionek znajdują się w podkatalogu FONT, w którym znajduje się kod przeglądarki WWW AFP.

Jeśli w dokumencie używane są czcionki niezdefiniowane w przeglądarce WWW AFP, jeśli któryś z użytkowników zmodyfikował rdzenne czcionki IBM lub jeśli któryś z użytkowników utworzył czcionki AFP, to należy zdefiniować je w plikach definicji czcionek, aby przeglądarka WWW AFP mogła poprawnie wyświetlać dokumenty. Szczegóły dotyczące odwzorowywania czcionek AFP, plików definicji czcionek oraz inne informacje techniczne związane z czcionkami AFP i TrueType znajdują się w publikacji *AFP Workbench Technical Reference*.

---

## Przeglądarka AFP Web viewer

Poniższe ustawienia można zastosować z widoku logicznego na serwerze do przeglądarki AFP Web Viewer.

- Kolor tła. Obsługiwane są poniższe kolory. Żaden inny kolor nie jest obsługiwany.
  - Zielony pasek (wyświetlany z białym tłem)
  - Zielony
  - Czerwony
  - Żółty
  - Czarny
  - Biały
  - Szary
- Kolor obrazu. Obsługiwane są poniższe kolory. Żaden inny kolor nie jest obsługiwany.

- Żółty
  - Niebieski
  - Czerwony
  - Purpurowy
  - Zielony
  - Niebieskozielony
  - Domyślny (powinien być wyświetlany jako czarny)
- Powiększenie.

**Uwaga:** Wybrany kolor obszaru nie ma zastosowania do przeglądarki AFP Web Viewer. Wybrany obszar zawsze ma biały tekst i czarne tło.

---

## Przeglądarka Image Web viewer

Poniższe informacje mają zastosowanie, gdy do wyświetlania obrazów wielostronicowych używana jest przeglądarka obrazów WWW.

**Uwaga:** Poniższa procedura wymaga edycji rejestru komputera. Nie zaleca się modyfikowania rejestru, jeśli nie jest to absolutnie niezbędne. W przypadku błędu w rejestrze komputer może przestać działać prawidłowo. Przed kontynuacją należy wykonać kopię zapasową rejestru i wiedzieć, jak odtworzyć rejestr do postaci, w jakiej znajdował się podczas ostatniego udanego uruchomienia komputera. Instrukcje na ten temat zawiera dokumentacja systemu Windows.

W przypadku obrazów wielostronicowych podczas przeciągania paska przewijania pojawia się obok niego małe okno. Zawiera ono numer strony odpowiadający pozycji paska oraz liczbę stron, które zajmuje obraz. Na przykład wartości 5 / 10 oznaczają, że obraz zajmuje dziesięć stron, a po zwolnieniu przycisku myszy stroną bieżącą będzie strona piąta.

Można wyłączyć tę funkcję, modyfikując wartość następującego klucza rejestru:

HKEY\_LOCAL\_MACHINE\Software\IBM\OnDemand Image Web Viewer\Preferences

Jeśli wartość łańcucha `PageNumberScroll` zostanie ustawiona na 0 (zero), podczas przeciągania paska przewijania okno z numerem strony nie będzie wyświetlane.

Jeśli wartość łańcucha `PageNumberToolbar` w tym samym kluczu będzie ustawiona na 1 (jeden), w przypadku obrazów wielostronicowych informacje o numerze strony będą wyświetlone w pasku narzędzi. Na przykład wartości 3 / 5 oznaczają, że obraz zajmuje pięć stron oraz że strona trzecia jest stroną bieżącą.

---

## Przeglądarka danych dla drukarki wierszowej w języku Java

IBM udostępnia obecnie zaawansowaną przeglądarkę danych dla drukarki wierszowej w języku Java. Ulepszenia obejmują rozszerzone funkcje drukowania, na przykład drukowanie całej szerokości strony. Graficzny interfejs użytkownika jest oparty na bibliotece Swing.

W katalogu apletów IBM umieścił dwie wersje przeglądarki danych dla drukarki wierszowej w języku Java:

**ODLineDataViewer.jar** jest starą wersją przeglądarki, która wymaga modułu dodatkowego języka Java w wersji 1.1.8 lub nowszej.

**ODLineDataViewer2.jar** jest nową wersją przeglądarki, która wymaga modułu dodatkowego języka Java w wersji 1.4.1 lub nowszej.



Klienci mogą używać nowej lub starej przeglądarki danych dla drukarki wierszowej w języku Java. Wybór określa się, ustawiając parametry w sekcji [DEFAULT BROWSER] pliku ARSWWW.INI. Ponadto nowa przeglądarka wymaga wersji 1.4.1 lub nowszej modułu dodatkowego języka Java do przeglądarki. Dodatkowe parametry w pliku ARSWWW.INI określają numer wersji i położenie pliku instalacyjnego modułu dodatkowego języka Java, jeśli nie jest on zainstalowany na danej stacji roboczej.

Tabela 2 zawiera nowe parametry, które są określone w pliku ARSWWW.INI i obsługują przeglądarkę danych dla drukarki wierszowej w języku Java.

*Tabela 2. Parametry w pliku ARSWWW.INI, które obsługują przeglądarkę danych dla drukarki wierszowej w języku Java*

Parametr	Wartość	Komentarz
ODApplet.version	1	Określa, że ma być wywołana stara wersja przeglądarki danych dla drukarki wierszowej w języku Java. Jeśli jest określony, pozostałe parametry są ignorowane. <b>Uwaga:</b> Jest to wartość domyślna. Pominięcie tego parametru również spowoduje użycie starej wersji przeglądarki danych dla drukarki wierszowej w języku Java przez produkt ODWEK.
	2	Określa, że ma być wywołana nowa (rozszerzona) wersja przeglądarki danych dla drukarki wierszowej w języku Java. Jeśli jest określony, pozostałe trzy parametry będą brane pod uwagę.

Tabela 2. Parametry w pliku ARSWWW.INI, które obsługują przeglądarkę danych dla drukarki wierszowej w języku Java (kontynuacja)

Parametr	Wartość	Komentarz
ODApplet.jre.path.IE	http://java.sun.com/getjava/installer.html	Dla przeglądarki Internet Explorer. Powoduje automatyczne pobranie i zainstalowanie najnowszej wersji modułu dodatkowego języka Java z serwisu WWW java.sun.com. Na stronie <a href="http://java.sun.com/getjava/install-windows.html">http://java.sun.com/getjava/install-windows.html</a> znajduje się podgląd działań, jakie są wykonywane podczas automatycznego pobierania i instalowania modułu dodatkowego języka Java. <b>Uwaga:</b> Po zainstalowaniu modułu dodatkowego może być konieczne zrestartowanie przeglądarki.
	<położenie>	Określa położenie pliku instalacyjnego modułu dodatkowego języka Java w sieci intranet. Położenie musi być określone za pomocą obsługiwanej przez przeglądarkę protokołu, na przykład http, file lub ftp. Na przykład:  file://nazwa_zasobu_współużytkowanego/java/moduły_dodatkowe/plugin.exe  <b>Uwaga:</b> Administrator musi najpierw pobrać plik instalacyjny modułu dodatkowego języka Java i umieścić go w podanym miejscu.  Dzięki określeniu położenia pliku instalacyjnego przeglądarka automatycznie zainstaluje moduł dodatkowy języka Java na stacji roboczej. Po zainstalowaniu modułu dodatkowego może być konieczne zrestartowanie przeglądarki.

Tabela 2. Parametry w pliku ARSWWW.INI, które obsługują przeglądarkę danych dla drukarki wierszowej w języku Java (kontynuacja)

Parametr	Wartość	Komentarz
ODApplet.jre.path.NN	http://java.sun.com/j2se/1.4.1/download.html	Dla przeglądarki Netscape. Określa, że w celu zainstalowania modułu dodatkowego języka Java należy otworzyć stronę pobierania środowiska JRE/J2SE. Następnie należy kliknąć odsyłacz umożliwiający pobranie pliku instalacyjnego modułu dodatkowego języka Java dla używanej platformy. Po pobraniu pliku instalacyjnego należy zainstalować moduł dodatkowy na stacji roboczej. Po zainstalowaniu modułu dodatkowego może być konieczne zrestartowanie przeglądarki.
	<położenie>	Określa położenie plików modułu dodatkowego w sieci intranet. Położenie musi być określone za pomocą obsługiwanej przez przeglądarkę protokołu, na przykład http, file lub ftp. Na przykład: http://Serwer_WWW/tmp/ondemand/java/plugins  <b>Uwaga:</b> Administrator musi najpierw pobrać pliki instalacyjne i umieścić je w podanym miejscu. Nie można podać ścieżki do konkretnego pliku, ponieważ nie wiadomo, w jakim systemie operacyjnym działa przeglądarka Netscape. Podany format umożliwia administratorowi pobranie w razie potrzeby modułu dodatkowego na inne platformy.  Użytkownik musi sam zainstalować moduł dodatkowy w języku Java na stacji roboczej. Po zainstalowaniu modułu dodatkowego może być konieczne zrestartowanie przeglądarki.

Tabela 2. Parametry w pliku ARSWWW.INI, które obsługują przeglądarkę danych dla drukarki wierszowej w języku Java (kontynuacja)

Parametr	Wartość	Komentarz
ODApplet.jre.version	<wersja>	Określa wersję modułu dodatkowego w języku Java, która ma zostać użyta. Należy podać wersję 1.4 lub nowszą. Aby obsługiwane było dowolne wydanie modułu dodatkowego na tym poziomie (na przykład 1.4.0, 1.4.0_03, 1.4.1_01), należy podać główny numer wersji (na przykład 1.4). Aby obsługiwane było tylko konkretne wydanie modułu dodatkowego w języku Java, należy podać pełny numer wersji (na przykład 1.4.1_01). Poprawne numery wersji są dostępne w serwisie WWW <a href="http://java.sun.com">java.sun.com</a> . Na przykład: 1.4  lub: 1.4.1_01

Poniższy przykład ilustruje sposób konfigurowania pliku ARSWWW.INI, aby obsługiwał starą wersję przeglądarki danych dla drukarki wierszowej w języku Java.

```
[DEFAULT BROWSER]
ODApplet.version=1
```

**Uwagi:**

1. W razie pominięcia parametru ODApplet.version w pliku ARSWWW.INI, produkt ODWEK użyje starej wersji przeglądarki.
2. Parametry ODApplet mają zasięg globalny i można je określić tylko w sekcji DEFAULT BROWSER. Parametry te podane w jakiegokolwiek innej sekcji będą ignorowane.

Poniższy przykład ilustruje sposób konfigurowania pliku ARSWWW.INI, aby obsługiwał nową (rozszerzoną) wersję przeglądarki danych dla drukarki wierszowej w języku Java oraz wersję 1.4 lub nowszą modułu dodatkowego języka Java. W przypadku przeglądarki Internet Explorer użytkownicy mogą automatycznie pobierać i instalować najnowszą wersję modułu dodatkowego języka Java z serwisu WWW [java.sun.com](http://java.sun.com). W przypadku przeglądarki Netscape administrator umieszcza pliki instalacyjne modułu dodatkowego w języku Java dla różnych platform w podanym miejscu na lokalnym serwerze WWW, aby użytkownicy nie musieli łączyć się ze stroną pobierania oprogramowania JRE/J2SE w serwisie [java.sun.com](http://java.sun.com).

**Uwaga:** Możliwość pobrania / zainstalowania modułu dodatkowego mają tylko użytkownicy, na których stacjach roboczych nie ma zainstalowanej wersji 1.4 lub nowszej modułu dodatkowego w języku Java.

```
[DEFAULT BROWSER]
ODApplet.version=2
ODApplet.jre.path.IE=http://java.sun.com/getjava/installer.html
ODApplet.jre.path.NN=http://localWebServer/java/plugins
ODApplet.jre.version=1.4
```

---

## Dalsze kroki

Po zainstalowaniu oprogramowania ODWEK, skonfigurowaniu pliku ARSWWW.INI i przykładowych aplikacji oraz zainstalowaniu przeglądarek na stacjach roboczych można zacząć korzystać z pakietu ODWEK.



---

## Dodatek A. Funkcje API CGI

W tym rozdziale znajdują się informacje dotyczące funkcji programistycznych dostępnych w pakiecie ODWEK. Jest on skierowany głównie do programistów zajmujących się integrowaniem aplikacji ODWEK z przeglądarkami WWW.

**Uwaga:** Wartości parametrów są standardowym tekstem. Istnieje możliwość, że będą one zawierać znaki mylne dla przeglądarek. Aby zapobiec ewentualnym błędom, należy wszystkie znaki specjalne zapisywać za pomocą odpowiadających im kodów szesnastkowych. Znaki specjalne to znaki sterujące oraz niektóre symbole alfanumeryczne. Na przykład łańcuch:

Data zapisu to 31/12/95

należy przekształcić na:

Data%20zapisu%20to31%2f12%2f95

Wartości parametrów to nazwy folderów, nazwy pól folderów i kryteria wyszukiwania.

## Dodanie adnotacji

Dodaje adnotację do podanego dokumentu

### Cel

Funkcja Dodanie adnotacji umożliwia dodawanie adnotacji do określonego dokumentu. Aby dodać adnotację, użytkownik musi mieć do tego uprawnienia w grupie aplikacji OnDemand. (Uprawnienie dostępu również umożliwia dodawanie adnotacji).

### Parametry

Tabela 3. Funkcja Dodanie adnotacji

Nazwa=Wartość	Zastosowanie
<b>_function=addnote</b>	Dodanie adnotacji.
<b>_server=wartość</b>	Nazwa serwera OnDemand.
<b>_user=wartość</b>	Identyfikator użytkownika OnDemand. Użytkownik musi mieć uprawnienie do dodawania adnotacji dla każdej grupy aplikacji zawierającej dokumenty, do których ma zostać dodana adnotacja. (Uprawnienie dostępu do grupy aplikacji umożliwia dodawanie adnotacji).
<b>_password=wartość</b>	Hasło użytkownika.
<b>_folder=wartość</b>	Nazwa folderu.
<b>_perm=wartość</b>	Określa, czy adnotacja jest publiczna (0), prywatna (1) czy też prywatna dla grupy (2). Adnotacje publiczne mogą wyświetlać wszyscy użytkownicy z uprawnieniem do wyświetlania adnotacji dla grupy aplikacji. Adnotacje prywatne może wyświetlać użytkownik, który ją utworzył, administratorzy grupy aplikacji i administratorzy systemu. Adnotacje prywatne dla grupy mogą wyświetlać użytkownicy w określonej grupie, administratorzy grupy aplikacji i administratorzy systemu. Parametr <b>_group</b> zawiera nazwę grupy. Wartością domyślną jest 0 (publiczna).
<b>_group=nazwa_grupy</b>	Jeśli parametr <b>_perm</b> ma wartość 2 (prywatna dla grupy), ten parametr określa nazwę grupy.
<b>_copy=wartość</b>	Określa, czy adnotacja powinna pozostać przyłączona do dokumentu, jeśli został on wyeksportowany do innego serwera. Wartością domyślną jest Off, co oznacza, że adnotacja nie jest przyłączana do dokumentu. Wartość On oznacza, że adnotacja zostanie przyłączona do dokumentu podczas eksportowania go do innego serwera.
<b>_text=wartość</b>	Tekst adnotacji.



Tabela 3. Funkcja Dodanie adnotacji (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_html</b> =wartość	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku ADDNOTE.HTML znajdującego się w katalogu określonym przez parametr <b>TEMPLATEDIR</b> w pliku ARSWWW.INI. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze <b>TEMPLATEDIR</b>. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze <b>TEMPLATEDIR</b>.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- -AOI# Marker-- -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik <b>TEMPLATE.HTM</b> jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji dodawania adnotacji.</p>
<b>_nohtml</b> =wartość	<p>Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.</p>
<b>_docid</b> =ID dokumentu	<p>Identyfikator dokumentu, do którego ma zostać przyłączona adnotacja. Identyfikator dokumentu jest zwracany przez funkcję tworzącą listę trafień.</p>
<b>_port</b> =wartość	<p>Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowolna podana tu wartość powoduje przesłonięcie parametru <b>PORT</b> z pliku ARSWWW.INI.</p>
<b>_codepage</b> =wartość	<p>Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowolna podana tu wartość powoduje przesłonięcie parametru <b>CODEPAGE</b> z pliku ARSWWW.INI.</p>
<b>_logoff</b> =1	<p>Automatycznie rozłącza połączenie użytkownika z serwerem OnDemand po dodaniu adnotacji. Podanie tego parametru eliminuje konieczność wywołania funkcji wylogowania przez aplikację w celu rozłączenia użytkownika. Jedyną poprawną wartością tego parametru jest 1 (jeden).</p>

## Sposób użycia

Następujące parametry są wymagane:

- \_function
- \_server
- \_user
- \_password
- \_text
- \_docid

Poniższe parametry są opcjonalne:

- `_perm`
- `_group` (wymagany jeśli parametr `_perm` określa adnotację prywatną dla grupy)
- `_html`
- `_nohtml`
- `_port`
- `_codepage`
- `_logoff`

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=addnote
&_server=od400&_user=WWW&_password=WWW
&_folder=wyciągi%20z%20kart%20kredytowych
&_text=Nota%20testowa%20od%20klienta%20internetowego%20OnDemand
&_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
&_perm=1&_logoff=1
```

## Zmiana hasła

Zmienia hasło logowania się do serwera OnDemand

### Cel

Funkcja Zmiana hasła umożliwia użytkownikom zmienianie swojego hasła do serwera OnDemand.

### Parametry

Tabela 4. Funkcja Zmiana hasła

Nazwa=Wartość	Zastosowanie
<b>_function=chgpassword</b>	Zmiana hasła do serwera OnDemand dla określonego identyfikatora użytkownika
<b>_server=wartość</b>	Nazwa serwera OnDemand.
<b>_user=wartość</b>	Identyfikator użytkownika OnDemand.
<b>_password=wartość</b>	Hasło użytkownika.
<b>_new_password=wartość</b>	Nowe hasło użytkownika.
<b>_html=wartość</b>	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku CHGPASSWORD.HTML znajdującego się w katalogu określonym przez parametr TEMPLATEDIR w pliku ARSWWW.INI. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze TEMPLATEDIR. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze TEMPLATEDIR.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- -AOI# Marker-- -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik TEMPLATE.HTM jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji zmiany hasła.</p>
<b>_nohtml=wartość</b>	Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<b>_port=wartość</b>	Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowolna podana tu wartość powoduje przesłonięcie parametru PORT z pliku ARSWWW.INI.
<b>_codepage=wartość</b>	Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowolna podana tu wartość powoduje przesłonięcie parametru CODEPAGE z pliku ARSWWW.INI.

Tabela 4. Funkcja Zmiana hasła (kontynuacja)

Nazwa=Wartość	Zastosowanie
<code>_cgibin=program</code>	<p>Używany przez program CGI podczas generowania następnej strony danych wyjściowych. Jeśli zostanie określony, to strona będzie zawierała wywołanie do podanego programu zamiast to programu domyślnego (ARSWWW.CGI). Parametr ten jest używany głównie przez programistów, którzy tworzą frontowy program CGI albo serwet do tego programu lub serwetu dostarczonego przez IBM.</p> <p>Wartość <i>program</i> może określać katalog względem dyrektywy <code>ServerRoot</code> lub <i>alias</i> zdefiniowany w pliku konfiguracyjnym serwera HTTP. Domyślnie ODWEK wczytuje program CGI z katalogu CGI-BIN.</p>
<code>_logoff=1</code>	<p>Automatycznie rozłącza połączenie użytkownika z serwerem OnDemand po zmianie hasła. Podanie tego parametru eliminuje konieczność wywołania funkcji wylogowania przez aplikację w celu rozłączenia użytkownika. Jedyną poprawną wartością tego parametru jest 1 (jeden).</p>

## Sposób użycia

Następujące parametry są wymagane:

- `_function`
- `_server`
- `_user`
- `_password`
- `_new_password`

Poniższe parametry są opcjonalne:

- `_html`
- `_nohtml`
- `_port`
- `_codepage`
- `_logoff`
- `_cgibin`

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=chgpassword
&_server=od400&_user=WWW&_password=WWW
&_newpassword=nowe_haslo&_html=template.htm&_logoff=1
```

## Lista trafień dokumentów

Wyświetla listę dokumentów pasujących do kryteriów wyszukiwania

### Cel

Funkcja Lista trafień dokumentów wyświetla listę dokumentów, które pasują do kryteriów wyszukiwania dla określonego folderu. Każdy dokument jest reprezentowany jako odsyłacz do właściwego dokumentu na serwerze OnDemand. Po kliknięciu dokumentu ODWEK wczytuje go z serwera i wyświetla w oknie przeglądarki za pomocą odpowiedniego programu.

### Parametry

Tabela 5. Funkcja Lista trafień dokumentów

Nazwa=Wartość	Zastosowanie
<b>_function=dochitlist</b>	Wyświetlenie listy dokumentów pasujących do kryteriów wyszukiwania.
<b>_server=wartość</b>	Nazwa serwera OnDemand.
<b>_user=wartość</b>	Identyfikator użytkownika OnDemand.
<b>_password=wartość</b>	Hasło użytkownika.
<b>_folder=wartość</b>	Nazwa folderu.
<i>nazwa pola folderu=wartość</i>	Nazwa pola wyszukiwania w folderze oraz wartość wyszukiwania. Można podać jeden lub wiele zestawów nazw pól i wartości wyszukiwania maksymalnie do liczby pól zdefiniowanych dla folderu.
<i>nazwa pola folderu2=wartość</i>	W przypadku pól wyszukiwania w folderze z operatorami wyszukiwania BETWEEN (pomiędzy) lub NOT BETWEEN (spoza zakresu) określa większą wartość graniczną zakresu wyszukiwania.
<i>nazwa pola folderuOP=wartość</i>	Operator, który przesłania domyślny operator w polu wyszukiwania w folderze. Strona <i>wartość</i> musi mieć jedną z następujących wartości: <b>1</b> Równe (Equal) <b>2</b> Różne (Not equal) <b>4</b> Mniejsze (Less than) <b>8</b> Mniejsze lub równe (Less than or Equal) <b>16</b> Większe (Greater Than) <b>32</b> Większe lub równe (Greater Than or Equal) <b>64</b> W (In) <b>128</b> Poza (Not In) <b>256</b> Podobne (Like) <b>512</b> Niepodobne (Not Like) <b>1024</b> Pomiędzy (Between) <b>2048</b> Spoza zakresu (Not Between)
<b>_display_fields=wartość[,wartość,...]</b>	Lista oddzielona przecinkami, która zawiera nazwy pól wyświetlanych dla folderu. Można podać jedną lub wiele nazw pól. Jeśli parametr ten nie zostanie określony, strona wyjściowa będzie zawierała wszystkie pola wyświetlane dla folderu.
<b>_sort_field=wartość[,wartość,...]</b>	Określa pole wyszukiwania dla folderu, którego serwer OnDemand używa do sortowania pozycji na liście dokumentów. W przypadku określenia wielu pól należy je oddzielić przecinkami. Na przykład: <b>_sort_field=Konto,Saldo+konta,Data</b> . Domyślne pola sortowania są zdefiniowane na stronie Informacje o polu dotyczącej konkretnego folderu.
<b>_sort_order=wartość[,wartość,...]</b>	Dla każdego pola wyszukiwania w folderze określonego w parametrze <b>sort_field</b> określa, czy OnDemand sortuje pozycje od pierwszej do ostatniej czy odwrotnie. Aby posortować pozycje od pierwszej do ostatniej, należy podać <b>A</b> (ascending - rosnąco). Aby posortować pozycje od ostatniej do pierwszej (malejąco), należy podać dowolny inny znak. Na przykład: <b>_sort_order=A,D,A</b> . Domyślny porządek sortowania jest określony na stronie Informacje o polu dla danego folderu.

Tabela 5. Funkcja Lista trafień dokumentów (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_max_hits</b> =wartość	<p>Określa maksymalną liczbę pozycji, jaką ODWEK zwraca w liście dokumentów niezależnie od liczby pozycji pasujących do zapytania. Na liście dokumentów pozycje pasujące do zapytania są umieszczane w kolejności ich załadowania do bazy danych.</p> <p>ODWEK używa pierwszej z poniższych wartości w celu określenia liczby pozycji, jaką ma wyświetlić na liście dokumentów:</p> <ol style="list-style-type: none"> <li>1. Wartość w polu Maksymalna liczba trafień (określona na stronie Uprawnienia do folderu). Wartość ta zastępuje wszystkie inne wartości.</li> <li>2. Wartość parametru <b>_max_hits</b>, jeśli została określona. Wartość ta przesłania parametr MAXHITS z pliku ARSWWW.INI.</li> <li>3. Wartość parametru MAXHITS, jeśli została podana.</li> <li>4. Jeśli nie została podana żadna z powyższych wartości, ODWEK wyświetla na liście dokumentów maksymalnie 200 pozycji.</li> </ol>
<b>_html</b> =wartość	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku DOCHITLIST.HTML znajdującego się w katalogu określonym przez parametr TEMPLATEDIR w pliku ARSWWW.INI. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze TEMPLATEDIR. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze TEMPLATEDIR.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- - -AOI# Marker-- - -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik TEMPLATE.HTM jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji lista trafień dokumentów.</p>
<b>_frame</b> =wartość	<p>Dane wyjściowe tej komendy będą zawierać atrybut <b>target</b>=wartość. Parametr ten upraszcza tworzenie ramek HTML. Jest to parametr opcjonalny.</p>
<b>_datefmt</b> =wartość	<p>Określa format wartości daty używanych przez aplikację ODWEK w celu wyszukiwania w bazie danych i wyświetlania pozycji, które pasują do zapytania. Domyślny format daty określony jest na stronie Informacje o polu dla folderu. Szczegóły dotyczące formatów danych obsługiwanych przez OnDemand znajdują się w podręczniku <i>IBM Content Manager OnDemand for iSeries V5R1 Common Server Administration Guide</i>, SC27-1161.</p>
<b>_nohtml</b> =wartość	<p>Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.</p>
<b>_port</b> =wartość	<p>Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowolna podana tu wartość powoduje przesłonięcie parametru PORT z pliku ARSWWW.INI.</p>

Tabela 5. Funkcja Lista trafień dokumentów (kontynuacja)

Nazwa=Wartość	Zastosowanie
<code>_codepage=wartość</code>	Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowolna podana tu wartość powoduje przesłonięcie parametru CODEPAGE z pliku ARSWWW.INI.
<code>_sql=łańcuch</code>	<p>Określa zapytanie SQL, którego OnDemand używa do wyszukiwania w folderze. Jeśli parametr ten zostanie określony, do wyszukiwania w folderze zostanie użyte zapytanie SQL, a nie określona para nazwa pola/wartość dla folderu. OnDemand nie sprawdza poprawności łańcucha zapytania.</p> <p>Używając łańcucha SQL, należy określić nazwy pól bazy danych i ich wartości dla grupy aplikacji. W przypadku zapytań o pola z datą należy określić wewnętrzne wartości daty serwera OnDemand. Na przykład datę 1 stycznia 1999 określa się następująco: 10593. Do wyświetlenia wewnętrznej wartości dowolnej daty można użyć komendy ARSDATE.</p> <p>Łańcuch SQL służy do wyszukiwania we wszystkich grupach aplikacji znajdujących się w folderze. Jeśli łańcuch SQL zawiera nazwę pola bazy danych, które jest tylko w jednej grupie aplikacji, to zapytanie nie powiedzie się.</p>
<code>_date1=wartość</code>	Służy do określania początkowej daty zakresu dat do wyszukiwania. W przypadku określenia parametrów <code>_date1</code> i <code>_date2</code> OnDemand ogranicza zapytanie do tabeli lub tabel zawierających jedną lub obie podane daty. Format określonego w zapytaniu łańcucha daty musi pasować do formatu wyświetlania pola folderu. (Do sprawdzenia formatu wyświetlania pola folderu można użyć klienta administracyjnego).
<code>_date2=wartość</code>	Służy do określania końcowej daty zakresu dat do wyszukiwania. W przypadku określenia parametrów <code>_date1</code> i <code>_date2</code> OnDemand ogranicza zapytanie do tabeli lub tabel zawierających jedną lub obie podane daty. Format określonego w zapytaniu łańcucha daty musi pasować do formatu wyświetlania pola folderu. (Do sprawdzenia formatu wyświetlania pola folderu można użyć klienta administracyjnego).
<code>_cgibin=program</code>	<p>Używany przez program CGI podczas generowania następnej strony danych wyjściowych. Jeśli zostanie określony, to strona będzie zawierała wywołanie do podanego programu zamiast to programu domyślnego (ARSWWW.CGI). Parametr ten jest używany głównie przez programistów, którzy tworzą frontowy program CGI albo serwet do tego programu lub serwetu dostarczonego przez IBM.</p> <p>Wartość <i>program</i> może określać katalog względem dyrektywy <code>ServerRoot</code> lub <i>alias</i> zdefiniowany w pliku konfiguracyjnym serwera HTTP. Domyślnie ODWEK wczytuje program CGI z katalogu <code>/QIBM/Proddata/OnDemand/www/bin</code>.</p>
<code>_or=wartość</code>	W celu połączenia pól wyszukiwania operatorem logicznym OR, należy podać 1 (jeden); pozycja musi pasować co najmniej do jednej z podanych wartości wyszukiwania. Wartością domyślną jest 0 (zero); oznacza ona, że OnDemand ma łączyć pola wyszukiwania za pomocą operatora logicznego AND (pozycja musi pasować do wszystkich podanych wartości wyszukiwania).
<code>_logoff=1</code>	Automatycznie rozłącza połączenie użytkownika z serwerem OnDemand po utworzeniu listy dokumentów. Podanie tego parametru eliminuje konieczność wywoływania funkcji wylogowania przez aplikację w celu rozłączenia użytkownika. Jedyną poprawną wartością tego parametru jest 1 (jeden).

## Sposób użycia

Następujące parametry są wymagane:

- `_function`
- `_server`
- `_user`
- `_password`
- `_folder`

Poniższe parametry są opcjonalne:

*nazwa pola folderu*  
*nazwa pola folderu2*  
*nazwa pola folderuOP*  
*\_display\_fields*  
*\_sort\_field*  
*\_sort\_order*  
*\_max\_hits*  
*\_frame*  
*\_datefmt*  
*\_sql*  
*\_date1*  
*\_date2*  
*\_or*  
*\_html*  
*\_nohtml*  
*\_port*  
*\_codepage*  
*\_logoff*  
*\_cgibin*

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=dochitlist  
&_server=od400&_user=WWW&_password=WWW  
&_folder=wyciagi%20z%20kart%20kredytowych  
&numer%20konta=1000100010009999&date=1%2f1%2f96&date2=12%2f31%2f96  
&nameOP=256&name=%AA  
&_sort_field=Konto,Saldo%20konta,Data&_sort_order=A,D,A  
&_logoff=1  
&_html=template.htm
```



## Wylogowanie się

Wylogowuje z serwera OnDemand

### Cel

Funkcja Wylogowanie się próbuje wylogować użytkownika z serwera OnDemand. Nazwa serwera i identyfikator użytkownika, którego należy wylogować, są umieszczane w informacjach cookie przeglądarki klienta przez funkcję Zalogowanie się. Jeśli serwer nie jest poprawnym serwerem OnDemand, zostanie zwrócony komunikat o błędzie. Jeśli identyfikator użytkownika nie jest zalogowany na podanym serwerze, zostanie zwrócony komunikat o błędzie.

### Parametry

Tabela 6. Funkcja Wylogowanie się

Nazwa=Wartość	Zastosowanie
<code>_function=logoff</code>	Wylogowuje z serwera OnDemand.
<code>_html=wartość</code>	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku LOGOFF.HTML znajdującego się w katalogu określonym przez parametr <code>TEMPLATEDIR</code> w pliku <code>ARSWWW.INI</code>. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze <code>TEMPLATEDIR</code>. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze <code>TEMPLATEDIR</code>.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- - -AOI# Marker-- - -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik <code>TEMPLATE.HTM</code> jest przykładowym plikiem szablonu dostarczanym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji wylogowania.</p>
<code>_nohtml=wartość</code>	Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<code>_port=wartość</code>	Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowolna podana tu wartość powoduje przesłonięcie parametru <code>PORT</code> z pliku <code>ARSWWW.INI</code> .

### Sposób użycia

Następujące parametry są wymagane:

`_function`

Poniższe parametry są opcjonalne:

\_html  
\_nohtml  
\_port

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=logoff  
&_html=template.htm
```

## Zalogowanie się

Zalogowuje użytkownika do serwera OnDemand

### Cel

Funkcja Zalogowanie się próbuje uzyskać dostęp do serwera OnDemand za pomocą wartości parametrów określających serwer, użytkownika i hasło. Sprawdza ona, czy podany użytkownik ma uprawnienie do zalogowania się na podanym serwerze oraz jego hasło. Jeśli użytkownik nie ma uprawnień do logowania się na podanym serwerze, zostanie zwrócony komunikat o błędzie. Jeśli serwer nie jest poprawnym serwerem OnDemand, zostanie zwrócony komunikat o błędzie. Jeśli hasło użytkownika nie jest poprawne, zostanie zwrócony komunikat o błędzie. Po udanym zalogowaniu się funkcja Zalogowanie się wyświetla stronę WWW zawierającą listę folderów, do których użytkownik ma dostęp.

### Parametry

Tabela 7. Funkcja Zalogowanie się

Nazwa=Wartość	Zastosowanie
<b>_function=logon</b>	Zalogowuje użytkownika do serwera OnDemand.
<b>_server=wartość</b>	Nazwa serwera OnDemand.
<b>_user=wartość</b>	Identyfikator użytkownika OnDemand.
<b>_password=wartość</b>	Hasło użytkownika.
<b>_new_password=wartość</b>	Nowe hasło użytkownika. Umożliwia zmianę hasła po udanym zalogowaniu się do serwera OnDemand. Jest to parametr opcjonalny.
<b>_html=wartość</b>	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku LOGON.HTML znajdującego się w katalogu określonym przez parametr <b>TEMPLATEDIR</b> w pliku ARSWWW.INI. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze <b>TEMPLATEDIR</b>. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze <b>TEMPLATEDIR</b>.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- -AOI# Marker-- -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik <b>TEMPLATE.HTM</b> jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji zalogowania.</p>
<b>_frame=wartość</b>	Dane wyjściowe tej komendy będą zawierać atrybut <b>target=wartość</b> . Parametr ten upraszcza tworzenie ramek HTML. Jest to parametr opcjonalny.
<b>_datefmt=wartość</b>	Określa format wartości daty używanych przez aplikację ODWEK w celu wyszukiwania w bazie danych i wyświetlania pozycji, które pasują do zapytania. Domyślny format daty określony jest na stronie Informacje o polu dla folderu. Szczegóły dotyczące formatów danych obsługiwanych przez OnDemand znajdują się w podręczniku <i>IBM Content Manager OnDemand for iSeries V5R1 Common Server Administration Guide</i> , SC27-1161.

Tabela 7. Funkcja Zalogowanie się (kontynuacja)

Nazwa=Wartość	Zastosowanie
<code>_nohtml=wartość</code>	Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<code>_port=wartość</code>	Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowlolna podana tu wartość powoduje przesłonięcie parametru PORT z pliku ARSWWW.INI.
<code>_codepage=wartość</code>	Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowlolna podana tu wartość powoduje przesłonięcie parametru CODEPAGE z pliku ARSWWW.INI.
<code>_cgibin=program</code>	<p>Używany przez program CGI podczas generowania następnej strony danych wyjściowych. Jeśli zostanie określony, to strona będzie zawierała wywołanie do podanego programu zamiast to programu domyślnego (ARSWWW.CGI). Parametr ten jest używany głównie przez programistów, którzy tworzą frontowy program CGI albo serwlet do tego programu lub serwletu dostarczonego przez IBM.</p> <p>Wartość <i>program</i> może określać katalog względem dyrektywy <code>ServerRoot</code> lub <i>alias</i> zdefiniowany w pliku konfiguracyjnym serwera HTTP. Domyślnie ODWEK wczytuje program CGI z katalogu CGI-BIN.</p>

## Sposób użycia

Następujące parametry są wymagane:

- `_function`
- `_server`
- `_user`
- `_password`

Poniższe parametry są opcjonalne:

- `_new_password`
- `_frame`
- `_datefmt`
- `_html`
- `_nohtml`
- `_port`
- `_codepage`
- `_logoff`
- `_cgibin`

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=logon
&_server=od400&_user=WWW&_password=WWW
&_html=template.htm
```

## Drukowanie dokumentu (serwer)

Wysyła jeden lub wiele dokumentów do podanej drukarki serwerowej

### Cel

Funkcja Drukowanie dokumentu wysyła kopie dokumentu do drukarki serwera OnDemand. Aby skorzystać z funkcji wydruku serwerowego, użytkownik musi mieć do tego uprawnienia w grupie aplikacji OnDemand. (Uprawnienie dostępu również umożliwia drukowanie dokumentów). Na podanym serwerze OnDemand musi być zdefiniowana przynajmniej jedna drukarka serwerowa.

### Parametry

Tabela 8. Funkcja Drukowanie dokumentu

Nazwa=Wartość	Zastosowanie
<b>_function=printdocs</b>	Drukuje dokumenty.
<b>_server=wartość</b>	Nazwa serwera OnDemand.
<b>_user=wartość</b>	Identyfikator użytkownika OnDemand. Użytkownik musi mieć uprawnienie do drukowania dokumentów dla każdej grupy aplikacji zawierającej dokumenty do wydrukowania. (Uprawnienie dostępu do grupy aplikacji umożliwia drukowanie dokumentów).
<b>_password=wartość</b>	Hasło użytkownika.
<b>_folder=wartość</b>	Nazwa folderu.
<b>_printer=wartość</b>	Nazwa drukarki na serwerze OnDemand.  Jeśli podana drukarka jest faksem lub drukarką z informacjami, można podać następujące parametry:  <b>_recv_name=wartość</b> Nazwa odbiorcy.  <b>_recv_comp=wartość</b> Nazwa firmy odbiorcy.  <b>_recv_fax=wartość</b> Numer faksu odbiorcy.  <b>_send_name=wartość</b> Nazwa nadawcy.  <b>_send_comp=wartość</b> Nazwa firmy nadawcy.  <b>_send_tel=wartość</b> Numer telefonu nadawcy.  <b>_send_fax=wartość</b> Numer faksu nadawcy.  <b>_send_cover=wartość</b> Zdefiniowana przez użytkownika nakładka, którą program obsługi wyjścia strony poprzedzającej scala z wartościami innych parametrów w celu utworzenia strony tytułowej dokumentu.  <b>_subject=wartość</b> Łańcuch odpowiadający tematowi dokumentu.  <b>_notes=wartość</b> Łańcuch określający uwagę dotyczącą dokumentu.

Tabela 8. Funkcja Drukowanie dokumentu (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_html</b> =wartość	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku PRINTDOCS.HTML znajdującego się w katalogu określonym przez parametr <b>TEMPLATEDIR</b> w pliku ARSWWW.INI. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze <b>TEMPLATEDIR</b>. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze <b>TEMPLATEDIR</b>.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- -AOI# Marker-- -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik <b>TEMPLATE.HTM</b> jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji drukowania dokumentów.</p>
<b>_nohtml</b> =wartość	<p>Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.</p>
<b>_docids</b> =lista_ID_dokumentów	<p>Lista identyfikatorów dokumentów, które mają zostać wydrukowane. Identyfikatory dokumentów są zwracane przez funkcję tworzącą listę trafień. Określając wiele identyfikatorów dokumentów należy oddzielić je znakami \003.</p> <p><b>Uwaga:</b> Jeśli liczba identyfikatorów dokumentów przekroczy 200, konieczne jest określenie parametru <b>_max_hits</b>.</p>
<b>_port</b> =wartość	<p>Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowolna podana tu wartość powoduje przesłonięcie parametru <b>PORT</b> z pliku ARSWWW.INI.</p>
<b>_codepage</b> =wartość	<p>Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowolna podana tu wartość powoduje przesłonięcie parametru <b>CODEPAGE</b> z pliku ARSWWW.INI.</p>

Tabela 8. Funkcja Drukowanie dokumentu (kontynuacja)

Nazwa=Wartość	Zastosowanie
<code>_max_hits</code> =wartość	<p>Parametr ten służy do określania liczby identyfikatorów dokumentów do przetworzenia. Należy podać wartość równą liczbie identyfikatorów dokumentów określonych w parametrze <code>_docids</code> lub większą.</p> <p><b>Uwaga:</b> Jeśli liczba identyfikatorów dokumentów przekroczy wartość podaną w parametrze <code>MAXHITS</code> w pliku <code>ARSWWW.CGI</code> (lub 200, jeśli nie została ona określona), konieczne jest określenie wartości parametru <code>_max_hits</code>. Jeśli parametr <code>_max_hits</code> nie zostanie określony (lub jeśli zostanie podana wartość parametru <code>MAXHITS</code>), zostanie przetworzonych maksymalnie 200 identyfikatorów dokumentów niezależnie od liczby identyfikatorów podanych w parametrze <code>_docids</code>.</p> <p>W celu określenia liczby identyfikatorów dokumentów do przetworzenia ODWEK używa jednej z poniższych wartości:</p> <ul style="list-style-type: none"> <li>• Wartość parametru <code>_max_hits</code>, jeśli została określona. Przesłania ona wartość parametru <code>MAXHITS</code>.</li> <li>• Wartość parametru <code>MAXHITS</code>, jeśli została podana.</li> <li>• Jeśli nie została podana żadna z powyższych wartości, ODWEK przetwarza maksymalnie 200 identyfikatorów dokumentów.</li> </ul>
<code>_logoff</code> =1	<p>Automatycznie rozłącza połączenie użytkownika z serwerem OnDemand po wydrukowaniu dokumentu. Podanie tego parametru eliminuje konieczność wywoływania funkcji wylogowania przez aplikację w celu rozłączenia użytkownika. Jedyną poprawną wartością tego parametru jest 1 (jeden).</p>

## Sposób użycia

Następujące parametry są wymagane:

`_function`  
`_server`  
`_user`  
`_password`  
`_folder`  
`_printer`  
`_docids`

Poniższe parametry są opcjonalne:

`_recv_name`  
`_recv_comp`  
`_recv_fax`  
`_send_name`  
`_send_comp`  
`_send_tel`  
`_send_fax`  
`_send_cover`  
`_subject`  
`_notes`  
`_max_hits`  
`_html`  
`_nohtml`  
`_port`  
`_codepage`  
`_logoff`

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=printdocs
&_server=od400&_user=WWW&_password=WWW
&_folder=wyciągi%20z%20kart%20kredytowych
&_printer=infoprint60
&_docids=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
&_logoff=1
```



## Wczytanie dokumentu

Wczytuje wybrany dokument z serwera OnDemand

### Cel

Funkcja Wczytanie dokumentu wczytuje wybrany dokument z serwera OnDemand. ODWEK wyświetla go w oknie przeglądarki WWW za pomocą apletu, przeglądarki lub innego programu powiązanego z typem dokumentu.

### Parametry

Tabela 9. Funkcja Wczytanie dokumentu

Nazwa=Wartość	Zastosowanie
<b>_function=retrieve</b>	Wczytuje wybrany dokument.
<b>_server=wartość</b>	Nazwa serwera OnDemand.
<b>_user=wartość</b>	Identyfikator użytkownika OnDemand.
<b>_password=wartość</b>	Hasło użytkownika.
<b>_folder=wartość</b>	Nazwa folderu.
<i>nazwa pola folderu=wartość</i>	Nazwa pola wyszukiwania w folderze oraz wartość wyszukiwania. Można podać jeden lub wiele zestawów nazw pól i wartości wyszukiwania maksymalnie do liczby pól zdefiniowanych dla folderu.
<b>_html=wartość</b>	<p>Gdy wystąpi błąd podczas wczytywania dokumentu, parametr ten określa plik HTML, którego ODWEK używa jako szablonu w celu wygenerowania strony WWW z komunikatem o błędzie. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku RETRIEVE.HTML znajdującego się w katalogu określonym przez parametr TEMPLATEDIR w pliku ARSWWW.INI. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze TEMPLATEDIR. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze TEMPLATEDIR.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- - -AOI# Marker-- - -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik TEMPLATE.HTM jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji wczytania.</p>
<b>_nohtml=wartość</b>	Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<b>_port=wartość</b>	Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowolna podana tu wartość powoduje przesłonięcie parametru PORT z pliku ARSWWW.INI.

Tabela 9. Funkcja Wczytanie dokumentu (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_codepage=wartość</b>	Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowolna podana tu wartość powoduje przesłonięcie parametru CODEPAGE z pliku ARSWWW.INI.
<b>_cgibin=program</b>	<p>Używany przez program CGI podczas generowania następnej strony danych wyjściowych. Jeśli zostanie określony, to strona będzie zawierała wywołanie do podanego programu zamiast to programu domyślnego (ARSWWW.CGI). Parametr ten jest głównie używany przez programistów, którzy tworzą frontowy program CGI lub serwet do tego programu lub serwetu dostarczonego przez IBM.</p> <p>Wartość <i>program</i> może określać katalog względem dyrektywy <i>ServerRoot</i> lub <i>alias</i> zdefiniowany w pliku konfiguracyjnym serwera HTTP. Domyślnie ODWEK czytuje program CGI z katalogu CGI-BIN.</p>
<b>_or=wartość</b>	W celu połączenia pól wyszukiwania operatorem logicznym OR, należy podać 1 (jeden); pozycja musi pasować co najmniej do jednej z podanych wartości wyszukiwania. Wartością domyślną jest 0 (zero); oznacza ona, że OnDemand ma łączyć pola wyszukiwania za pomocą operatora logicznego AND (pozycja musi pasować do wszystkich podanych wartości wyszukiwania).
<b>_afp=wartość</b>	<p>Podczas wczytywania dokumentu AFP z serwera OnDemand wartość tego parametru określa, jakie czynności powinna wykonać aplikacja ODWEK przed wysłaniem dokumentu do klienta. Na przykład niektórzy użytkownicy przekształcają dokumenty AFP do formatu HTML za pomocą usługi AFP2WEB Transform i używają apletu AFP2HTML w celu wyświetlenia danych wyjściowych HTML. Powinni oni określić parametr <b>_afp=HTML</b>, aby aplikacja ODWEK przekształcała dokument AFP przed wysłaniem go do klienta.</p> <p>Wartością <i>wartość</i> może być:</p> <p><b>ASCII</b> ODWEK przekształca dokument AFP na tekst ASCII.</p> <p><b>HTML</b> ODWEK przekształca dokument AFP w dokument HTML przy użyciu usługi AFP2WEB Transform.</p> <p><b>NATIVE</b> ODWEK wyodrębnia i dekompresuje dokument AFP i jego zasoby z serwera OnDemand.  <b>Uwaga:</b> Jeśli zostanie podana wartość <b>_afp=NATIVE</b>, należy sprawdzić, czy typ zawartości MIME określa żadaną przeglądarkę (więcej informacji znajduje się w sekcji “[MIMETYPES]” na stronie 25).</p> <p><b>PDF</b> ODWEK przekształca dokument AFP w dokument PDF przy użyciu usługi AFP2WEB Transform.</p> <p><b>PLUGIN</b> ODWEK nie przekształca dokumentu AFP (jest to zachowanie domyślne).</p>
<b>_email=wartość</b>	<p>Podczas wczytywania dokumentu EMAIL z serwera OnDemand wartość tego parametru określa, jakie czynności powinna wykonać aplikacja ODWEK przed wysłaniem dokumentu do klienta. Wartością <i>wartość</i> może być:</p> <p><b>NATIVE</b> ODWEK wyodrębnia i dekompresuje dokument EMAIL z serwera OnDemand.  <b>Uwaga:</b> Jeśli zostanie podana wartość <b>_email=NATIVE</b>, należy sprawdzić, czy typ zawartości MIME określa żadaną przeglądarkę (więcej informacji znajduje się w sekcji “[MIMETYPES]” na stronie 25).</p> <p><b>HTML</b> ODWEK przekształca dokument EMAIL do formatu HTML.</p>

Tabela 9. Funkcja Wczytanie dokumentu (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_line</b> =wartość	<p>Podczas wczytywania dokumentu zawierającego dane dla drukarki wierszowej z serwera OnDemand wartość tego parametru określa, jakie czynności powinna wykonać aplikacja ODWEK przed wysłaniem dokumentu do klienta. Wartością <i>wartość</i> może być:</p> <p><b>APPLET</b> ODWEK przekształca dokument zawierający dane dla drukarki wierszowej za pomocą apletu Line Data (zachowanie domyślne).</p> <p><b>ASCII</b> ODWEK przekształca dokument zawierający dane dla drukarki wierszowej w tekst ASCII.</p> <p><b>NATIVE</b> ODWEK wyodrębnia i dekompresuje dokument zawierający dane dla drukarki wierszowej z serwera OnDemand.  <b>Uwaga:</b> Jeśli zostanie podana wartość <b>_line=NATIVE</b>, należy sprawdzić, czy typ zawartości MIME określa żadaną przeglądarkę (więcej informacji znajduje się w sekcji “[MIMETYPES]” na stronie 25).</p>
<b>_docid</b> =ID dokumentu	Identyfikator dokumentu do wczytania. Identyfikator dokumentu jest zwracany przez funkcję tworzącą listę trafień.
<b>_logoff</b> =1	Automatycznie rozłącza połączenie użytkownika z serwerem OnDemand po wczytaniu dokumentu. Podanie tego parametru eliminuje konieczność wywołania funkcji wylogowania przez aplikację w celu rozłączenia użytkownika. Jedyną poprawną wartością tego parametru jest 1 (jeden).

## Sposób użycia

Następujące parametry są wymagane:

- \_function
- \_server
- \_user
- \_password
- \_folder

Poniższe parametry są opcjonalne:

- nazwa pola folderu*
- \_docid
- \_or
- \_afp
- \_email
- \_line
- \_html
- \_nohtml
- \_port
- \_codepage
- \_logoff
- \_cgibin

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=retrieve
&_server=od400&_user=WWW&_password=WWW
&_folder=wyciagi%20z%20kart%20kredytowych
&numer%20konta=1000100010009999&data=1%2f1%2f96
&_html=template.htm&_logoff=1
```

## Kryteria wyszukiwania

Wyświetla kryteria wyszukiwania dla określonego folderu

### Cel

Funkcja kryteria wyszukiwania wyświetla kryteria wyszukiwania dotyczące określonego folderu za pomocą formularza. Użytkownik może zaakceptować domyślne kryteria wyszukiwania lub wprowadzić nowe, aby wyszukać określone dokumenty. Po kliknięciu przycisku Wyślij ODWEK wyświetla stronę WWW z listą dokumentów, które spełniają kryteria wyszukiwania.

### Parametry

Tabela 10. Funkcja Kryteria wyszukiwania

Nazwa=Wartość	Zastosowanie
<code>_function=searchcrit</code>	Wyświetla kryteria wyszukiwania dla określonego folderu.
<code>_server=wartość</code>	Nazwa serwera OnDemand.
<code>_user=wartość</code>	Identyfikator użytkownika OnDemand.
<code>_password=wartość</code>	Hasło użytkownika.
<code>_folder=wartość</code>	Nazwa folderu do wyszukiwania.
<code>_html=wartość</code>	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku SEARCHCRIT.HTML znajdującego się w katalogu określonym przez parametr <code>TEMPLATEDIR</code> w pliku <code>ARSWWW.INI</code>. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze <code>TEMPLATEDIR</code>. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w zmiennej <code>TEMPLATEDIR</code>.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- -AOI# Marker-- -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik <code>TEMPLATE.HTM</code> jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji określającej kryteria wyszukiwania.</p>
<code>_frame=wartość</code>	Dane wyjściowe tej komendy będą zawierać atrybut <code>target=wartość</code> . Parametr ten upraszcza tworzenie ramek HTML. Jest to parametr opcjonalny.
<code>_datefmt=wartość</code>	Określa format wartości daty używanych przez aplikację ODWEK w celu wyszukiwania w bazie danych i wyświetlania pozycji, które pasują do zapytania. Domyślny format daty określony jest na stronie Informacje o polu dla folderu. Szczegóły dotyczące formatów danych obsługiwanych przez OnDemand znajdują się w podręczniku <i>IBM Content Manager OnDemand for iSeries V5R1 Common Server Administration Guide</i> , SC27-1161.

Tabela 10. Funkcja Kryteria wyszukiwania (kontynuacja)

Nazwa=Wartość	Zastosowanie
<code>_nohtml=wartość</code>	Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<code>_port=wartość</code>	Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowolna podana tu wartość powoduje przesłonięcie parametru PORT z pliku ARSWWW.INI.
<code>_codepage=wartość</code>	Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowolna podana tu wartość powoduje przesłonięcie parametru CODEPAGE z pliku ARSWWW.INI.
<code>_cgibin=program</code>	<p>Używany przez program CGI podczas generowania następnej strony danych wyjściowych. Jeśli zostanie określony, to strona będzie zawierała wywołanie do podanego programu zamiast to programu domyślnego (ARSWWW.CGI). Parametr ten jest używany głównie przez programistów, którzy tworzą frontowy program CGI albo serwlet do tego programu lub serwletu dostarczonego przez IBM.</p> <p>Wartość <i>program</i> może określać katalog względem dyrektywy <code>ServerRoot</code> lub <i>alias</i> zdefiniowany w pliku konfiguracyjnym serwera HTTP. Domyślnie ODWEK wczytuje program CGI z katalogu CGI-BIN.</p>
<code>_logoff=1</code>	Automatycznie rozłącza połączenie użytkownika z serwerem OnDemand po wyświetleniu kryteriów wyszukiwania. Podanie tego parametru eliminuje konieczność wywołania funkcji wylogowania przez aplikację w celu rozłączenia użytkownika. Jedyną poprawną wartością tego parametru jest 1 (jeden).

## Sposób użycia

Następujące parametry są wymagane:

- `_function`
- `_server`
- `_user`
- `_password`
- `_folder`

Poniższe parametry są opcjonalne:

- `_frame`
- `_datefmt`
- `_html`
- `_nohtml`
- `_port`
- `_codepage`
- `_logoff`
- `_cgibin`

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=searchcrit
&_server=od400&_user=WWW&_password=WWW
&_folder=wyciagi%20z%20kart%20kredytowych&_html=template.htm
&_logoff=1
```

## Aktualizowanie dokumentu

Aktualizuje jedną lub wiele wartości bazy danych dla podanego dokumentu

### Cel

Funkcja Aktualizowanie dokumentu umożliwia autoryzowanym użytkownikom aktualizowanie dokumentów. Powoduje ona aktualizację jednej lub wielu wartości bazy danych dla podanego dokumentu.

### Parametry

Tabela 11. Funkcja Aktualizowanie dokumentu

Nazwa=Wartość	Zastosowanie
<b>_function=updatedoc</b>	Aktualizuje bazę danych.
<b>_server=wartość</b>	Nazwa serwera OnDemand.
<b>_user=wartość</b>	Identyfikator użytkownika OnDemand. Użytkownik musi mieć uprawnienia do aktualizowania dokumentów dla grupy aplikacji.
<b>_password=wartość</b>	Hasło użytkownika.
<b>_folder=wartość</b>	Nazwa folderu.
<i>nazwa pola folderu=wartość</i>	Nazwa pola, które ma zostać zaktualizowane, oraz wartość, jaka ma znaleźć się w tym polu. Można podać jeden lub wiele zestawów nazw pól i wartości, maksymalnie do liczby pól zdefiniowanych dla folderu.
<b>_html=wartość</b>	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku UPDATE.HTML znajdującego się w katalogu określonym przez parametr <b>TEMPLATEDIR</b> w pliku ARSWWW.INI. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze <b>TEMPLATEDIR</b>. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze <b>TEMPLATEDIR</b>.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- - -AOI# Marker-- - -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik <b>TEMPLATE.HTM</b> jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji aktualizacji.</p>
<b>_nohtml=wartość</b>	Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<b>_docid=ID_dokumentu</b>	Identyfikator dokumentu do aktualizacji. Identyfikator dokumentu jest zwracany przez funkcję tworzącą listę trafień.

Tabela 11. Funkcja Aktualizowanie dokumentu (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_port=wartość</b>	Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowolna podana tu wartość powoduje przesłonięcie parametru PORT z pliku ARSWWW.INI.
<b>_codepage=wartość</b>	Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowolna podana tu wartość powoduje przesłonięcie parametru CODEPAGE z pliku ARSWWW.INI.
<b>_logoff=1</b>	Automatycznie rozłącza połączenie użytkownika z serwerem OnDemand po zaktualizowaniu dokumentu. Podanie tego parametru eliminuje konieczność wywoływania funkcji wylogowania przez aplikację w celu rozłączenia użytkownika. Jedyną poprawną wartością tego parametru jest 1 (jeden).

## Sposób użycia

Następujące parametry są wymagane:

`_function`  
`_server`  
`_user`  
`_password`  
`_folder`

Poniższe parametry są opcjonalne:

*nazwa pola folderu*  
`_docid`  
`_html`  
`_nohtml`  
`_port`  
`_codepage`  
`_logoff`

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=updatedoc
&_server=od400&_user=WWW&_password=WWW
&_folder=wyciagi%20z%20kart%20kredytowych
&numer%20konta=1000100010009999
&_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
&_html=template.htm&_logoff=1
```

## Podgląd adnotacji

Wyświetla adnotacje załączone do podanego dokumentu

### Cel

Funkcja Podgląd adnotacji umożliwia wyświetlanie adnotacji powiązanych z określonym dokumentem. Aby wyświetlić adnotacje, użytkownik musi mieć do tego uprawnienia w grupie aplikacji OnDemand. (Uprawnienie dostępu również umożliwia wyświetlanie adnotacji).

### Parametry

Tabela 12. Funkcja Podgląd adnotacji

Nazwa=Wartość	Zastosowanie
<b>_function=getnotes</b>	Wyświetla adnotacje.
<b>_server=wartość</b>	Nazwa serwera OnDemand.
<b>_user=wartość</b>	Identyfikator użytkownika OnDemand. Użytkownik musi mieć uprawnienie do wyświetlania adnotacji dla każdej grupy aplikacji zawierającej adnotacje, które mają zostać wyświetlone. (Uprawnienie dostępu do grupy aplikacji umożliwia wyświetlanie adnotacji).
<b>_password=wartość</b>	Hasło użytkownika.
<b>_folder=wartość</b>	Nazwa folderu.
<b>_html=wartość</b>	<p>Określa plik HTML, którego ODWEK używa jako szablonu do generowania wyjściowej strony WWW. Wartością może być nazwa pliku lub * (gwiazdka). Jeśli wartością jest gwiazdka, ODWEK używa pliku GETNOTES.HTML znajdującego się w katalogu określonym przez parametr TEMPLATEDIR w pliku ARSWWW.INI. Jeśli wartością jest nazwa pliku bez nazwy ścieżki, plik ten musi znajdować się w katalogu określonym w parametrze TEMPLATEDIR. Jeśli wartość zawiera nazwę ścieżki, jest ona traktowana jako względna w stosunku do katalogu określonego w parametrze TEMPLATEDIR.</p> <p>Zawartość pliku HTML jest definiowana przez użytkownika. Musi on jednak zawierać następujący wiersz komentarza:</p> <pre>&lt;!-- -AOI# Marker-- -&gt;</pre> <p>Położenie wiersza z komentarzem określa miejsce, w którym ODWEK umieszcza dane wyjściowe. Wszystkie wiersze powyżej tego wiersza zostaną wyświetlone przed danymi wyjściowymi wygenerowanymi przez program ODWEK. Wszystkie wiersze poniżej tego wiersza zostaną wyświetlone po danych wyjściowych wygenerowanych przez program ODWEK.</p> <p>Plik TEMPLATE.HTM jest przykładowym plikiem szablonu dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji podglądu adnotacji.</p>
<b>_nohtml=wartość</b>	Określa typ danych wyjściowych generowanych przez aplikację ODWEK. Wartością domyślną jest 0 (zero); oznacza ona, że ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości 1 (jeden) oznacza, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 137 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<b>_docid=ID dokumentu</b>	Identyfikator dokumentu, który zawiera adnotacje do wyświetlenia. Identyfikator dokumentu jest zwracany przez funkcję tworzącą listę trafień.



Tabela 12. Funkcja Podgląd adnotacji (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_port=wartość</b>	Numer portu serwera OnDemand. Wartość domyślna 0 (zero) oznacza, że serwer ma używać portu podanego w Tabeli usług (WRKSRVTBLE). Jeśli w Tabeli usług nie ma odpowiedniego numeru portu, serwer OnDemand będzie próbował użyć portu 1445. Dowlolna podana tu wartość powoduje przesłonięcie parametru PORT z pliku ARSWWW.INI.
<b>_codepage=wartość</b>	Strona kodowa bazy danych serwera OnDemand. Domyślnie używana jest strona kodowa serwera HTTP. Jeśli strona kodowa serwera jest inna niż strona kodowa bazy danych, należy ją określić. Dowlolna podana tu wartość powoduje przesłonięcie parametru CODEPAGE z pliku ARSWWW.INI.
<b>_logoff=1</b>	Automatycznie rozłącza połączenie użytkownika z serwerem OnDemand po wyświetleniu adnotacji. Podanie tego parametru eliminuje konieczność wywołania funkcji wylogowania przez aplikację w celu rozłączenia użytkownika. Jedyną poprawną wartością tego parametru jest 1 (jeden).

## Sposób użycia

Następujące parametry są wymagane:

- \_function
- \_server
- \_user
- \_password
- \_folder
- \_docid

Poniższe parametry są opcjonalne:

- \_html
- \_nohtml
- \_port
- \_codepage
- \_logoff

## Sample Function Call

```
http://www.company.com/cgi-bin/arswww.cgi?_function=getnotes
&_server=od400&_user=WWW&_password=WWW
&_folder=wyciagi%20z%20kart%20kredytowych
&_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
&_logoff=1
```



---

## Dodatek B. Serwlety w języku Java

Serwlet Java działa jako kontroler aplikacji WWW, wykonując funkcje i typowe zadania przed realizacją określonego działania i po nim, na przykład zadania związane z zarządzaniem połączeniem z serwerem OnDemand.

Dostępne są funkcje obsługujące następujące typowe zadania:

- zalogowanie i wylogowanie się,
- wyszukiwanie,
- wczytywanie, drukowanie i aktualizacja dokumentów,
- dodawanie i przeglądanie adnotacji,
- zmiana hasła

Aby użyć serwletu w danej aplikacji, należy zastosować zestaw funkcji i parametrów tej aplikacji.

Serwlet Java stosuje te same funkcje, co program CGI. Spis funkcji, opisy i parametry zawiera Dodatek A, "Funkcje API CGI", na stronie 59.



---

## Dodatek C. Funkcje API języka Java

Dokumentacja funkcji API języka Java jest dostępna w formacie HTML razem z oprogramowaniem ODWEK.

Aby można było ją wyświetlić, należy w systemie zainstalować oprogramowanie ODWEK, a następnie wyodrębnić pliki dokumentacji z pliku ODApiDoc.zip do katalogu /QIBM/ProdData/OnDemand/www/. Metoda wyodrębniania powinna zachowywać strukturę katalogów, w której w archiwum zapisane są pliki.

Aby wyświetlić dokumentację, po wyodrębnieniu plików otwórz w przeglądarce WWW plik index.html.



---

## Dodatek D. Podręcznik programowania funkcji API w języku Java

Aplikacyjne interfejsy programistyczne języka Java to zestaw klas, które służą do obsługi danych na serwerze OnDemand. W tej sekcji opisano funkcje API języka Java, implementacje funkcji dokumentów w języku Java oraz funkcje obsługujące połączenia z Internetem.

Funkcje API języka Java obsługują:

- Model COM dostępu do danych.
- Funkcje wyszukiwania i aktualizacji na serwerach OnDemand. **Uwaga:** Rozdział 1, “Przegląd”, na stronie 1 opisuje ograniczenia związane z dostępem do serwera OnDemand w przypadku serwera z systemem OS/390 w wersji 2.
- Implementacje klient/serwer dla użytkowników aplikacji w języku Java.

---

### Architektura klient/serwer

Funkcje API stanowią wygodny interfejs programistyczny dla użytkowników aplikacji. Mogą one rezydować zarówno na serwerze, jak i kliencie OnDemand (z tym samym interfejsem w obu przypadkach), a aplikacje mogą znajdować się na systemie lokalnym lub zdalnym. Funkcje API klienta komunikują się z serwerem w celu uzyskania dostępu do danych przez sieć. Komunikację pomiędzy klientem i serwerem realizują klasy; nie ma potrzeby używania dodatkowych programów.

Klasy funkcji API znajdują się w jednym pakiecie: `com.ibm.edms.od`.

---

### Tworzenie pakietów w środowisku języka Java

Klasy funkcji API znajdują się w jednym pakiecie: `com.ibm.edms.od`. Są to:

#### **`com.ibm.edms.od.ODCallback`**

Klasa ta jest używana razem ze wszystkimi metodami, w których serwer zwraca dane w trakcie przetwarzania.

#### **`com.ibm.edms.od.ODCriteria`**

Klasa ta określa kryteria wyszukiwania z folderu OnDemand. Zawiera ona metody służące do ustawiania operatora wyszukiwania i wartości wyszukiwania.

#### **`com.ibm.edms.od.ODException`**

Klasa ta określa wyjątki, które mogą pojawić się podczas korzystania z funkcji API.

#### **`com.ibm.edms.od.ODFolder`**

Klasa ta określa folder OnDemand. Obiekt ten zwracany jest po udanym wywołaniu metody `ODServer.openFolder()`. Ta klasa zawiera informacje dotyczące kryteriów wyszukiwania dla folderu. Obiekty te modyfikuje się w celu zawężenia zapytania kierowanego do serwera.

#### **`com.ibm.edms.od.ODHit`**

Klasa ta określa dokument OnDemand.

#### **`com.ibm.edms.od.ODNote`**

Klasa ta określa adnotację OnDemand.

#### **`com.ibm.edms.od.ODServer`**

Klasa ta określa połączenie z serwerem OnDemand. Za jej pomocą można zalogować się, wylogować się i zmienić hasło. Po udanym logowaniu się obiekt

będzie zawierał listę wszystkich folderów, do których dana sesja ma dostęp. **Uwaga:** Dostęp do tego obiektu serwera należy uzyskiwać w środowisku jednowątkowym. Jedynym wyjątkiem jest anulowanie operacji serwera.

## Wskazówki dotyczące programowania

Pakiet `com.ibm.edms.od` należy zaimportować do aplikacji ODWEK.

Do uruchamiania aplikacji ODWEK, które korzystają z funkcji API języka Java nie jest potrzebny serwer HTTP ani serwer aplikacji WWW. Interpreter języka Java można uruchamiać w aplikacjach ODWEK.

Aby uruchomić interpreter języka Java w aplikacji ODWEK:

1. Skopiuj plik `arswww.ini` do zdefiniowanego przez użytkownika katalogu wykonawczego.
2. Skopiuj bibliotekę współużytkowaną do katalogu, w którym znajduje się skopiowany plik `arswww.ini`:

*Tabela 13. Nazwa pliku biblioteki współużytkowanej*

System operacyjny	Biblioteka współużytkowana
AIX	<code>libarswwwsl.a</code>
HP-UX	<code>libarswwwsl.sl</code>
Linux	<code>libarswwwsl.so</code>
Solaris	<code>libarswwwsl.so</code>
Windows	<code>arswwwsl.dll</code>

3. W przypadku systemów Windows skopiuj poniższe pliki do katalogu, do którego skopiowano plik `arswww.ini`:  
`ARSSCKNT.DLL`  
`ARSCT32.DLL`
4. Uruchamiając interpreter języka Java w aplikacji, podaj nazwę katalogu zdefiniowanego przez użytkownika. Przykład znajduje się w sekcji “Uruchamianie aplikacji ODWEK” na stronie 95.

---

## Konfigurowanie środowiska systemowego

Podczas konfigurowania środowiska systemów AIX, HP-UX, Linux, Solaris lub Windows, należy określić następujące ustawienia:

**pakiet** Należy zaimportować wszystkie aplikacje ODWEK.

- `com.ibm.edms.od`

**Pliki biblioteki**

**Obiekty współużytkowane dla systemów AIX, HP-UX, Linux i Solaris**

**Biblioteki DLL dla systemu Windows**

## Ustawianie zmiennych środowiskowych

Tworząc aplikację ODWEK, należy skonfigurować środowisko.

### AIX

W środowisku systemu AIX należy ustawić następujące zmienne środowiskowe, aby skonfigurować środowisko programistyczne do tworzenia aplikacji ODWEK.

**PATH** Upewnij się, że zmienna `PATH` zawiera ścieżkę `/usr/lpp/ars/www`



**LIBPATH** Upewnij się, że zmienna LIBPATH zawiera ścieżkę /usr/lpp/ars/www

**LD\_LIBRARY\_PATH**

Upewnij się, że zmienna LD\_LIBRARY\_PATH zawiera ścieżkę /usr/lpp/ars/www

**CLASSPATH** Upewnij się, że zmienna CLASSPATH zawiera ścieżkę do pliku /usr/lpp/ars/www/api/ODApi.jar, który jest biblioteką klas.

## HP-UX

W środowisku systemu HP-UX należy ustawić następujące zmienne środowiskowe, aby skonfigurować środowisko programistyczne do tworzenia aplikacji ODWEK.

**PATH** Upewnij się, że zmienna PATH zawiera ścieżkę /opt/ondemand/www

**LIBPATH** Upewnij się, że zmienna LIBPATH zawiera ścieżkę /opt/ondemand/www

**LD\_LIBRARY\_PATH**

Upewnij się, że zmienna LD\_LIBRARY\_PATH zawiera ścieżkę /opt/ondemand/www

**CLASSPATH** Upewnij się, że zmienna CLASSPATH zawiera ścieżkę do pliku /opt/ondemand/www/api/ODApi.jar, który jest biblioteką klas.

## Linux

W środowisku systemu Linux należy ustawić następujące zmienne środowiskowe, aby skonfigurować środowisko programistyczne do tworzenia aplikacji ODWEK.

**PATH** Upewnij się, że zmienna PATH zawiera ścieżkę /opt/ondemand/www

**LIBPATH** Upewnij się, że zmienna LIBPATH zawiera ścieżkę /opt/ondemand/www

**LD\_LIBRARY\_PATH**

Upewnij się, że zmienna LD\_LIBRARY\_PATH zawiera ścieżkę /opt/ondemand/www

**CLASSPATH** Upewnij się, że zmienna CLASSPATH zawiera ścieżkę do pliku /opt/ondemand/www/api/ODApi.jar, który jest biblioteką klas.

## Solaris

W środowisku systemu Solaris należy ustawić następujące zmienne środowiskowe, aby skonfigurować środowisko programistyczne do tworzenia aplikacji ODWEK.

**PATH** Upewnij się, że zmienna PATH zawiera ścieżkę /opt/ondemand/www

**LIBPATH** Upewnij się, że zmienna LIBPATH zawiera ścieżkę /opt/ondemand/www

**LD\_LIBRARY\_PATH**

Upewnij się, że zmienna LD\_LIBRARY\_PATH zawiera ścieżkę /opt/ondemand/www

**CLASSPATH** Upewnij się, że zmienna CLASSPATH zawiera ścieżkę do pliku /opt/ondemand/www/api/ODApi.jar, który jest biblioteką klas.

## Windows

W środowisku systemu Windows należy ustawić następujące zmienne środowiskowe, aby skonfigurować środowisko programistyczne do tworzenia aplikacji ODWEK.

**PATH** Upewnij się, że zmienna PATH zawiera x:\yyyyyyyy \DLL; gdzie x jest napędem, na którym zainstalowano ODWEK, a yyyyyyyy jest katalogiem instalacyjnym oprogramowania ODWEK.

**CLASSPATH** Upewnij się, że zmienna CLASSPATH zawiera x:\yyyyyyyy \WWW

\API\ODApi.jar gdzie x jest napędem, na którym zainstalowano ODWEK, a yyyyyyyyy jest katalogiem instalacyjnym oprogramowania biblioteki klas.

---

## Informacje diagnostyczne i związane ze śledzeniem

Do rozwiązywania problemów, jakie mogą się pojawić w aplikacjach korzystających z funkcji API języka Java, można użyć śledzenia i obsługi wyjątków.

### Śledzenie

Następujące parametry w pliku ARSWWW.INI powodują zapisanie informacji śledzenia do pliku arswwww.log w podanym katalogu:

```
[DEBUG]
LOG=1
LOGDIR=/ars/www/protokol
```

**Uwaga:** Ponieważ w pliku protokołu może znaleźć się znaczna ilość informacji, IBM zaleca włączanie protokołowania tylko w razie potrzeby, na przykład podczas próby odtworzenia sytuacji, w której wystąpił problem. Jeśli zachodzi konieczność włączenia protokołowania na dłuższy czas, należy upewnić się, że ścieżki do plików protokołu określają urządzenia pamięci masowej, na których jest dużo wolnego miejsca. Należy pamiętać o usuwaniu starych plików protokołu z systemu.

Dodatek J, "Narzędzia do określania problemu", na stronie 143 zawiera informacje dotyczące innych narzędzi, których można użyć do zbierania informacji o systemie i dokumentach.

### Obsługa wyjątków

Gdy funkcja API języka Java napotka wyjątek, to zgłasza go. Zgłoszenie wyjątku powoduje utworzenie obiektu wyjątku klasy ODEException lub jednej z jej podklas.

Podczas tworzenia klasy ODEException funkcja API protokołuje informacje diagnostyczne do pliku protokołu pod warunkiem, że włączone jest protokołowanie. W sekcji "Śledzenie" znajduje się więcej informacji dotyczących pliku protokołu używanego przez funkcje API języka Java.

Po przechwyceniu wyjątku ODEException możliwe jest wyświetlenie wszystkich komunikatów o błędach, kodów błędów i stanów błędów, które wystąpiły podczas pracy. Po przechwyceniu błędu wysyłany jest komunikat o błędzie określający miejsce, w którym przechwycono wyjątek. Dostępne są także identyfikatory błędu i wyjątku. Poniższy kod stanowi przykład procedury obsługi zgłoszenia i przechwycenia:

```
try
{
    odServer = new ODServer( );
    odServer.initialize( argv[9], "TcUpdate.java" );
    System.out.println( "Zalogowanie do " + argv[0] + "..." );
    odServer.logon( argv[0], argv[1], argv[2] );
    odServer.logoff( );
    odServer.terminate( );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( "          ID = " + e.getErrorId( ) );
    System.out.println( " komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}
```

---

## Stałe

Stałe dostępne dla funkcji API języka Java są opisane w pomocy elektronicznej. Więcej informacji na ten temat zawiera Dodatek C, "Funkcje API języka Java", na stronie 89.

---

## Uruchamianie aplikacji ODWEK

Do uruchamiania aplikacji ODWEK można używać interpretera języka Java. Tworząc, kompilując i uruchamiając aplikację ODWEK, należy uwzględnić następujące uwagi:

1. Utwórz aplikację ODWEK przy użyciu metod dostępnych poprzez funkcje API języka Java. Zaimportuj pakiet funkcji API języka Java do pliku aplikacji ODWEK. Na przykład:

```
//*****  
import java.util.*;  
import java.io.*;  
import com.ibm.edms.od.*;  
  
public class Logon  
{  
    public static void main ( String argv[] )  
    {  
        .  
        .  
    }  
}
```

2. Skompiluj plik aplikacji ODWEK ( .java ) za pomocą kompilatora javac, aby uzyskać plik .class. Instrukcje dotyczące kompilowania aplikacji w języku Java znajdują się w dokumentacji tego języka.

3. Uruchom aplikację (plik .class) za pomocą interpretera języka Java. Na przykład:

```
java Logon serwer ID_użytkownika hasło /tmp/ondemand/www
```

Gdzie Zalogowanie to nazwa pliku .class, serwer, ID\_użytkownika i hasło to parametry aplikacji, a /tmp/ondemand/www jest zdefiniowanym przez użytkownika katalogiem wykonawczym, w którym znajduje się kopia pliku arswwww.ini. **Uwaga:** W przykładzie tym przyjęto, że ścieżki do bibliotek klasy ODWEK i serwletu określono w zmiennych środowiskowych (patrz sekcja "Konfigurowanie środowiska systemowego" na stronie 92).

---

## Nawiązywanie połączenia z serwerem OnDemand

Obiekt klasy ODServer reprezentuje połączenie z serwerem OnDemand i zarządza nim, zapewniając obsługę transakcji i uruchamiając komendy serwera. Dodatek C, "Funkcje API języka Java", na stronie 89 zawiera informacje o tym, gdzie znaleźć spis metod i ich opisy w postaci elektronicznej.

Nawiązując połączenie z serwerem OnDemand, należy znać jego wymagania; na przykład hasło nie może mieć więcej niż osiem znaków.

### Nawiązywanie połączenia

Klasa ODServer zawiera metody używane do nawiązywania i rozłączania połączenia z serwerem OnDemand. W poniższym przykładzie nazwą serwera bibliotecznego OnDemand jest SERWBIB1, identyfikatorem użytkownika jest ADMIN, a hasłem HASŁO. Przykładowy program tworzy obiekt ODServer dla serwera OnDemand, nawiązuje połączenie, wykonuje operacje na serwerze (pominięte w przykładzie) i rozłącza połączenie.

```

odServer = new ODServer( );
odServer.initialize( "c:\odwekdir", "Przykład" );
System.out.println( "Zalogowanie do " + "SERWBIB1" + "..." );
odServer.logon( "SERWBIB1", "ADMIN", "HASŁO" );
.
.
.
odServer.logoff( );
odServer.terminate( );

```

W sekcji “Praca z serwerem OnDemand” znajduje się cała aplikacja, z której zaczerpnięty został powyższy przykład.

## Ustawianie i uzyskiwanie haseł

Za pomocą metod w klasie ODServer można ustawić hasło użytkownika na serwerze OnDemand lub uzyskać je z serwera. W poniższym przykładzie przedstawiono sposoby ustawiania i uzyskiwania hasła użytkownika na serwerze bibliotecznym OnDemand.

```

odServer = new ODServer( );
odServer.setServer( "SERWBIB1" );
odServer.setUserId( "ADMIN" );
odServer.setPassword( "HASŁO" );

System.out.println( "Zalogowanie do " + "SERWBIB1" + "..." );

odServer.logon( odServer.getServerName( ),
               odServer.getUserId( ),
               odServer.getPassword( ),
               ODConstant.CONNECT_TYPE_LOCAL,
               0 );

```

W sekcji “Praca z serwerem OnDemand” znajduje się cała aplikacja, z której zaczerpnięty został powyższy przykład.

---

## Praca z serwerem OnDemand

Obiekt klasy ODServer reprezentuje połączenie z serwerem OnDemand i zarządza nim, zapewniając obsługę transakcji i uruchamiając komendy serwera.

W poniższym przykładzie użyto metod ODServer w celu przygotowania do zalogowania się, określenia nazwy aplikacji, (opcjonalnego) wyświetlenia katalogu lokalnego, wyświetlenia nazwy serwera, identyfikatora użytkownika i hasła, wyświetlenia typu połączenia, wyświetlenia i ustawienia portu oraz odłączenia od serwera.

Poniższy przykład ilustruje użycie następujących metod ODServer:

- initialize
- logon
- logoff
- terminate
- getConnectType
- getLocalDir
- getPassword
- getPort
- getServerName
- getUserId
- setApplicationName
- setConnectType
- setLocalDir
- setPassword
- setPort

- setServer
- setUserId

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Katalog konfiguracyjny (położenie pliku arswwww.ini)
- (Opcjonalny) katalog serwera lokalnego

Przykład pracy z serwerem OnDemand:

```

/*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcServerMisc
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        String str;
        int j;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 4 )
        {
            System.out.println( "składnia: java TcServerMisc <server> <ID_użytkownika> <hasło> <katalog_konfiguracyjny>
                [<katalog_serwera_lokalnego>]" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Użyć następujących metod ODServer: setServer, setUserId i setPassword" );
            System.out.println( " w celu przygotowania do zalogowania się" );
            System.out.println( " Określić nazwę aplikacji" );
            System.out.println( " Wyświetlić:" );
            System.out.println( " Katalog lokalny" );
            System.out.println( " Nazwę serwera" );
            System.out.println( " ID użytkownika" );
            System.out.println( " Hasło" );
            System.out.println( " Typ połączenia" );
            System.out.println( " Ustawić i wyświetlić port" );
            System.out.println( " Ustawić typ połączenia" );
            System.out.println( " Wylogować się" );
            System.out.println( "" );
            System.out.println( "Sprawdź, czy wszystkie informacje są poprawne." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[3], "TcServerMisc.java" );
            odServer.setServer( argv[0] );
            odServer.setUserId( argv[1] );
            odServer.setPassword( argv[2] );

            System.out.println( "Zalogowanie do " + argv[0] + "..." );
            if ( argv.length == 4 )
                odServer.logon( );
            else
            {
                if ( argv.length == 5 )
                {
                    odServer.setLocalDir( argv[4] );
                    odServer.logon( odServer.getServerName( ),
                        odServer.getUserId( ),
                        odServer.getPassword( ),
                        ODConstant.CONNECT_TYPE_LOCAL,
                        0,
                        odServer.getLocalDir( ) );
                }
            }

            //-----
            // Przetestowanie różnych metod

```

```

//-----
System.out.println( "Ustawianie nazwy aplikacji TcServerMisc.java..." );
odServer.setApplicationName( "TcServerMisc.java" );

System.out.println( "Katalog lokalny: " + odServer.getLocalDir( ) );
System.out.println( "Nazwa serwera: " + odServer.getServerName( ) );
System.out.println( "ID użytkownika: " + odServer.getUserId( ) );
System.out.println( "Hasło: " + odServer.getPassword( ) );
System.out.println( "Typ połączenia: " + getConnectTypeName( odServer.getConnectType( ) ) );

j = odServer.getPort( );
System.out.println( "Ustawianie portu nr " + j + "..." );
odServer.setPort( j );
System.out.println( "Port: " + j );

if ( argv.length == 4 )
{
    System.out.println( "Ustawianie typu połączenia ODConstant.CONNECT_TYPE_TCPIP..." );
    odServer.setConnectType( ODConstant.CONNECT_TYPE_TCPIP );
}
else
{
    System.out.println( "Ustawianie typu połączenia ODConstant.CONNECT_TYPE_LOCAL..." );
    odServer.setConnectType( ODConstant.CONNECT_TYPE_LOCAL );
}

//-----
// Procedura czyszcząca
//-----
System.out.println( "Wylogowanie..." );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - w razie potrzeby przeanalizuj" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( "    ID = " + e.getErrorId( ) );
    System.out.println( "    komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}

}

static String getConnectTypeName( char type )
{
    String str;

    switch( type )
    {
        case ODConstant.CONNECT_TYPE_TCPIP:
            str = "TCPIP";
            break;
        case ODConstant.CONNECT_TYPE_LOCAL:
            str = "LOCAL";
            break;
        default:
            str = "*** Nieznany typ połączenia";
            break;
    }

    return str;
}
}

```

---

## Wyświetlanie grup aplikacji w folderze

Obiekt klasy ODFolder reprezentuje folder OnDemand.

W poniższym przykładzie zastosowano metody ODFolder w celu wyświetlenia liczby grup aplikacji, które można wyszukiwać w folderze, i nazwy każdej z grup.

Poniższy przykład ilustruje użycie następujących metod ODFolder:

- getNumApplGroups
- getApplGroups
- close

Zastosowano tu także metody `ODServer` w celu przygotowania do zalogowania, otwarcia podanego folderu i wylogowania się. Poniższy przykład ilustruje użycie następujących metod `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Katalog konfiguracyjny (położenie pliku `arswww.ini`)
- (Opcjonalny) katalog serwera lokalnego

Przykład tworzenia listy grup aplikacji w folderze:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcApplGrp
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        Object[] appl_grps;
        int j;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "składnia: java TcApplGrp <serwer> <ID_użytkownika> <hasło> <folder> <katalog_konfiguracyjny>
                [<katalog_serwera_lokalnego>]" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Wyświetlić jego nazwę" );
            System.out.println( " Wyświetlić liczbę grup aplikacji" );
            System.out.println( " Wyświetlić nazwę każdej z tych grup" );
            System.out.println( " " );
            System.out.println( "-----" );
            System.out.println( " " );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcListCriteria.java" );

            System.out.println( "Zalogowanie do " + argv[0] + "..." );
            if ( argv.length == 5 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 6 )
                    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

            //-----
            // Otwarcie podanego folderu
            //-----
            System.out.println( "Otwieranie folderu " + argv[3] + "..." );
            odFolder = odServer.openFolder( argv[3] );

            //-----
            // Wyświetlenie liczby grup aplikacji i ich nazw
            //-----
            System.out.println( "Liczba grup aplikacji w folderze: " + odFolder.getNumApplGroups( ) );
            appl_grps = odFolder.getApplGroups( );
            for ( j = 0; j < appl_grps.length; j++ )
                System.out.println( " " + appl_grps[j].toString( ) );

            //-----

```

```

// Procedura czyszcząca
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - w razie potrzeby przeanalizuj wyniki" );
System.out.println( "" );
}

catch ( ODEException e )
{
System.out.println( "Klasa ODEException: " + e );
System.out.println( " ID = " + e.getErrorId( ) );
System.out.println( " komunikat = " + e.getErrorMsg( ) );
e.printStackTrace( );
}

catch ( Exception e2 )
{
System.out.println( "wyjątek: " + e2 );
e2.printStackTrace( );
}
}
}

```

---

## Wyszukiwanie w folderze

Obiekt klasy `ODFolder` reprezentuje folder OnDemand. Obiekt klasy `ODCriteria` reprezentuje kryteria wyszukiwania w folderze OnDemand. Obiekt klasy `ODHit` reprezentuje dokument OnDemand.

W poniższym przykładzie użyto metod `ODFolder` w celu otwarcia podanego folderu, wyświetlenia jego nazwy, opisu, kolejności i kryteriów wyszukiwania, wykonania wyszukiwania oraz jego zamknięcia. Metod `ODCriteria` użyto w celu ustawienia bieżącego operandu i wartości wyszukiwania. Metody `ODHit` zastosowano w celu uzyskania wartości wyświetlania dokumentu, jego typu, trwałego identyfikatora, położenia oraz typu zawartości MIME.

Poniższy przykład ilustruje użycie następujących metod `ODFolder`:

- `getName`
- `getDescription`
- `getDisplayOrder`
- `getCriteria`
- `search`
- `getSearchMessage`
- `close`

Poniższy przykład ilustruje użycie następujących metod `ODCriteria`:

- `getName`
- `setOperand`
- `setSearchValue`
- `setSearchValues`

Poniższy przykład ilustruje użycie następujących metod `ODHit`:

- `getDisplayValue`
- `getDisplayValues`
- `getDocType`
- `getMimeType`
- `getDocLocation`
- `getDocId`

Zastosowano tu także metody `ODServer` w celu przygotowania do zalogowania, otwarcia podanego folderu i wylogowania się. Poniższy przykład ilustruje użycie następujących metod `ODServer`:

- `initialize`



- logon
- openFolder
- terminate

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Nazwa kryterium
- Operator (musi mieć jedną z wartości: eq, ne, lt, le, gt, ge, in, ni, li, nl, be, nb)
- Wartość wyszukiwania 1
- (opcjonalna) Wartość wyszukiwania 2
- Katalog konfiguracyjny (położenie pliku arswwww.ini)

**Uwaga:** Liczba trafień może być ograniczona wartością parametru MAXHITS w pliku arswwww.ini.

Przykład wyszukiwania w folderze:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcSearch
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        ODHit odHit;
        Enumeration values_enum;
        Vector hits;
        String[] display_crit;
        String header, line1, line2, hit_value, useable_value;
        boolean mismatch_detected;
        int j, k, opr;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy prace
        //-----
        if ( argv.length < 9 )
        {
            System.out.println( "składnia: java TcSearch <serwer> <ID_użytkownika> <hasło> <folder> <kryteria> <operator> <wartości1>
            <wartości2> <katalog_konfiguracyjny>" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Wyświetlić nazwę i opis folderu" );
            System.out.println( " Pobrać podane kryteria" );
            System.out.println( " Określić operatora" );
            System.out.println( " Określić operand(y)" );
            System.out.println( " Wyszukać w folderze" );
            System.out.println( " Wyświetlić komunikat wyszukiwania (jeśli został wysłany)" );
            System.out.println( " Wyświetlić liczbę trafień" );
            System.out.println( " Wyświetlić liczbę trafień, po 3 wiersze na każde trafienie:" );
            System.out.println( " 1. Trafienia zwrócone przez metodę ODHit.getDisplayValue" );
            System.out.println( " 2. Trafienia zwrócone przez metodę ODHit.getDisplayValues" );
            System.out.println( " 3. Typ dokumentu, typ mime, położenie dokumentu i wartości identyfikatora dokumentu" );
            System.out.println( "" );
            System.out.println( "Sprawdź, czy wiersze 1 i 2 listy trafień są takie same i czy wartości" );
            System.out.println( "listy trafień są takie same, jak wyświetlone przez klienta dla systemu Windows." );
            System.out.println( "Jeśli plik arswwww.ini ogranicza liczbę trafień, na liście może znajdować się" );
            System.out.println( "mniej trafień niż przy użyciu klienta dla systemu Windows." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[8], "TcSearch.java" );
            System.out.println( "Zalogowanie do " + argv[0] + "..." );
            odServer.logon( argv[0], argv[1], argv[2] );

            //-----
            // Otwarcie podanego folderu i znalezienie żądanych kryteriów
            //-----
            System.out.println( "Otwieranie folderu " + argv[3] + "..." );
            odFolder = odServer.openFolder( argv[3] );
            System.out.println( "Nazwa=" + odFolder.getName( ) + " Opis=" + odFolder.getDescription( ) + "" );
        }
    }
}
```

```

System.out.println( "Pobieranie " + argv[4] + " kryteriów..." );
odCrit = odFolder.getCriteria( argv[4] );

//-----
// Konwersja parametru operatora do wewnętrznej wartości i ustawienie
// operatora kryteriów
//-----
System.out.println( "Ustawienie operatora na " + argv[5] + "..." );
if ( argv[5].equals( "eq" ) )
    opr = ODConstant.OPEqual;
else if ( argv[5].equals( "ne" ) )
    opr = ODConstant.OPNotEqual;
else if ( argv[5].equals( "lt" ) )
    opr = ODConstant.OPLessThan;
else if ( argv[5].equals( "le" ) )
    opr = ODConstant.OPLessThanEqual;
else if ( argv[5].equals( "gt" ) )
    opr = ODConstant.OPGreaterThan;
else if ( argv[5].equals( "ge" ) )
    opr = ODConstant.OPGreaterThanEqual;
else if ( argv[5].equals( "in" ) )
    opr = ODConstant.OPIn;
else if ( argv[5].equals( "ni" ) )
    opr = ODConstant.OPNotIn;
else if ( argv[5].equals( "li" ) )
    opr = ODConstant.OPLike;
else if ( argv[5].equals( "nl" ) )
    opr = ODConstant.OPNotLike;
else if ( argv[5].equals( "be" ) )
    opr = ODConstant.OPBetween;
else if ( argv[5].equals( "nb" ) )
    opr = ODConstant.OPNotBetween;
else
    opr = -1;

System.out.println( "Ustawienie operandów..." );
odCrit.setOperand( opr );

if ( opr == ODConstant.OPBetween || opr == ODConstant.OPNotBetween )
{
    odCrit.setSearchValues( argv[6], argv[7] );
    System.out.println( " " + odCrit.getName() + " " + getOperatorName( opr ) + " " + argv[6] + " i " + argv[7] );
}
else
{
    odCrit.setSearchValue( argv[6] );
    System.out.println( " " + odCrit.getName() + " " + getOperatorName( opr ) + " " + argv[6] );
}

//-----
// Wyszukiwanie w folderze
//-----
System.out.println( " Wyszukiwanie " + argv[3] + "..." );
hits = odFolder.search( );
System.out.println( "Komunikat wyszukiwania: " + odFolder.getSearchMessage( ) );
System.out.println( " Liczba trafień: " + hits.size( ) );

//-----
// Wyświetlenie trafień
//-----
mismatch_detected = false;
if ( hits != null && hits.size( ) > 0 )
{
    display_crit = odFolder.getDisplayOrder( );
    header = " ";
    for( j = 0; j < display_crit.length; j++ )
        header = header + display_crit[j] + "--";
    System.out.println( " -----" );
    System.out.println( header + " (z metody ODHit.getDisplayValue)" );
    System.out.println( header + " (z metody ODHit.getDisplayValues)" );
    System.out.println( " DocType--MimeType--DocLocation--DocId" );
    System.out.println( " -----" );
    for ( j = 0; j < hits.size( ); j++ )
    {
        odHit = (ODHit)hits.elementAt( j );
        line1 = " ";
        for ( k = 0; k < display_crit.length; k++ )
        {
            hit_value = odHit.getDisplayValue( display_crit[k] );
            useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
            line1 = line1 + useable_value + "--";
        }
        System.out.println( line1 );
        line2 = " ";
        for ( values_enum = odHit.getDisplayValues( ); values_enum.hasMoreElements( ); )
        {
            hit_value = (String)values_enum.nextElement( );
            useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
            line2 = line2 + useable_value + "--";
        }
        System.out.println( line2 );
        System.out.println( " " + getDocTypeString( odHit.getDocType( ) ) +
            " --" + odHit.getMimeType( ) +
            " --" + getLocationString( odHit.getDocLocation( ) ) +
            " --" + odHit.getDocId( ) );
        if ( !line1.equals( line2 ) )
            mismatch_detected = true;
    }
}

//-----
// Procedura czyszcząca
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - w razie potrzeby przeanalizuj" );

```

```

System.out.println( "" );
if ( mismatch_detected )
{
    System.out.println( "*** Znaleziono co najmniej jedno niedopasowanie" );
    System.out.println( "*** pomiędzy wierszami 1 i 2 trafienia" );
    System.out.println( "" );
}
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( "          ID = " + e.getErrorId( ) );
    System.out.println( "          komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}

static String getOperatorName( int oper )
{
    String str;

    switch( oper )
    {
        case ODEConstant.OPEqual:
            str = "Jest równe";
            break;
        case ODEConstant.OPNotEqual:
            str = "Nie jest równe";
            break;
        case ODEConstant.OPLessThan:
            str = "Mniejsze niż";
            break;
        case ODEConstant.OPLessThanEqual:
            str = "Mniejsze lub równe";
            break;
        case ODEConstant.OPGreaterThan:
            str = "Większe od";
            break;
        case ODEConstant.OPGreaterThanEqual:
            str = "Większe lub równe";
            break;
        case ODEConstant.OPIn:
            str = "Zawiera się";
            break;
        case ODEConstant.OPNotIn:
            str = "Nie zawiera się";
            break;
        case ODEConstant.OPLike:
            str = "Podobne";
            break;
        case ODEConstant.OPNotLike:
            str = "Niepodobne";
            break;
        case ODEConstant.OPBetween:
            str = "Pomiędzy";
            break;
        case ODEConstant.OPNotBetween:
            str = "Nie pomiędzy";
            break;
        default:
            str = "Nieznany operator";
            break;
    }

    return str;
}

static String getDocTypeString( char type )
{
    String str;

    switch( type )
    {
        case ODEConstant.FileTypeAFP:
            str = "AFP";
            break;
        case ODEConstant.FileTypeBMP:
            str = "BMP";
            break;
        case ODEConstant.FileTypeEMAIL:
            str = "EMAIL";
            break;
        case ODEConstant.FileTypeGIF:
            str = "GIF";
            break;
        case ODEConstant.FileTypeJFIF:
            str = "JFIF";
            break;
        case ODEConstant.FileTypeLINE:
            str = "LINE";
            break;
        case ODEConstant.FileTypeMETA:
            str = "META";
            break;
        case ODEConstant.FileTypeNONE:
            str = "NONE";
            break;
        case ODEConstant.FileTypePCX:
            str = "PCX";
            break;
        case ODEConstant.FileTypePDF:
            str = "PDF";
            break;
    }
}

```

```

        case ODConstant.FileTypePNG:
            str = "PNG";
            break;
        case ODConstant.FileTypeTIFF:
            str = "TIFF";
            break;
        case ODConstant.FileTypeUSRDEF:
            str = "USRDEF";
            break;
        default:
            str = "*** Niepoprawny typ dokumentu ***";
            break;
    }

    return str;
}

static String getLocationString( int loc )
{
    String str;

    switch( loc )
    {
        case ODConstant.DocLocationCache:
            str = "Pamięć podręczna";
            break;
        case ODConstant.DocLocationArchive:
            str = "Archiwum";
            break;
        case ODConstant.DocLocationExternal:
            str = "Zewnętrzne";
            break;
        case ODConstant.DocLocationUnknown:
            str = "Nieznane";
            break;
        default:
            str = "*** Niepoprawne położenie dokumentu ***";
            break;
    }

    return str;
}
}

```

---

## Wyszukiwanie w folderze za pomocą łańcucha SQL

W poniższym przykładzie użyto metod ODFolder w celu otwarcia podanego folderu, uruchomienia wyszukiwania za pomocą podanego łańcucha SQL i zamknięcia folderu. Metody ODHit zastosowano w celu wyświetlenia liczby pozycji, które pasują do zapytania, oraz listy dokumentów.

Poniższy przykład ilustruje użycie następujących metod ODFolder:

- setAppGroupForSearchWithSQL
- search
- getDisplayOrder
- close

Poniższy przykład ilustruje użycie następujących metod ODHit:

- getDisplayValue

Zastosowano tu także metody ODServer w celu przygotowania do zalogowania, otwarcia podanego folderu i wylogowania się. Poniższy przykład ilustruje użycie następujących metod ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Nazwa grupy aplikacji
- Łańcuch SQL
- Katalog konfiguracyjny (położenie pliku arswwww.ini)

## Przykład wyszukiwania w folderze za pomocą łańcucha SQL:

```

//*****
//
// Przykład: TcSearchWithSQL
//
// Ten program testujący wykonuje następujące czynności:
// Zalogowanie do podanego serwera
// Otwarcie podanego folderu
// Wyszukiwanie w folderze za pomocą łańcucha SQL
// Wyświetlenie liczby trafień
// Wyświetlenie listy trafień
//
// Testuje on następujące metody:
// ODServer
//   initialize
//   logon
//   openFolder
//   logoff
//   terminate
// ODFolder
//   setAppGroupForSearchWithSQL
//   search
//   getDisplayOrder
//   close
// ODHit
//   getDisplayValue
//
// Parametry:
// 1. Nazwa serwera
// 2. ID użytkownika
// 3. Hasło
// 4. Nazwa folderu
// 5. Nazwa grupy aplikacji
// 6. łańcuch SQL
// 7. Katalog konfiguracyjny (zawiera plik arswww.ini)
//
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcSearchWithSQL
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Enumeration values_enum;
        Vector hits;
        String[] display_crit;
        String server, userid, password, folder, directory;
        String sql, appl_group;
        String header, line, hit_value, useable_value;
        int j, k;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 6 )
        {
            System.out.println( "składnia: java TcSearchWithSQL <serwer> <ID_użytkownika> <hasło> <folder> <grupa_aplikacji> <łańcuch_SQL>
                                <katalog_konfiguracyjny>" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Uruchomić wyszukiwanie w folderze za pomocą podanego łańcucha SQL" );
            System.out.println( " Wyświetlić liczbę trafień" );
            System.out.println( " Wyświetlić listę trafień" );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            server = argv[0];
            userid = argv[1];
            password = argv[2];
            folder = argv[3];
            appl_group = argv[4];
            sql = argv[5];
            directory = argv[6];

            odServer = new ODServer( );
            odServer.initialize( directory, "TcSearchWithSQL.java" );
            System.out.println( "Zalogowanie do serwera " + server + "..." );
            odServer.logon( server, userid, password );

            //-----
            // Otwarcie podanego folderu
            //-----
            System.out.println( "Otwieranie folderu " + folder + "..." );
            odFolder = odServer.openFolder( folder );

            //-----
            // Wyszukiwanie w folderze
            //-----
            if ( appl_group.length( ) > 0 )

```

```

    {
        System.out.println( "Ustawianie grupy aplikacji do wyszukiwania: " + appl_group );
        odFolder.setAppGroupForSearchWithSQL( appl_group );
    }

    //-----
    // Wyszukiwanie w folderze
    //-----
    System.out.println( " Wyszukiwanie w folderze " + folder + "...";
    hits = odFolder.search( sql );
    System.out.println( " Liczba trafień: " + hits.size( ) );

    //-----
    // Wyświetlenie trafień
    //-----
    if ( hits != null && hits.size( ) > 0 )
    {
        display_crit = odFolder.getDisplayOrder( );
        header = " ";
        for( j = 0; j < display_crit.length; j++ )
            header = header + display_crit[j] + "--";
        System.out.println( " -----" );
        System.out.println( header );
        System.out.println( " -----" );
        for ( j = 0; j < hits.size( ); j++ )
        {
            odHit = (ODHit)hits.elementAt( j );
            line = " ";
            for ( k = 0; k < display_crit.length; k++ )
            {
                hit_value = odHit.getDisplayValue( display_crit[k] );
                useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
                line = line + useable_value + "--";
            }
            System.out.println( line );
        }
    }

    //-----
    // Procedura czyszcząca
    //-----
    odFolder.close( );
    odServer.logoff( );
    odServer.terminate( );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );
    System.out.println( "Test zakończony - w razie potrzeby przeanalizuj" );
    System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( " ID = " + e.getErrorId( ) );
    System.out.println( " komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}
}

```

## Anulowanie wyszukiwania

W poniższym przykładzie użyto metody `ODServer.cancel` w celu anulowania trwającego wyszukiwania.

Metod `ODServer`, `ODFolder` i `ODCriteria` użyto w celu zalogowania się do serwera, otwarcia folderu i ustawienia kryterium daty na 1970-2001. Następnie zainicjowano drugi wątek w celu uruchomienia wyszukiwania. Po zakończeniu działania tego wątku wyświetlana jest lista trafień. Następnie ponownie zainicjowano drugi wątek, również w celu uruchomienia wyszukiwania. Proces jest zawieszany na 0,5 sekundy, a następnie wyszukiwanie jest anulowane. Po zakończeniu działania tego wątku wyświetlana jest lista trafień.

Poniższy przykład ilustruje użycie następujących metod `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

Poniższy przykład ilustruje użycie następujących metod `ODFolder`:

- `getCriteria`

- search
- close

Poniższy przykład ilustruje użycie następujących metod ODCriteria:

- setOperand
- setSearchValues

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Katalog konfiguracyjny (położenie pliku arswwww.ini)
- (Opcjonalny) katalog serwera lokalnego

Przykład anulowania wyszukiwania:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

class TestThread extends Thread
{
    ODFolder odFolder;

    TestThread( ODFolder fld )
    {
        odFolder = fld;
    }

    public void run( )
    {
        Vector hits;

        try
        {
            System.out.println( " Drugi wątek wyszukuje..." );
            hits = odFolder.search( );
            System.out.println( " Wyszukiwanie zakończone, liczba trafień: " + hits.size( ) );
        }

        catch ( ODEException e )
        {
            System.out.println( "Klasa ODEException: " + e );
            System.out.println( " ID = " + e.getErrorId( ) );
            System.out.println( " komunikat = " + e.getErrorMsg( ) );
            e.printStackTrace( );
        }

        catch ( Exception e2 )
        {
            System.out.println( "wyjątek: " + e2 );
            e2.printStackTrace( );
        }
    }
}

public class TcCancelSearch
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        TestThread search_thread;
        int j;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "składnia: java TcCancelSearch <serwer> <ID_użytkownika> <hasło> <folder> <katalog_konfiguracyjny>
                [<katalog_serwera_lokalnego>]" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Określić kryterium daty równe 1970-2001" );
            System.out.println( " Zainicjować drugi wątek, który przeprowadzi wyszukiwanie" );
            System.out.println( " Wyświetlić liczbę trafień po zakończeniu działania drugiego wątku" );
            System.out.println( " Zainicjować drugi wątek, który przeprowadzi wyszukiwanie" );
            System.out.println( " Zawiesić wykonywanie programu na 0,5 sekundy" );
        }
    }
}
```

```

System.out.println( " Anulować wyszukiwanie" );
System.out.println( " Wyświetlić liczbę trafień po zakończeniu działania drugiego wątku" );
System.out.println( "" );
System.out.println( "Sprawdź, czy wybrano folder zawierający kryterium o nazwie Data." );
System.out.println( "Sprawdź, czy folder zawiera wiele trafień i czy plik arswww.ini" );
System.out.println( "nie ogranicza zbytnio dopuszczalnej liczby trafień." );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );

//-----
// Zalogowanie do podanego serwera
//-----
odServer = new ODServer( );
odServer.initialize( argv[4], "TcCancelSearch.java" );

System.out.println( "Zalogowanie do " + argv[0] + "..." );
if ( argv.length == 5 )
    odServer.logon( argv[0], argv[1], argv[2] );
else
    if ( argv.length == 6 )
        odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

//-----
// Otwarcie podanego folderu i wyświetlenie jego nazwy i opisu
//-----
System.out.println( "Otwieranie folderu " + argv[3] + "..." );
odFolder = odServer.openFolder( argv[3] );
odCrit = odFolder.getCriteria( "Date" );
odCrit.setOperand( ODConstant.OPBetween );
odCrit.setSearchValues( "01/01/70", "01/01/01" );

//-----
// Uruchomienie wyszukiwania w oddzielnym wątku, krótkie zawieszenie,
// przywrócenie wykonywania programu i anulowanie wyszukiwania
//-----
System.out.println( "Główny wątek inicjuje wyszukiwanie (bez próby anulowania)..." );
search_thread = new TestThread( odFolder );
search_thread.start( );
search_thread.join( );

System.out.println( "Główny wątek inicjuje wyszukiwanie (z próbą anulowania)..." );
search_thread = new TestThread( odFolder );
search_thread.start( );
System.out.println( "Główny wątek zawieszony na 0,5 sekundy..." );
( Thread.currentThread( ) ).sleep( 500 );
System.out.println( "Główny wątek próbuje anulować wyszukiwanie..." );
odServer.cancel( );
System.out.println( "Główny wątek powróci po próbie anulowania..." );
search_thread.join( );

//-----
// Procedura czyszcząca
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - sprawdź, czy drugie wyszukiwanie," );
System.out.println( "które anulowano, przyniosło mniej trafień niż pierwsze." );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( "    ID = " + e.getErrorId( ) );
    System.out.println( "    komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}
}

```

---

## Wyświetlanie kryteriów wyszukiwania

Poniższy przykład ilustruje zastosowanie metod `ODCriteria` w celu wyświetlenia kryteriów wyszukiwania w danym folderze. Dla każdego pola wyszukiwania program ten wyświetla nazwę pola, domyślny operator, poprawne w danym polu operatory, typ pola i wszelkie domyślne wartości wyszukiwania. Wartości domyślne są wyświetlane przez metody `ODCriteria.getSearchValues` i `ODCriteria.getValues`. Stałe wartości wyszukiwania dla wszystkich zdefiniowanych pól wyszukiwania są zdefiniowane jako `FixedChoice` lub `Segment`.

Poniższy przykład ilustruje użycie następujących metod `ODCriteria`:



- setOperand
- getValidOperands
- getType
- getValues
- setSearchValues
- getFixedValues

Poniższy przykład ilustruje użycie następujących metod ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

Poniższy przykład ilustruje użycie następujących metod ODFolder:

- getCriteria
- close

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Katalog konfiguracyjny (położenie pliku arswwww.ini)
- (Opcjonalny) katalog serwera lokalnego

Przykład wyświetlania kryteriów wyszukiwania:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListCriteria
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        Enumeration crit_enum;
        Vector value_vec;
        String[] search_values, fixed_values;
        int[] valid_oprs;
        int j, opr;
        char ffield_type;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "składnia: java TcListCriteria <serwer> <ID_użytkownika> <hasło> <folder> <katalog_konfiguracyjny>
                                [<katalog_serwera_lokalnego>]" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Wyświetlić nazwę i opis folderu" );
            System.out.println( " Wyświetlić liczbę kryteriów dla folderu" );
            System.out.println( " Dla każdego kryterium wyświetlić" );
            System.out.println( "   Nazwę" );
            System.out.println( "   Domyślny operator" );
            System.out.println( "   Poprawne operatory" );
            System.out.println( "   Typ pola" );
            System.out.println( "   Wartości domyślne (za pomocą metody ODCrit.getSearchValues)" );
            System.out.println( "   Wartości domyślne (za pomocą metody ODCrit.getValues)" );
            System.out.println( "   Stałe wartości (tylko dla kryteriów FixedChoice i Segment)" );
            System.out.println( "" );
            System.out.println( "Upewnij się, że dla żadnego z operatorów nie podano 'Nieznany operator,'" );
            System.out.println( "dla żadnego z typów pól nie podano 'Nieznany typ', że wartości domyślne" );
            System.out.println( "są takie same dla każdej metody oraz że wszystkie wyświetlone informacje" );
            System.out.println( "są takie same, jak w przy użyciu klienta dla systemu Windows." );
            System.out.println( "" );
            System.out.println( "-----" );
        }
    }
}
```

```

System.out.println( "" );

//-----
// Zalogowanie do podanego serwera
//-----
odServer = new ODServer( );
odServer.initialize( argv[4], "TcListCriteria.java" );

System.out.println( "Zalogowanie do " + argv[0] + "..." );
if ( argv.length == 5 )
    odServer.logon( argv[0], argv[1], argv[2] );
else
    if ( argv.length == 6 )
        odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

//-----
// Otwarcie podanego folderu i wyświetlenie jego nazwy i opisu
//-----
System.out.println( "Otwieranie folderu " + argv[3] + "..." );
odFolder = odServer.openFolder( argv[3] );
System.out.println( "Nazwa=" + odFolder.getName( ) + " Opis=" + odFolder.getDescription( ) + "" );
System.out.println( "Liczba kryteriów wynosi " + odFolder.getNumCriteria( ) + ", oto one:" );

//-----
// Dla każdego kryterium dla folderu:
//-----
for ( crit_enum = odFolder.getCriteria( ); crit_enum.hasMoreElements( ); )
{
    //-----
    // Wyświetlenie nazwy kryterium
    //-----
    System.out.println( "" );
    odCrit = (ODCriteria)crit_enum.nextElement( );
    System.out.println( odCrit.getName( ) );

    //-----
    // Wyświetlenie domyślnego operatora
    //-----
    opr = odCrit.getOperand( );
    System.out.println( " Domyślny operator" );
    System.out.println( " " + getOperatorName( opr ) );

    //-----
    // Wyświetlenie poprawnych operatorów
    //-----
    valid_oprs = odCrit.getValidOperands( );
    System.out.println( " Poprawne operatory:" );
    for ( j = 0; j < valid_oprs.length; j++ )
        System.out.println( " " + getOperatorName( valid_oprs[j] ) );

    //-----
    // Wyświetlenie typu pola
    //-----
    field_type = odCrit.getType( );
    System.out.println( " Typ:" );
    System.out.println( " " + getTypeName( field_type ) );

    //-----
    // Wyświetlenie wartości domyślnych za pomocą metody ODCrit.getValues( )
    //-----
    value_vec = odCrit.getValues( );
    System.out.println( " Wartości domyślne (metoda ODCrit.getValues):" );
    System.out.println( " " + value_vec.elementAt( 0 ) + "" );
    System.out.println( " " + value_vec.elementAt( 1 ) + "" );

    //-----
    // Wyświetlenie wartości domyślnych za pomocą metody ODCrit.getSearchValues( )
    //-----
    search_values = odCrit.getSearchValues( );
    System.out.println( " Wartości domyślne (metoda ODCrit.getSearchValues):" );
    for ( j = 0; j < search_values.length; j++ )
        System.out.println( " " + search_values[j] + "" );

    //-----
    // Wyświetlenie stałych wyborów
    //-----
    switch ( field_type )
    {
        case ODConstant.InputTypeChoice:
        case ODConstant.InputTypeSegment:
            fixed_values = odCrit.getFixedValues( );
            System.out.println( " Stałe wartości (tylko dla pól typu FixedChoice i Segment):" );
            for ( j = 0; j < fixed_values.length; j++ )
                System.out.println( " " + fixed_values[j] + "" );
            break;
    }
}

//-----
// Procedura czyszcząca
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - w razie potrzeby przeanalizuj i porównaj wyniki" );
System.out.println( " z wynikami pracy klienta dla systemu Windows." );
System.out.println( "" );
}

catch ( ODException e )

```

```

    {
        System.out.println( "Klasa ODException: " + e );
        System.out.println( "          ID = " + e.getErrorId( ) );
        System.out.println( "    komunikat = " + e.getErrorMsg( ) );
        e.printStackTrace( );
    }

    catch ( Exception e2 )
    {
        System.out.println( "wyjątek: " + e2 );
        e2.printStackTrace( );
    }
}

static String getOperatorName( int oper )
{
    String str;

    switch( oper )
    {
        case ODConstant.OPEqual:
            str = "Jest równe";
            break;
        case ODConstant.OPNotEqual:
            str = "Nie jest równe";
            break;
        case ODConstant.OPLessThan:
            str = "Mniejsze niż";
            break;
        case ODConstant.OPLessThanEqual:
            str = "Mniejsze lub równe";
            break;
        case ODConstant.OPGreaterThan:
            str = "Większe od";
            break;
        case ODConstant.OPGreaterThanEqual:
            str = "Większe lub równe";
            break;
        case ODConstant.OPIn:
            str = "Zawiera się";
            break;
        case ODConstant.OPNotIn:
            str = "Nie zawiera się";
            break;
        case ODConstant.OPLike:
            str = "Podobne";
            break;
        case ODConstant.OPNotLike:
            str = "Niepodobne";
            break;
        case ODConstant.OPBetween:
            str = "Pomiędzy";
            break;
        case ODConstant.OPNotBetween:
            str = "Nie pomiędzy";
            break;
        default:
            str = "*** Nieznany operator";
            break;
    }

    return str;
}

static String getTypeName( char type )
{
    String str;

    switch( type )
    {
        case ODConstant.InputTypeNormal:
            str = "Normalne";
            break;
        case ODConstant.InputTypeTextSearch:
            str = "Wyszukiwanie tekstów";
            break;
        case ODConstant.InputTypeNoteTextSearch:
            str = "Wyszukiwanie tekstów uwag";
            break;
        case ODConstant.InputTypeNoteColor:
            str = "Kolor uwagi";
            break;
        case ODConstant.InputTypeChoice:
            str = "Stały wybór";
            break;
        case ODConstant.InputTypeSegment:
            str = "Segment";
            break;
        default:
            str = "*** Nieznany typ";
            break;
    }

    return str;
}
}

```

---

## Wyświetlanie folderów i informacji o nich

W poniższym przykładzie zastosowano metody ODServer w celu wydrukowania wiersza zawierającego liczbę folderów w podanym serwerze, które można wyszukiwać przy użyciu podanego identyfikatora użytkownika. Dla każdego folderu drukowany jest jeden wiersz z nazwą i opisem folderu.

Poniższy przykład ilustruje użycie następujących metod ODServer:

- initialize
- logon
- getNumFolders
- getFolderNames
- getFolderDescription
- logoff
- terminate

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Katalog konfiguracyjny (położenie pliku arswwww.ini)
- (Opcjonalny) katalog serwera lokalnego

Przykład wyświetlenia listy folderów i informacji o nich:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListFolders
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        Enumeration folders_enum;
        String folder_name, folder_desc;
        int num_folders;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 4 )
        {
            System.out.println( "składnia: java TcListFolders <serwer> <ID_użytkownika> <hasło> <katalog_konfiguracyjny>
                [<katalog_serwera_lokalnego>]" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Wyświetlić wiersz zawierający liczbę folderów w serwerze dostępnych dla danego
                użytkownika" );
            System.out.println( " Wyświetlić dla każdego folderu jeden wiersz zawierający nazwę i opis folderu" );
            System.out.println( "" );
            System.out.println( "Wyświetlone informacje powinny być takie same, jak w przy użyciu klienta dla systemu
                Windows" );
            System.out.println( "(z zaznaczonym przyciskiem 'Wszystkie', jeśli jest dostępny), ale kolejność folderów" );
            System.out.println( "może być inna w zależności od podanego serwera." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[3], "TcListFolders.java" );

            System.out.println( "Zalogowanie do " + argv[0] + "..." );
            if ( argv.length == 4 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 5 )
                    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[4] );
        }
    }
}
```

```

//-----
// Wyświetlenie liczby dostępnych folderów
//-----
num_folders = odServer.getNumFolders( );
System.out.println( "" );
System.out.println( "Liczba folderów dostępnych dla użytkownika " + argv[1] + " na serwerze " + argv[0]
+ ":" + num_folders );

//-----
// Wyświetlenie nazw i opisów folderów
//-----
for ( folders_enum = odServer.getFolderNames( ); folders_enum.hasMoreElements( ); )
{
    folder_name = (String)folders_enum.nextElement( );
    folder_desc = odServer.getFolderDescription( folder_name );
    System.out.println( " " + folder_name + " --- " + folder_desc );
}

//-----
// Procedura czyszcząca
//-----
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - w razie potrzeby porównaj wyniki z klientem dla systemu Windows" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( " ID = " + e.getErrorId( ) );
    System.out.println( " komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}
}

```

---

## Wyświetlanie listy dokumentów

W poniższym przykładzie użyto metod `ODFolder` i `ODHit` w celu wyszukiwania w folderze w oparciu o domyślne kryteria wyszukiwania, wydrukowania liczby dokumentów pasujących do zapytania oraz wyświetlenia tych dokumentów.

Poniższy przykład ilustruje użycie następujących metod `ODFolder`:

- `getName`
- `getDisplayOrder`
- `search`
- `close`

Poniższy przykład ilustruje użycie następujących metod `ODHit`:

- `getDisplayValue`

Przykład ten ilustruje także użycie następujących metod `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Katalog konfiguracyjny (położenie pliku `arswww.ini`)

## Przykład wyświetlania listy dokumentów:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcSortedHitlist
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Vector hits;
        String[] display_crit;
        String server, userid, password, folder, value;
        int j, k;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "składnia: java TcSortedHitlist <server> <ID_użytkownika> <hasło> <folder>
                <katalog_konfiguracyjny> );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Uruchomić wyszukiwanie w folderze za pomocą kryteriów domyślnych" );
            System.out.println( " Wyświetlić komunikat wyszukiwania (jeśli został wysłany)" );
            System.out.println( " Wyświetlić liczbę trafień" );
            System.out.println( " Wyświetlić listę trafień" );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Zalogowanie do serwera
            //-----
            server = argv[0];
            userid = argv[1];
            password = argv[2];
            folder = argv[3];
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcSortedHitlist.java" );
            System.out.println( "Zalogowanie do serwera " + server + " jako " + userid + "/" + password + "..." );
            odServer.logon( server, userid, password );

            //-----
            // Otwarcie folderu i uruchomienie wyszukiwania
            //-----
            System.out.println( "Otwieranie folderu " + folder + "..." );
            odFolder = odServer.openFolder( folder );
            System.out.println( "Wyszukiwanie w folderze za pomocą kryteriów domyślnych..." );
            hits = odFolder.search( );
            System.out.println( " Liczba trafień: " + hits.size( ) );

            //-----
            // Wyświetlenie trafień
            //-----
            if ( hits != null && hits.size( ) > 0 )
            {
                display_crit = odFolder.getDisplayOrder( );
                value = " ";
                for( j = 0; j < display_crit.length; j++ )
                    value = value + display_crit[j] + " ";
                System.out.println( value );
                for ( j = 0; j < hits.size( ); j++ )
                {
                    odHit = (ODHit)hits.elementAt( j );
                    value = " ";
                    for ( k = 0; k < display_crit.length; k++ )
                        value = value + odHit.getDisplayValue( display_crit[k] ) + " ";
                    System.out.println( value );
                }
            }

            //-----
        }
    }
}
```

```

// Procedura czyszcząca
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - sprawdź, czy kolejność trafień jest" );
System.out.println( " taka sama, jak wyświetlona przez klienta systemu Windows" );
System.out.println( "" );
}

catch ( ODEException e )
{
System.out.println( "Klasa ODEException: " + e );
System.out.println( " ID = " + e.getErrorId( ) );
System.out.println( " komunikat = " + e.getErrorMsg( ) );
e.printStackTrace( );
}

catch ( Exception e2 )
{
System.out.println( "wyjątek: " + e2 );
e2.printStackTrace( );
}
}
}

```

---

## Wczytywanie dokumentu

W poniższym przykładzie przedstawiono trzy różne metody wczytywania dokumentu:

- ODServer
- ODFolder
- ODHit

Poniższy przykład ilustruje zalogowanie się do podanego serwera, otwarcie określonego folderu, uruchomienie wyszukiwania w nim za pomocą kryteriów domyślnych, wyświetlenie trafień, wczytanie danych pierwszego trafienia za pomocą funkcji `ODHit.retrieve`, `ODServer.retrieve` oraz `ODFolder.retrieve`. W przykładzie tym wyświetlana jest długość danych wczytanych za pomocą każdej z metod, następnie program porównuje długość i zawartość danych wczytanych za pomocą poszczególnych metod i wyświetla wyniki porównania.

Poniższy przykład ilustruje użycie następujących metod `ODServer`:

- initialize
- logon
- openFolder
- retrieve
- logoff
- terminate

Poniższy przykład ilustruje użycie następujących metod `ODFolder`:

- search
- retrieve
- close

Poniższy przykład ilustruje użycie następujących metod `ODHit`:

- getDocId
- retrieve

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Katalog konfiguracyjny (położenie pliku `arswww.ini`)

- (Opcjonalny) katalog serwera lokalnego

#### Przykład wczytywania dokumentu:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcRetrieve
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        TcCallback callback;
        Vector hits;
        Vector hit_to_retrieve;
        byte[] data_from_hit;
        byte[] data_from_server;
        byte[] data_from_folder;
        int j;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "składnia: java TcRetrieve <serwer> <ID_użytkownika> <hasło> <folder> <katalog_konfiguracyjny>
                [<katalog_serwera_lokalnego>]" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Uruchomić wyszukiwanie w folderze za pomocą kryteriów domyślnych" );
            System.out.println( " Wyświetlić liczbę trafień" );
            System.out.println( " Wczytać dane pierwszego trafienia za pomocą metody ODHit.retrieve" );
            System.out.println( " Wczytać dane pierwszego trafienia za pomocą metody ODServer.retrieve" );
            System.out.println( " Wczytać dane pierwszego trafienia za pomocą metody ODFolder.retrieve" );
            System.out.println( " Wyświetlić długość danych wczytanych za pomocą każdej z metod" );
            System.out.println( " Porównać długość i zawartość danych wczytanych za pomocą każdej z metod" );
            System.out.println( " Wyświetlić wyniki porównania" );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcRetrieve.java" );
            System.out.println( "Zalogowanie do " + argv[0] + "..." );
            if ( argv.length == 5 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                odServer.logon( argv[0], argv[1], argv[2], ODServer.CONNECT_TYPE_LOCAL, 0, argv[5] );

            //-----
            // Otwarcie podanego folderu i wyszukiwanie za pomocą kryteriów domyślnych
            //-----
            System.out.println( "Otwieranie folderu " + argv[3] + "..." );
            odFolder = odServer.openFolder( argv[3] );
            System.out.println( "Wyszukiwanie za pomocą kryteriów domyślnych..." );
            hits = odFolder.search( );
            System.out.println( "Liczba trafień: " + hits.size( ) );

            //-----
            // Procedury wczytujące i porównujące
            //-----
            if ( hits.size( ) > 0 )
            {
                odHit = (ODHit)hits.elementAt( 0 );
                System.out.println( "Wczytywanie danych pierwszego trafienia za pomocą metody ODHit.retrieve..." );
                data_from_hit = odHit.retrieve( "" );
                System.out.println( "Wczytywanie danych pierwszego trafienia za pomocą metody ODServer.retrieve..." );
                data_from_server = odServer.retrieve( odHit.getDocId( ), argv[3], "" );
                hit_to_retrieve = new Vector( );
                hit_to_retrieve.addElement( odHit );
                System.out.println( "Wczytywanie danych pierwszego trafienia za pomocą metody ODFolder.retrieve
                    (metoda wywołania zwrótnego)..." );
                callback = new TcCallback( );
                odFolder.retrieve( hit_to_retrieve, callback );
                data_from_folder = callback.getData( );
                System.out.println( "Długość danych z:" );
                System.out.println( " metody ODHit.retrieve=" + data_from_hit.length );
                System.out.println( " metody ODServer.retrieve=" + data_from_server.length );
                System.out.println( " metody ODFolder.retrieve=" + data_from_folder.length );
                if ( data_from_hit.length == data_from_server.length )
                {
                    for ( j = 0; j < data_from_hit.length; j++ )

```



```

        {
            if ( data_from_hit[j] != data_from_server[j] )
                break;
        }
        if ( j == data_from_hit.length )
        {
            System.out.println( "ODHit i ODServer: porównanie długości i zawartości" );
            if ( data_from_hit.length == data_from_folder.length )
            {
                for ( j = 0; j < data_from_folder.length; j++ )
                {
                    if ( data_from_hit[j] != data_from_folder[j] )
                        break;
                }
                if ( j == data_from_folder.length )
                    System.out.println( "ODHit i ODFolder: porównanie długości i zawartości" );
                else
                {
                    System.out.println( "*** ODHit i ODFolder: niezgodność danych w pozycji " + j );
                    System.out.println( " Dane wg metody ODHit to: " + data_from_hit[j] );
                    System.out.println( " Dane wg metody ODFolder to: " + data_from_folder[j] );
                }
            }
            else
                System.out.println( "*** ODHit i ODFolder: niezgodność długości" );
        }
        else
        {
            System.out.println( "*** ODHit i ODServer: niezgodność danych w pozycji " + j );
            System.out.println( " Dane wg metody ODHit to: " + data_from_hit[j] );
            System.out.println( " Dane wg metody ODServer to: " + data_from_server[j] );
        }
    }
    else
        System.out.println( "*** ODHit i ODServer: niezgodność długości" );
}
else
    System.out.println( "Brak dokumentu do wczytania" );

//-----
// Procedura czyszcząca
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - przeanalizuj wyniki porównania" );
System.out.println( "" );
System.out.println( "Jeśli określonym w pliku arswm.ini typem danych jest 'native' (rodzimy), wszystkie " );
System.out.println( "dane i ich długość powinny się zgadzać; jeśli nie, należy spodziewać się różnic." );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( " ID = " + e.getErrorId( ) );
    System.out.println( " komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}
}
}

```

W poniższym przykładzie użyto metod ODCallback w celu masowego wczytania danych dokumentu.

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcCallback extends ODCallback
{
    byte[] data_from_folder;
    boolean init = true;

    TcCallback( )
    {
    }

    public void HitHandleCallback( int hit, int off, int len )
    {
    }

    public boolean HitCallback( String docid, char type, String[] values )

```

```

        throws Exception
    {
        return true;
    }

    public boolean DataCallback( byte[] data )
    {
        byte[] temp;
        int j, k;

        //-----
        // Jeśli odebrano pierwszy blok danych, inicjuje kontener; w przeciwnym razie
        // dopisuje nowe dane do odebranych poprzednio.
        //-----
        if ( init )
        {
            data_from_folder = data;
            init = false;
        }
        else
        {
            temp = new byte[ data_from_folder.length + data.length ];
            for ( j = 0; j < data_from_folder.length; j++ )
                temp[j] = data_from_folder[j];
            k = data_from_folder.length;
            for ( j = 0; j < data.length; j++ )
                temp[k++] = data[j];
            data_from_folder = temp;
        }

        return true;
    }

    public byte[] getData( )
    {
        return data_from_folder;
    }
}

```

---

## Drukowanie dokumentu

W poniższym przykładzie użyto metod `ODServer` i `ODFolder` w celu wyświetlenia drukarek dostępnych dla serwera oraz wydrukowania dokumentu na podanej drukarce. Zastosowano tu także metody `ODServer` w celu przygotowania do zalogowania, otwarcia podanego folderu i wylogowania się.

Poniższy przykład ilustruje użycie następujących metod `ODServer`:

- initialize
- logon
- openFolder
- getServerPrinters
- logoff
- terminate

Poniższy przykład ilustruje użycie następujących metod `ODFolder`:

- search
- printDocs
- close

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Nazwa drukarki
- Katalog konfiguracyjny (położenie pliku `arswww.ini`)
- (Opcjonalny) katalog serwera lokalnego

## Przykład drukowania dokumentu:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcPrintHit
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Vector hits, hit_to_print;
        String [] printers;
        String printer_name;
        boolean match;
        int j;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 6 )
        {
            System.out.println( "składnia: java TcPrintHit <serwer> <ID_uzytkownika> <haslo> <folder> <katalog_konfiguracyjny>
                <drukarka> [<katalog_serwera_lokalnego>]" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Wyświetlić listę drukarek dostępnych dla serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Uruchomić wyszukiwanie w folderze za pomocą kryteriów domyślnych" );
            System.out.println( " Wyświetlić liczbę trafień" );
            System.out.println( " Wydrukować pierwsze trafienie na podanej drukarce serwera" );
            System.out.println( " " );
            System.out.println( "-----" );
            System.out.println( " " );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            odServer = new ODServer ( );
            odServer.initialize( argv[5], "TcPrintHit.java" );
            System.out.println( "Zalogowanie do " + argv[0] + "..." );
            if ( argv.length == 6 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

            //-----
            // Sprawdzenie, czy serwer ma dostęp do jakichś drukarek
            //-----
            System.out.println( "Mczytywanie listy drukarek serwera..." );
            printer_name = argv[4];
            printers = odServer.getServerPrinters( );
            if ( printers.length > 0 )
            {
                //-----
                // Wyświetlenie listy dostępnych drukarek
                //-----
                System.out.println( "Liczba drukarek dostępnych dla serwera: " + printers.length + "; oto one:" );
                match = false;
                for( j = 0; j < printers.length; j++ )
                {
                    System.out.println( " " + printers[j] );
                    if ( printers[j].equals( printer_name ) )
                        match = true;
                }

                if ( match )
                {
                    //-----
                    // Otwarcie podanego folderu i wyszukiwanie za pomocą kryteriów domyślnych
                    //-----
                    System.out.println( "Otwieranie folderu " + argv[3] + "..." );
                    odFolder = odServer.openFolder( argv[3] );
                    System.out.println( "Wyszukiwanie za pomocą kryteriów domyślnych..." );
                    hits = odFolder.search( );
                    System.out.println( " Liczba trafień: " + hits.size( ) );

                    //-----
                    // Wydrukowanie pierwszego trafienia na podanej drukarce serwera
                    //-----
                    if ( hits.size( ) > 0 )
                    {
                        hit_to_print = new Vector( );
                        odHit = (ODHit)hits.elementAt( 0 );
                        hit_to_print.addElement( odHit );
                        System.out.println( "Drukowanie pierwszego trafienia na drukarce " + printer_name + "..." );
                        odFolder.printDocs( hit_to_print, printer_name );
                    }
                    else
                        System.out.println( "Brak dokumentu do wydrukowania" );

                    odFolder.close( );
                }
            }
            else
                System.out.println( "Podana drukarka ( " + printer_name + " ) nie jest dostępna dla tego serwera" );
        }
    }
}
```

```

else
    System.out.println( "Brak drukarek dostępnych dla serwera" );

//-----
// Procedura czyszcząca
//-----
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - przeanalizuj wyniki" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( "    ID = " + e.getErrorId( ) );
    System.out.println( "    komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}
}

```

---

## Wyświetlanie informacji o uwagach

W poniższym przykładzie użyto metod ODNNote w celu wyświetlenia szczegółowych informacji dotyczących uwagi. Program ten wykonuje następujące czynności: loguje się do określonego serwera, otwiera podany folder przy użyciu kryteriów domyślnych, a następnie wyświetla liczbę trafień, liczbę uwag powiązanych z pierwszym dokumentem oraz szczegółowe informacje dotyczące każdej uwagi dołączonej do dokumentu. Wyświetlone są: pozycja uwagi na stronie dokumentu, kolor tła, data i godzina dołączenia uwagi do dokumentu, identyfikator użytkownika, który utworzył uwagę oraz inne atrybuty.

Poniższy przykład ilustruje użycie następujących metod ODNNote:

- getColor
- getDateTime
- getGroupName
- getOffsetX
- getOffsetY
- getPageNum
- getText
- getUserid
- isOkToCopy
- isPublic

Przykład ten ilustruje także użycie następujących metod ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

Przykład ten ilustruje także użycie następujących metod ODFolder:

- search
- close

Przykład ten ilustruje także użycie następujących metod ODHit:

- getNotes

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło

- Nazwa folderu
- Katalog konfiguracyjny (położenie pliku arswwww.ini)
- (Opcjonalny) katalog serwera lokalnego

Przykład wyświetlania informacji o uwagach:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListNotes
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        ODNote odNote;
        Vector hits, notes;
        int j;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "składnia: java TcListNotes <serwer> <ID_użytkownika> <hasło> <folder> <katalog_konfiguracyjny>
                [katalog_serwera_lokalnego]" );
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Uruchomić wyszukiwanie w folderze za pomocą kryteriów domyślnych" );
            System.out.println( " Wyświetlić liczbę trafień" );
            System.out.println( " Wyświetlić liczbę uwag powiązanych z pierwszym trafieniem" );
            System.out.println( " Wyświetlić informacje dla każdej uwagi" );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcListNotes.java" );
            System.out.println( "Zalogowanie do " + argv[0] + "..." );
            if ( argv.length == 5 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

            //-----
            // Otwarcie podanego folderu i wyszukiwanie za pomocą kryteriów domyślnych
            //-----
            System.out.println( "Otwieranie folderu " + argv[3] + "..." );
            odFolder = odServer.openFolder( argv[3] );
            System.out.println( "Wyszukiwanie za pomocą kryteriów domyślnych..." );
            hits = odFolder.search( );
            System.out.println( " Liczba trafień: " + hits.size( ) );

            //-----
            // Wyświetlenie informacji dotyczących wszystkich uwag dla pierwszego trafienia
            //-----
            if ( hits.size( ) > 0 )
            {
                odHit = (ODHit)hits.elementAt( 0 );
                notes = odHit.getNotes( );
                System.out.println( " Liczba uwag dla pierwszego trafienia " + notes.size( ) );
                for ( j = 0; j < notes.size( ); j++ )
                {
                    odNote = (ODNote)notes.elementAt( j );
                    System.out.println( " " + (j+1) + " . Tekst=" + odNote.getText( ) + "" );
                    System.out.println( " ID użytkownika=" + odNote.getUserId( ) );
                    System.out.println( " Strona=" + odNote.getPageNum( ) );
                    System.out.println( " Kolor=" + odNote.getColor( ) );
                    System.out.println( " Data=" + odNote.getDateTime( ) );
                    System.out.println( " Grupa=" + odNote.getGroupName( ) );
                    System.out.println( " Pozycja=( " + odNote.getOffsetX( ) + " , " + odNote.getOffsetY( ) + " )" );
                    System.out.println( " Czy kopiować=" + odNote.isOkToCopy( ) );
                    System.out.println( " Publiczna=" + odNote.isPublic( ) );
                }
            }
            else
                System.out.println( "Brak dokumentu - nie można wyświetlić uwag" );

            //-----
            // Procedura czyszcząca
            //-----
            odFolder.close( );
        }
    }
}

```

```

        odServer.logoff( );
        odServer.terminate( );
        System.out.println( "" );
        System.out.println( "-----" );
        System.out.println( "" );
        System.out.println( "Test zakończony - sprawdź, czy informacje są takie same," );
        System.out.println( " taka sama, jak wyświetlona przez klienta systemu Windows" );
        System.out.println( "" );
    }

    catch ( ODEException e )
    {
        System.out.println( "Klasa ODEException: " + e );
        System.out.println( " ID = " + e.getErrorId( ) );
        System.out.println( " komunikat = " + e.getErrorMsg( ) );
        e.printStackTrace( );
    }

    catch ( Exception e2 )
    {
        System.out.println( "wyjątek: " + e2 );
        e2.printStackTrace( );
    }
}

```

---

## Dodawanie uwagi

Obiekt klasy ODHit reprezentuje dokument OnDemand. W poniższym przykładzie użyto metod ODHit w celu wyświetlenia liczby uwag powiązanych z dokumentem oraz dodania nowej uwagi o następujących atrybutach:

- Określony tekst uwagi
- OkToCopy=false
- Public=false (czyli uwaga prywatna)
- Pusta nazwa grupy

Poniższy przykład ilustruje użycie następujących metod ODHit:

- getNotes
- addNote

W przykładzie tym użyto również metod ODServer w celu przygotowania logowania się, otwarcia podanego folderu i wylogowania się, a także metod ODFolder w celu wyszukiwania w folderze, uzyskiwania liczby trafień odpowiadających zapytaniu oraz zamykania folderu.

Poniższy przykład ilustruje użycie następujących metod ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

Poniższy przykład ilustruje użycie następujących metod ODFolder:

- search
- getHits
- close

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Tekst uwagi
- Katalog konfiguracyjny (położenie pliku arswwww.ini)
- (Opcjonalny) katalog serwera lokalnego

Przykład dodawania adnotacji:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

```

```

public class TcAddNote
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        ODNote odNote;
        Vector hits, notes;
        int j;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 6 )
            System.out.println( "składnia: java TcAddNote <server> <ID_użytkownika> <hasło> <folder> <katalog_konfiguracyjny>
            <tekst_uwagi> [<katalog_serwera_lokalnego>]" );

        {
            return;
        }

        try
        {
            //-----
            // Przygotowanie
            //-----
            System.out.println( "Ten test powinien:" );
            System.out.println( " Zalogować się do podanego serwera" );
            System.out.println( " Otworzyć podany folder" );
            System.out.println( " Uruchomić wyszukiwanie w folderze za pomocą kryteriów domyślnych" );
            System.out.println( " Wyświetlić liczbę trafień" );
            System.out.println( " Wyświetlić liczbę uwag powiązanych z pierwszym trafieniem" );
            System.out.println( " Dodać nową uwagę o następujących atrybutach:" );
            System.out.println( " Określony tekst uwagi" );
            System.out.println( " OkToCopy=false" );
            System.out.println( " Public=false (tj. uwaga prywatna)" );
            System.out.println( " Pusta nazwa grupy" );
            System.out.println( " " );
            System.out.println( "-----" );
            System.out.println( " " );

            //-----
            // Zalogowanie do podanego serwera
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[5], "TcAddNote.java" );
            System.out.println( "Zalogowanie do " + argv[0] + "..." );
            if ( argv.length == 6 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

            //-----
            // Otwarcie podanego folderu i wyszukiwanie za pomocą kryteriów domyślnych
            //-----
            System.out.println( "Otwieranie folderu " + argv[3] + "..." );
            odFolder = odServer.openFolder( argv[3] );
            System.out.println( "Wyszukiwanie za pomocą kryteriów domyślnych..." );
            odFolder.search( );
            hits = odFolder.getHits( );
            System.out.println( " Liczba trafień: " + hits.size( ) );

            //-----
            // Dodanie nowej uwagi
            //-----
            if ( hits.size( ) > 0 )
            {
                odHit = (ODHit)hits.elementAt( 0 );
                notes = odHit.getNotes( );
                System.out.println( " Liczba uwag dla pierwszego trafienia " + notes.size( ) );

                odNote = new ODNote( );
                odNote.setText( argv[4] );
                odNote.setGroupName( " " );
                odNote.setOkToCopy( false );
                odNote.setPublic( false );

                System.out.println( " Dodawanie nowej uwagi z:" );
                System.out.println( " Tekst=" + odNote.getText( ) + " " );
                System.out.println( " Czy kopiować=" + odNote.isOkToCopy( ) );
                System.out.println( " Publiczna=" + odNote.isPublic( ) );
                System.out.println( " Grupa=" + odNote.getGroupName( ) );

                odHit.addNote( odNote );
            }
            else
                System.out.println( "Brak dokumentu - nie można wyświetlić uwag" );

            //-----
            // Procedura czyszcząca
            //-----
            odFolder.close( );
            odServer.logoff( );
            odServer.terminate( );
            System.out.println( " " );
            System.out.println( "-----" );
            System.out.println( " " );
            System.out.println( "Test zakończony - sprawdź, czy nowa uwaga została poprawnie" );
            System.out.println( " dodana, wyświetlając ją za pomocą klienta dla systemu Windows" );
            System.out.println( " " );
        }

        catch ( ODEException e )
        {
            System.out.println( "Klasa ODEException: " + e );
            System.out.println( " ID = " + e.getErrorId( ) );
            System.out.println( " komunikat = " + e.getErrorMsg( ) );
            e.printStackTrace( );
        }
    }
}

```

```

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}

```

---

## Aktualizowanie dokumentu

W poniższym przykładzie pokazano, jak aktualizować dokument.

Użyto w nim metod ODServer, ODFolder i ODCriteria w celu podłączenia się do serwera przy użyciu określonego identyfikatora użytkownika i hasła, otwarcia podanego folderu, ustawienia wartości wyszukiwania w dwóch polach wyszukiwania, wyczyszczenia pola Data wyszukiwania oraz uruchomienia wyszukiwania w folderze. Następnie użyte są metody ODHit dla dokumentu, który pasuje do zapytania, w celu zaktualizowania jednej lub wielu wartości w bazie danych.

Poniższy przykład ilustruje użycie następujących metod ODServer:

- initialize
- logon
- openFolder
- logoff
- terminate

Poniższy przykład ilustruje użycie następujących metod ODFolder:

- getName
- getDisplayOrder
- getCriteria
- search
- closeinitialize

Poniższy przykład ilustruje użycie następujących metod ODCriteria:

- setOperand
- setSearchValue

Poniższy przykład ilustruje użycie następujących metod ODHit:

- getDisplayValue
- update

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nazwa folderu
- Nazwa kryterium 1
- Wartość wyszukiwania 1
- Nazwa kryterium 2
- Wartość wyszukiwania 2
- Nowa wartość wyszukiwania, zastępująca wartość wyszukiwania 2
- Katalog konfiguracyjny (położenie pliku arswwww.ini)

Przykład aktualizowania dokumentu:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcUpdate
{

```



```

public static void main ( String argv[] )
{
    ODServer odServer;
    ODFolder odFolder;
    ODCriteria odCrit;
    ODHit odHit;
    Hashtable hash;
    Vector hits;
    String[] display_crit;
    String line, crit1, crit2, value1, value2, new_value;
    int j;

    //-----
    // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
    //-----
    if ( argv.length < 10 )
    {
        System.out.println( "składnia: java TcUpdate <server> <ID_użytkownika> <hasło> <folder> <kryterium1>
            <kryterium2> + "<kryterium2> <wartość2> <nowa_wartość2>" );
        return;
    }

    try
    {
        System.out.println( "Ten test powinien:" );
        System.out.println( " Zalogować się do podanego serwera" );
        System.out.println( " Otworzyć podany folder" );
        System.out.println( " Ustawić wartości wyszukiwania" );
        System.out.println( " Wyszukać w folderze" );
        System.out.println( " Zmienić wartość 2. podanego kryterium dla pierwszego trafienia" );
        System.out.println( " na nową" );
        System.out.println( "" );
        System.out.println( "Za pomocą klienta systemu Windows sprawdź, czy wartość została zmieniona." );
        System.out.println( "" );
        System.out.println( "-----" );
        System.out.println( "" );

        //-----
        // Zalogowanie do podanego serwera
        //-----
        odServer = new ODServer( );
        odServer.initialize( argv[9], "TcUpdate.java" );
        System.out.println( "Zalogowanie do " + argv[0] + "..." );
        odServer.logon( argv[0], argv[1], argv[2] );

        //-----
        // Otwarcie podanego folderu i ustawienie żądanych kryteriów
        //-----
        crit1 = argv[4];
        crit2 = argv[6];
        value1 = argv[5];
        value2 = argv[7];
        new_value = argv[8];
        System.out.println( "Otwieranie folderu " + argv[3] + "..." );
        odFolder = odServer.openFolder( argv[3] );
        odCrit = odFolder.getCriteria( crit1 );
        odCrit.setOperand( ODConstant.OPEqual );
        odCrit.setSearchValue( value1 );
        odCrit = odFolder.getCriteria( crit2 );
        odCrit.setOperand( ODConstant.OPEqual );
        odCrit.setSearchValue( value2 );

        //-----
        // Wyszukiwanie w folderze
        //-----
        System.out.println( " Wyszukiwanie dla " + crit1 + " = " + value1 + " i " + crit2 + " = " + value2
            + "..." );
        hits = odFolder.search( );

        //-----
        // Jeśli było co najmniej jedno trafienie
        //-----
        if ( hits != null && hits.size( ) > 0 )
        {
            //-----
            // Wyświetlenie wartości pierwszego trafienia
            //-----
            System.out.println( " Dla pierwszego trafienia:" );
            line = " ";
            display_crit = odFolder.getDisplayOrder( );
            for ( j = 0; j < display_crit.length; j++ )
                line = line + display_crit[j] + " ";
            System.out.println( line );
            line = " ";
            odHit = (ODHit)hits.elementAt( 0 );
            for ( j = 0; j < display_crit.length; j++ )
                line = line + odHit.getDisplayValue( display_crit[j] ) + " ";
            System.out.println( line );

            //-----
            // Utworzenie tabeli mieszającej istniejących par kryterium/wartość z wyjątkiem kryterium 2,
            // które przyjmie nową wartość. Aktualizacja wartości dla trafień.

```

```

//-----
System.out.println( " Wartość " + crit2 + " = " + value2 + " zastąpiono nową: " + crit2 + " = "
                    + new_value );
hash = new Hashtable( );
for ( j = 0; j < display_crit.length; j++ )
{
    if ( display_crit[j].equals( crit2 ) )
        hash.put( display_crit[j], new_value );
    else
        hash.put( display_crit[j], odHit.getDisplayValue( display_crit[j] ) );
}
odHit.update( hash );
}
else
    System.out.println( "Brak trafień" );

//-----
// Procedura czyszcząca
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Test zakończony - za pomocą klienta dla systemu Windows" );
System.out.println( " sprawdź, czy wartość została zmieniona." );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( " ID = " + e.getErrorId( ) );
    System.out.println( " komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}
}

```

---

## Zmiana hasła

W poniższym przykładzie użyto metody `changePassword` klasy `ODServer` w celu zmiany hasła podanego użytkownika na nowe. Zastosowano tu także metody `ODServer` w celu przygotowania do zalogowania i wylogowania się.

Poniższy przykład ilustruje użycie następujących metod `ODServer`:

- `initialize`
- `logon`
- `changePassword`
- `logoff`
- `terminate`

W przykładzie użyto następujących parametrów wykonawczych:

- Nazwa serwera
- Identyfikator użytkownika
- Hasło
- Nowe hasło
- Katalog konfiguracyjny (położenie pliku `arswww.ini`)
- (Opcjonalny) katalog serwera lokalnego

Przykład zmiany hasła:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcChangePassword
{
    public static void main ( String argv[] )
    {
        ODServer odServer;

```

```

String server, userid, original_password, new_password;

//-----
// Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
//-----
if ( argv.length < 5 )
{
    System.out.println( "składnia: java TcChangePassword <server> <ID_użytkownika> <hasło> <nowe_hasło> <katalog_konfiguracyjny>
        [<katalog_serwera_lokalnego>]" );
    return;
}

try
{
    //-----
    // Przygotowanie
    //-----
    System.out.println( "Ten test powinien:" );
    System.out.println( " Zalogować się do serwera za pomocą podanego hasła" );
    System.out.println( " Zmienić hasło na nowe" );
    System.out.println( " Wylogować się" );
    System.out.println( " Zalogować się do serwera za pomocą nowego hasła" );
    System.out.println( " Zmienić hasło z powrotem na stare" );
    System.out.println( " Wylogować się" );
    System.out.println( "" );
    System.out.println( "Jeśli program zostanie wykonany bez wyjątków, nie jest potrzebna" );
    System.out.println( "dalsza analiza." );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );

    //-----
    // Utworzenie podanego serwera
    //-----
    server = argv[0];
    userid = argv[1];
    original_password = argv[2];
    new_password = argv[3];
    odServer = new ODServer( );
    odServer.initialize( argv[4], "TcChangePassword.java" );

    //-----
    // Zalogowanie się do serwera za pomocą pierwotnego hasła
    //-----
    System.out.println( "Zalogowanie do serwera " + server + " za pomocą pierwotnego hasła..." );
    if ( argv.length == 5 )
        odServer.logon( server, userid, original_password );
    else
        if ( argv.length == 6 )
            odServer.logon( server, userid, original_password, ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

    //-----
    // Zmiana hasła na nowe i wylogowanie się
    //-----
    System.out.println( "Zmiana hasła na nowe..." );
    odServer.changePassword( new_password );
    System.out.println( "Wylogowanie..." );
    odServer.logoff( );

    //-----
    // Zalogowanie się do serwera za pomocą nowego hasła
    //-----
    System.out.println( "Zalogowanie do serwera " + server + " za pomocą nowego hasła..." );
    if ( argv.length == 5 )
        odServer.logon( server, userid, new_password );
    else
        if ( argv.length == 6 )
            odServer.logon( server, userid, new_password, ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

    //-----
    // Zmiana hasła z powrotem na stare i wylogowanie się
    //-----
    System.out.println( "Zmiana hasła z powrotem na stare..." );
    odServer.changePassword( original_password );
    System.out.println( "Wylogowanie..." );
    odServer.logoff( );

    //-----
    // Procedura czyszcząca
    //-----
    odServer.terminate( );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );
    System.out.println( "Program zakończony pomyślnie" );
    System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "Klasa ODEException: " + e );
    System.out.println( " ID = " + e.getErrorId( ) );
    System.out.println( " komunikat = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "wyjątek: " + e2 );
    e2.printStackTrace( );
}
}

```



---

## Dodatek E. Transformowanie AFP do HTML

Proces transformowania z AFP do HTML przekształca dokumenty i zasoby AFP w dokumenty HTML. Wymaga on dostępności usługi AFP2WEB Transform oferowanej przez IBM Printing Systems Division. Administrator musi zainstalować i skonfigurować usługę AFP2WEB Transform na serwerze HTTP. Więcej informacji na temat oferty AFP2WEB Transform można uzyskać u przedstawiciela IBM. Konieczne jest także określenie opcji konfiguracyjnych dla dokumentów i zasobów AFP, które mają być przekształcane za pomocą usługi AFP2WEB Transform. W tej sekcji opisano sposób określania opcji konfiguracyjnych.

**Uwaga:** W tym dokumencie nazwa AFP2HTML.INI oznacza plik konfiguracyjny. Aby określić plik zawierający opcje konfiguracyjne, zapoznaj się z sekcją “CONFIGFILE” na stronie 23.

Plik AFP2HTML.INI zawiera opcje konfiguracyjne usługi AFP2WEB Transform. Zazwyczaj konfiguruje się plik AFP2HTML.INI, podając opcje dla konkretnej aplikacji AFP. Można także określić zestaw opcji domyślnych. Usługa AFP2WEB Transform korzysta z nich podczas przekształcania dokumentów i zasobów dla aplikacji AFP, które nie zostały określone w pliku AFP2HTML.INI. Więcej szczegółów dotyczących opcji i procesu konwersji znajduje się w dokumentacji usługi AFP2WEB Transform.

Dodatkowe informacje dotyczące pliku AFP2HTML.INI znajdują się w następujących tematach:

- Format pliku AFP2HTML.INI
- Opcje usługi AFP2WEB Transform
- Wyświetlanie przekształconych dokumentów

**Uwaga:** Aby przekształcić dokumenty za pomocą apletu AFP2HTML, należy także określić parametr AFPVIEWING=HTML w sekcji DEFAULT BROWSER (lub sekcjach dotyczących innych przeglądarek) pliku ARSWWW.INI. Szczegóły znajdują się w sekcji “AFPVIEWING” na stronie 32. (Planując użycie funkcji API Retrieve Document (Wczytanie dokumentu), należy określić parametr \_afp=HTML. Szczegóły znajdują się w sekcji “Wczytanie dokumentu” na stronie 77.) Należy także określić katalog, w którym znajdują się programy AFP2WEB Transform (patrz “CONFIGFILE” na stronie 23).

---

### Format pliku AFP2HTML.INI

Poniżej przedstawiono przykład pliku AFP2HTML.INI:

```
[CREDIT-CREDIT]UseApplet=FALSE
ScaleFactor=1.0
CreateGIF=TRUE
SuppressFonts=FALSE
FontMapFile=creditFontMap.cfg
ImageMapFile=creditImageMap.cfg

[default]
ScaleFactor=1.0
CreateGIF=TRUE
SuppressFonts=FALSE
FontMapFile=fontmap.cfg
ImageMapFile=imagemap.cfg
```

Struktura tego pliku jest podobna do innych plików INI systemu Windows; zawiera jedną sekcję dla każdej aplikacji AFP oraz jedną sekcję domyślną. Wiersz tytułowy sekcji określa grupę aplikacji i aplikację. Na przykład wiersz tytułowy:

```
[CREDIT-CREDIT]
```

określa grupę aplikacji CREDIT i aplikację CREDIT. W wierszu tytułowym znak – (myślnik) oddziela nazwy. Nazwy te muszą odpowiadać nazwom grupy aplikacji i aplikacji zdefiniowanym dla serwera OnDemand. Jeśli grupa aplikacji zawiera więcej niż jedną aplikację, to dla każdej aplikacji należy utworzyć jedną sekcję.

Opcje w sekcji [default] (domyślne) są wykorzystywane przez usługę AFP2WEB Transform w celu przetwarzania dokumentów dla aplikacji AFP, które nie zostały zdefiniowane w pliku AFP2HTML.INI. Wartości domyślne są także stosowane, jeśli w sekcji dotyczącej określonej aplikacji AFP brak którejs z opcji.

Opcja UseApplet jest dyrektywą dla aplikacji ODWEK. Określa, czy aplet AFP2HTML będzie służył do wyświetlania danych wyjściowych z usługi AFP2WEB Transform. Wartością domyślną jest TRUE. Wartość FALSE (aplet AFP2HTML nie zostanie użyty do wyświetlenia danych wyjściowych) oznacza, że dane wyjściowe zostaną sformatowane i wyświetlone przez przeglądarkę WWW.

Pozostałe pięć opcji do dyrektywy dla usługi AFP2WEB Transform. W sekcji “Opcje usługi AFP2WEB Transform” w skrócie opisano, jak są one wykorzystywane przez tę usługę.

## Opcje usługi AFP2WEB Transform

Tabela 14 zawiera listę opcji, które można określić w pliku AFP2HTML.INI w celu przekształcenia dokumentów za pomocą usługi AFP2WEB Transform.

Tabela 14. Opcje usługi AFP2WEB Transform

Opcja w pliku AFP2HTML.INI	Opis
AllObjects	Określa sposób przetwarzania przez aplikację ODWEK dokumentów przechowywanych w serwerze OnDemand jako duże obiekty. Wartością domyślną jest 0 (zero). Oznacza ona, że aplikacja ODWEK wczyta tylko pierwszy segment dokumentu. Jeśli użytkownik określi 1 (jeden), to aplikacja ODWEK pobierze wszystkie segmenty i przekształci je przed wysłaniem dokumentu do klienta. <b>Uwaga:</b> Po włączeniu obsługi dużych obiektów dla bardzo dużych dokumentów zauważalne będzie znaczne opóźnienie w wyświetlaniu dokumentów na kliencie.
ScaleFactor	Skaluje dane wyjściowe o podany współczynnik skali. Wartością domyślną jest 1.0. Podanie wartości ScaleFactor=2,0 powoduje przeskalowanie danych wyjściowych i wyświetlenie dokumentu jako dwukrotnie większego w porównaniu z wielkością domyślną. Podanie wartości ScaleFactor=0,5 powoduje przeskalowanie dokumentu do połowy wielkości domyślnej. Wielkość domyślna jest uzyskiwana z ustawień parametru Zoom (Powiększenie) na stronie Logical Views (Widoki logiczne) w aplikacji OnDemand.
SuppressFonts	Określa, czy łańcuchy tekstowe AFP mają być transformowane. Wartość SuppressFonts=TRUE określa, że tekst pisany czcionką znajdującą się w pliku odwzorowania czcionek nie ma być transformowany. Wartością domyślną jest FALSE, co oznacza, że wszystkie łańcuchy tekstowe AFP mają być transformowane. Plik odwzorowania czcionek jest określony w opcji FontMapFile.

Tabela 14. Opcje usługi AFP2WEB Transform (kontynuacja)

Opcja w pliku AFP2HTML.INI	Opis
FontMapFile	Określa pełną nazwę ścieżki do pliku odwzorowania czcionek. Plik odwzorowania czcionek zawiera listę czcionek, które wymagają specjalnego przetwarzania. Domyślnie nosi on nazwę imagfont.cfg i mieści się w katalogu, w którym znajdują się programy usługi AFP2WEB Transform. Szczegóły dotyczące pliku odwzorowania czcionek znajdują się w dokumentacji usługi AFP2WEB Transform.
ImageMapFile	Określa plik odwzorowania obrazów. Może on służyć do usuwania obrazów z danych wyjściowych, poprawiania wyglądu obrazów zaciemnianych oraz zastępowania istniejących obrazów obrazami utworzonymi przez usługę AFP2WEB Transform. Odwzorowanie obrazów, które są takie same we wszystkich dokumentach AFP (na przykład logo firmy) skraca czas potrzebny na transformację dokumentów. Jeśli plik ten zostanie określony, musi znajdować się w katalogu, w którym mieszczą się programy usługi AFP2WEB Transform. Szczegóły dotyczące pliku odwzorowania obrazów znajdują się w dokumentacji usługi AFP2WEB Transform.

**Uwaga:** Podczas przekształcania dokumentów ODWEK wysyła poniższe opcje do usługi AFP2WEB Transform. Nie są one określone w pliku AFP2HTML.INI.

- Orientacja. Określa wartość rotacji, której należy użyć podczas wyświetlania dokumentu. Wartość domyślna jest uzyskiwana z ustawień parametru Orientation (Orientacja) na stronie View Information (Informacje widoku) w aplikacji OnDemand.
- Kolor obrazu. Określa, jakiego koloru użyć do wyświetlania obrazów i grafiki. Wartość domyślna jest uzyskiwana z ustawień parametru Image Color (Kolor obrazu) na stronie Logical Views (Widoki logiczne) w aplikacji OnDemand.

## Wyświetlanie przekształconych dokumentów

Opcja UseApplet w pliku AFP2HTML.INI jest dyrektywą dla aplikacji ODWEK, która określa, czy aplet AFP2HTML ma być używany do wyświetlania przekształconych danych wyjściowych. Wartością domyślną jest TRUE. Wartość FALSE (aplet AFP2HTML nie zostanie użyty do wyświetlenia danych wyjściowych) oznacza, że dane wyjściowe zostaną sformatowane i wyświetlone przez przeglądarkę WWW.

IBM zaleca korzystanie z apletu AFP2HTML do wyświetlania przekształconych dokumentów za każdym razem. Jeśli dokument jest przechowywany w serwerze OnDemand jako duży obiekt, to aplet AFP2HTML doda elementy sterujące ułatwiające przechodzenie do dowolnej strony dokumentu.





---

## Dodatek F. Transformowanie AFP do PDF

Usługa AFP2PDF Transform przekształca dokumenty i zasoby AFP w dokumenty PDF. Jest ona oferowana przez IBM Printing Systems Division. Administrator musi zainstalować i skonfigurować usługę AFP2PDF Transform na serwerze HTTP. Więcej informacji na temat oferty AFP2PDF Transform można uzyskać u przedstawiciela IBM. Konieczne jest także określenie opcji konfiguracyjnych dla dokumentów i zasobów AFP, które mają być przekształcane za pomocą usługi AFP2PDF Transform. W tej sekcji opisano sposób określania opcji konfiguracyjnych.

**Uwaga:** W tym dokumencie nazwa AFP2PDF.INI określa plik konfiguracyjny. Aby określić plik zawierający opcje konfiguracyjne, należy zapoznać się z sekcją “CONFIGFILE” na stronie 24.

Plik AFP2PDF.INI zawiera opcje konfiguracyjne usługi AFP2PDF Transform. Plik AFP2PDF.INI zazwyczaj konfiguruje się, podając opcje dla konkretnej aplikacji AFP. Można także określić zestaw opcji domyślnych. Usługa AFP2PDF Transform korzysta z nich podczas przekształcania dokumentów i zasobów dla aplikacji AFP, które nie zostały określone w pliku AFP2PDF.INI. Więcej szczegółów dotyczących opcji i procesu konwersji znajduje się w dokumentacji usługi AFP2PDF Transform.

Dodatkowe informacje dotyczące pliku AFP2PDF.INI znajdują się w następujących tematach:

- Określanie pliku AFP2PDF.INI
- Wyświetlanie przekształconych dokumentów

**Uwaga:** Aby przekształcić dokumenty, należy także określić parametr AFPVIEWING=PDF w sekcji DEFAULT BROWSER (lub sekcjach dotyczących innych przeglądarek) pliku ARSWWW.INI. Szczegóły znajdują się w sekcji “AFPVIEWING” na stronie 32. (Planując użycie funkcji API Retrieve Document (Wczytanie dokumentu), należy określić parametr \_afp=PDF. Szczegóły znajdują się w sekcji “Wczytanie dokumentu” na stronie 77.)

---

### Określanie pliku AFP2PDF.INI

Poniżej przedstawiono przykład pliku AFP2PDF.INI:

```
[CREDIT-CREDIT]OptionsFile=  
ImageMapFile=creditImageMap.cfg
```

```
[default]  
OptionsFile=  
ImageMapFile=imagemap.cfg  
AllObjects=0
```

Struktura tego pliku jest podobna do innych plików INI systemu Windows; zawiera jedną sekcję dla każdej aplikacji AFP oraz jedną sekcję domyślną. Wiersz tytułowy sekcji określa grupę aplikacji i aplikację. Na przykład wiersz tytułowy:

```
[CREDIT-CREDIT]
```

określa grupę aplikacji CREDIT i aplikację CREDIT. W wierszu tytułowym znak – (myślnik) oddziela nazwy. Nazwy te muszą odpowiadać nazwom grupy aplikacji i aplikacji zdefiniowanym dla serwera OnDemand. Jeśli grupa aplikacji zawiera więcej niż jedną aplikację, to dla każdej aplikacji należy utworzyć jedną sekcję.

Parametry określone w sekcji [default] (domyślne) są wykorzystywane przez usługę AFP2PDF Transform w celu przetwarzania dokumentów dla aplikacji AFP, które nie zostały zdefiniowane w pliku AFP2PDF.INI. Parametry domyślne są także stosowane, jeśli w sekcji dotyczącej określonej aplikacji AFP brak któregośkolwiek z podanych parametrów.

Parametr OptionsFile określa pełną nazwę ścieżki do pliku, który zawiera opcje transformowania używane przez usługę AFP2PDF Transform. Opcje transformowania są stosowane względem dokumentów AFP, które wymagają specjalnego przetwarzania. Szczegóły dotyczące pliku opcji transformowania czcionek znajdują się w dokumentacji usługi AFP2PDF Transform.

Parametr ImageMapFile określa plik odwzorowania obrazów. Może on służyć do usuwania obrazów z danych wyjściowych, poprawiania wyglądu obrazów zaciemnianych oraz zastępowania istniejących obrazów obrazami utworzonymi przez usługę AFP2PDF Transform. Odwzorowanie obrazów, które są takie same w większości dokumentów AFP (na przykład logo firmy), skraca czas potrzebny na transformację dokumentów. Jeśli plik ten zostanie określony, musi znajdować się w katalogu, w którym mieszczą się programy usługi AFP2PDF Transform. Aby określić katalog zawierający programy usługi AFP2PDF Transform, zapoznaj się z sekcją "INSTALLDIR" na stronie 24. Szczegóły dotyczące pliku odwzorowania obrazów znajdują się w dokumentacji usługi AFP2PDF Transform.

Parametr AllObjects określa sposób przetwarzania przez aplikację ODWEK dokumentów przechowywanych w serwerze OnDemand jako duże obiekty. Wartością domyślną jest 0 (zero). Oznacza ona, że aplikacja ODWEK wczyta tylko pierwszy segment dokumentu. Jeśli użytkownik określi 1 (jeden), to aplikacja ODWEK pobierze wszystkie segmenty i przekształci je przed wysłaniem dokumentu do klienta. **Uwaga:** Po włączeniu obsługi dużych obiektów dla bardzo dużych dokumentów zauważalne będzie znaczne opóźnienie w wyświetlaniu dokumentów na kliencie.

---

## Wyświetlanie przekształconych dokumentów

Aby wyświetlić przekształcone dokumenty za pomocą programu Adobe Acrobat, należy pobrać i zainstalować odpowiedni moduł dodatkowy dla przeglądarki WWW używanych w organizacji.

---

## Dodatek G. Pliki konfiguracyjne serwera HTTP

W tej sekcji znajdują się przykłady następujących plików konfiguracyjnych serwera HTTP:

- Serwer HTTP Apache
- WebSphere Application Server

**Uwaga:** Pomoc podczas konfigurowania można uzyskać, zapoznając się z dokumentacją serwera HTTP.

---

### Serwer HTTP Apache

Poniżej znajduje się przykładowy zbiór konfiguracyjny serwera HTTP. (Najpierw należy utworzyć konfigurację HTTP za pomocą klienta HTTP Admin.) Poniższa przykładowa konfiguracja dotyczy instancji QUSROND i serwera HTTP ODAPACHE. Opcje konfiguracyjne zapisane pogrubioną czcionką są wymagane dla lub pokrewne z aplikacją ODWEK. Notatki konfiguracyjne zapisane są pogrubioną kursywą w nawiasach. Notatek tych nie należy wprowadzać do zbioru konfiguracji. Wiersze komentarzy zaczynają się od znaku #. Nazwa zbioru i ścieżka w systemie IFS na serwerze, na którym znajduje się ten zbiór konfiguracji, to: /www/odapache/conf/httpd.conf

Uwagi:

1. 'Oryginalny' serwer HTTP nie jest już obsługiwany.
2. ODWEK wymaga, aby przeglądarka użytkownika końcowego akceptowała format UTF-8. W przeglądarce Internet Explorer firmy Microsoft, należy wybrać Narzędzia > Opcje internetowe, a następnie wybrać zakładkę Zaawansowane. W części Przeglądanie należy zaznaczyć opcję Zawsze wysyłaj adresy URL jako UTF-8.

```
=====
Listen *:ppppp    (Ustawić na port używany przez przeglądarkę do łączenia z aplikacją ODWEK.
                  Domyślnym numerem portu jest 80.)
DocumentRoot /www/odapache/htdocs
ServerRoot /www/odapache
DefaultType text/plain
Options -ExecCGI -FollowSymLinks -SymLinksIfOwnerMatch -Includes-IncludesNoExec
-Indexes -MultiViews
ErrorLog logs/error_log
LogLevel Warn
DirectoryIndex index.html
HostNameLookups off
RuleCaseSense OFF
LimitRequestBody 102400
ServerName f.q.h.n (Należy wpisać pełną nazwę hosta lub adres TCP/IP serwera,
                  na którym uruchomiony jest ODWEK.)
UseCanonicalName Off
DefaultFsCCSID 0037 (Należy podać identyfikator CCSID instancji OnDemand.)
DefaultNetCCSID 1208
CGIConvMode EBCDIC (W przypadku języków DBCS należy zmienić tę opcję na EBCDIC_JCD)
ScriptLogLength 200
ScriptLog /www/odapache/logs/cgi_log
Alias /logon /www/odapache/htdocs/oda_logon.html
(Powyzszy wiersz wskazuje nazwę skryptu logowania i jego położenie.)
AliasMatch ^/images/(.*)$ /www/odapache/htdocs/images/$1
AliasMatch ^/applets/com/ibm/edms/od/(.*)$ /QIBM/ProdData/OnDemand/www/applets/$1
AliasMatch ^/applets/(.*)$ /QIBM/ProdData/OnDemand/www/applets/$1
ScriptAliasMatch ^/scripts/arswww.cgi$ /QSYS.LIB/QRDARS.LIB/ARS3WCGI.PGM
ScriptAliasMatch ^/scripts/arswww\.cgi/(.*)$ /QSYS.LIB/QRDARS.LIB/ARS3WCGI.PGM
AddType www/unknown cab
AddType www/unknown jar
AlwaysDirectoryIndex On
DirectoryIndex index.html
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%{Cookie}n \"%r\" %t" cookie
LogFormat "%{User-agent}i" agent
```

```

LogFormat "%{Referer}i -> %U" referer
LogFormat "%h %l %u %t \"%r\" %>s %b" common
CustomLog logs/access_log combined
SetEnvIf "User-Agent" "Mozilla/2" nokeepalive
SetEnvIf "User-Agent" "JDK/1\0" force-response-1.0
SetEnvIf "User-Agent" "Java/1\0" force-response-1.0
SetEnvIf "User-Agent" "RealPlayer 4\0" force-response-1.0
SetEnvIf "User-Agent" "MSIE 4\0b2;" nokeepalive
SetEnvIf "User-Agent" "MSIE 4\0b2;" force-response-1.0

# Uprawnienia dostępu do katalogu głównego
<Directory />
Order Deny,Allow
    Deny From all
    Options None
    Options +ExecCGI
    AllowOverride NoneLimit
<Except GET HEAD OPTIONS TRACE POST PUT>
</LimitExcept>
</Directory>

# Dostęp do dokumentów katalogu serwera HTTP
<Directory /www/odapache/htdocs>
Order Allow,Deny
    Allow From all
</Directory>

# Dostęp do katalogu ProdData .... katalog apletu danych dla drukarki
<Directory /QIBM/ProdData/OnDemand/www/applets/>
Order Allow,Deny
    Allow From all
</Directory>

# Dostęp do katalogu ProdData .... katalog obrazów
<Directory /QIBM/ProdData/OnDemand/www/images>
Order Allow,Deny
    Allow From all
Options +FollowSymLinks
</Directory>

# Dostęp do katalogu ProdData .... katalog przykładów
<Directory /QIBM/ProdData/OnDemand/www/samples/>
Order Allow,Deny
    Allow From all
Options +FollowSymLinks
</Directory>

# Dostęp do katalogu instancji serwera HTTP
<Directory /www/odapache/>
Order Allow,Deny
    Allow From all
</Directory>

# Dostęp do katalogu biblioteki QRDARS
<Directory /QSYS.LIB/QRDARS.LIB>
Order Allow,Deny
    Allow From all
</Directory>

=====

```

---

## WebSphere Application Server

Przykładowy plik konfiguracyjny serwera WebSphere można uzyskać na stronie WWW wsparcia dla IBM Content Manager OnDemand for iSeries pod adresem: <http://www.ibm.com/software/data/ondemand/400/support.html>. W kategorii Technotes (Uwagi techniczne) należy znaleźć odsyłacz *ODWEK WebSphere example*.

---

## Dodatek H. Brak danych wyjściowych HTML

ODWEK używa dyrektywy `_nohtml` w celu określenia typu danych wyjściowych generowanych przez dowolną funkcję (na przykład Zalogowanie się). Domyślnie ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości `_nohtml=1` oznacza, ODWEK ma generować dane wyjściowe w formacie ASCII oddzielnym separatorami. W tym rozdziale opisano generowane przez ODWEK dane wyjściowe w formacie ASCII oddzielnym separatorami.

---

### Dane wyjściowe ASCII oddzielone separatorami

Generowane przez ODWEK dane wyjściowe ASCII oddzielone separatorami to zbiór rekordów wyjściowych zawierających wartości składające się z łańcuchów znaków, słów kluczowych oraz ograniczników i separatorów funkcji, rekordów i łańcuchów.

- Wartości składające się z łańcuchów znaków do dane wyjściowe funkcji inne niż słowa kluczowe, ograniczniki i separatory. Na przykład wartościami składającymi się z łańcuchów znaków są: następna funkcja do wywołania, nazwa folderu, nazwy pól folderu, operatory wyszukiwania i wartości pól.
- Słowa kluczowe to określone łańcuchy znaków. Słowa kluczowymi są na przykład: ACTION, DOC, FOLDER, NUMROWS i ROW.
- Ograniczniki funkcji to łańcuchy znaków [BEGIN] i [END].
- Ogranicznikiem rekordu jest znak nowego wiersza, `\n`. Wszystkie rekordy są oddzielone znakiem nowego wiersza.
- Domyślnie ogranicznikami i separatorami łańcuchów są: daszek (`^`) oraz lewy i prawy nawias kwadratowy (`[ i ]`). Na przykład:  
`[nazwa_folderu^opis_folderu]`

Jeśli słowo kluczowe zawiera więcej niż jedną wartość składającą się z łańcuchów znaków, wartości te są oddzielone daszkiem. Każdy zestaw wartości składający się na słowo kluczowe jest oddzielony lewym i prawym nawiasem kwadratowym.

Niektóre wartości składające się z łańcuchów znaków mogą być zapisywane jako lista, oddzielone daszkiem i zamknięte w nawiasy kwadratowe. Na przykład lista poprawnych operatorów wyszukiwania dla pola może wyglądać następująco:

```
[1^2^4^8^16^32]
```

Domyślne znaki ograniczników i separatorów łańcuchów można przesłonić. Szczegóły znajdują się w sekcji “[NO HTML]” na stronie 30.

- Łańcuch pusty oznaczają dwa cudzysłowy bez żadnej wartości pomiędzy nimi (`""`). Listę pustą oznaczają lewy i prawy nawias kwadratowy bez żadnej wartości pomiędzy nimi (`[ ]`).

---

### Zalogowanie się

Poniżej przedstawiono przykład danych wyjściowych ASCII oddzielonych separatorami wygenerowanych przez funkcję Zalogowanie się:

```
[BEGIN]\nACTION=searchCriteriaUrl\nFOLDER=[nazwa_folderu^opis_folderu]\nFOLDER=[nazwa_folderu^opis_folderu]\n
```

⋮  
[END]\n

## Uwagi

1. Łańcuch `searchCriteriaUrl` określa nazwę następnej funkcji do wykonania i przekazywane do niej parametry.
2. Łańcuch `nazwa_folderu` określa nazwę folderu. Nie jest ona ujmowana w cudzysłowy.
3. Łańcuch `opis_folderu` jest opisem folderu. Opis nie jest ujmowany w cudzysłowy.

---

## Kryteria wyszukiwania

Poniżej przedstawiono przykład danych ASCII oddzielonych separatorami wygenerowanych przez funkcję Kryteria wyszukiwania:

```
[BEGIN]\nACTION=hitListUrl\nDISPLAY_ORDER=[pole1^pole2^...poleN]\nNUMROWS=liczba_wierszy\nROW=[nazwa_kryterium^[[poprawne_operatory]^operator_domyślny]^  
[typ^powiązane_dane]\n⋮  
[END]\n
```

## Uwagi

1. Łańcuch `hitListUrl` określa nazwę następnej funkcji do wykonania i przekazywane do niej parametry.
2. Słowo kluczowe `DISPLAY_ORDER` określa kolejność wyświetlania pól folderu.
3. Łańcuch `liczba_wierszy` określa liczbę rekordów w słowie kluczowym `ROW`, które następuje dalej. Dla każdego pola wyszukiwania funkcja generuje jedno słowo kluczowe `ROW`.
4. Łańcuch `nazwa_kryterium` określa kryteria wyszukiwania pola. Kryteria wyszukiwania nie są ujmowane w cudzysłowy.
5. Łańcuch `poprawny_operator` jest listą liczb całkowitych, które odpowiadają poprawnym operatorom wyszukiwania dla danego pola:

<b>1</b>	Równe (Equal)
<b>2</b>	Nierówne (Not Equal)
<b>4</b>	Mniejsze niż (Less Than)
<b>8</b>	Mniejsze lub równe (Less Than or Equal)
<b>16</b>	Większe niż (Greater Than)
<b>32</b>	Większe lub równe (Greater Than or Equal)
<b>64</b>	W (In)
<b>128</b>	Poza (Not In)
<b>256</b>	Podobne (Like)
<b>512</b>	Niepodobne (Not Like)
<b>1024</b>	Pomiędzy (Between)
<b>2048</b>	Spoza zakresu (Not Between)
6. Łańcuch `operator_domyślny` jest liczbą całkowitą odpowiadającą domyślnemu operatorowi wyszukiwania.
7. Łańcuch `typ` oznacza typ pola wyszukiwania:

<b>A</b>	Wyszukiwanie tekstu adnotacji
----------	-------------------------------

- C Wybór
- N Normalne
- S Segment
- T Wyszukiwanie tekstowe
- Z Wyszukiwanie koloru adnotacji

8. Łańcuch powiązane\_dane jest listą powiązaną z wartościami operator\_domyślny i typ. Patrz Tabela 15.

Tabela 15. Operator domyślny i typ danych wejściowych powiązane z wartością powiązane\_dane

operator_domyślny	typ	powiązane_dane
Pomiędzy, Spoza zakresu (Between, Not Between)	N	Wartość pusta: [ ] lub lista: [pole_domyślne1^...^pole_domyślneN] Na przykład: ["31/01/96"^"31/01/97"] ["31/01/96"^" [""^"31/01/97"]
Inne poprawne operatory	A, N, T, Z	Wartość pusta: [ ] lub pojedyncza wartość łańcucha odpowiadająca domyślnej wartości pola
Inne poprawne operatory	C, S	[ [lista_wyborów]^domyślny_wyбір] Na przykład: [["JFIF"^"TIFF"^"PCX"]^"TIFF"] [["JFIF"^"TIFF"^"PCX"]^" ["JFIF"^"TIFF"^"PCX"]^" ["JFIF"^"TIFF"^"PCX"]^" ["JFIF"^"TIFF"^"PCX"]^"

## Lista trafień dokumentów

Poniżej przedstawiono przykład danych wyjściowych ASCII oddzielonych separatorami wygenerowanych przez funkcję Lista trafień dokumentów:

```
[BEGIN]\n
ACTION=hitListURL\n
MSG=W tym folderze można wyświetlić tylko 20 dokumentów.
DOC=[kryterium1^kryterium2^kryteriumN^ID_dokumentu^typ_pliku^
położenie_dokumentu]\n
:
[END]\n
```

## Uwagi

1. Łańcuch hitListURL określa nazwę następnej funkcji do wykonania i przekazywane do niej parametry.
2. Słowo kluczowe MSG to przykład komunikatu o błędzie w danych wyjściowych ASCII oddzielonych separatorami. Domyślnie ODWEK wysyła komunikat o błędzie do klienta. Jeśli jednak funkcja zawiera dyrektywę \_nohtml=1, ODWEK zamiast tego generuje tekst komunikatu w danych wyjściowych ASCII oddzielonych separatorami.
3. Łańcuchy kryterium1, kryterium2 i kryteriumN oznaczają wartości kryteriów wyszukiwania. Są one zwracane w takiej samej kolejności, jak na liście dokumentów. Nie są ujmowane w cudzysłowy.
4. Łańcuch ID\_dokumentu jest identyfikatorem dokumentu.
5. Łańcuch typ\_pliku określa typ danych dokumentu:
  - A AFP
  - B BMP

<b>E</b>	e-mail
<b>F</b>	JFIF
<b>G</b>	GIF
<b>L</b>	dane wierszowe
<b>N</b>	Brak
<b>O</b>	zdefiniowane w OnDemand
<b>P</b>	PDF
<b>T</b>	TIFF
<b>U</b>	zdefiniowane przez użytkownika
<b>X</b>	PCX

6. Łańcuch `położenie_dokumentu` określa miejsce dokumentu:

<b>0</b>	Nieznane
<b>1</b>	Pamięć podręczna OnDemand
<b>2</b>	Pamięć archiwalna
<b>3</b>	Zewnętrzna pamięć podręczna

---

## Podgląd adnotacji

Poniżej przedstawiono przykład danych wyjściowych ASCII oddzielonych separatorami wygenerowanych przez funkcję Podgląd adnotacji:

```
[BEGIN]\n
NOTE 4: 15:42:44 czasu środkowoeuropejskiego czwartek 19 listopada 1998...\n
Publiczne - nie można skopiować do innego serwera\n
Uwaga testowa z klienta internetowego OnDemand\n
[END]\n
```

---

## Komunikat o błędzie

Poniżej przedstawiono przykład danych wyjściowych ASCII oddzielonych separatorami wygenerowanych w momencie wystąpienia błędu:

```
[ERROR]\n
ID=nnnn\n
MSG=tekst_komunikatu_o_błędzie\n
```

## Uwagi

1. Łańcuch `nnnn` to numer komunikatu o błędzie.
2. Łańcuch `tekst_komunikatu_o_błędzie` to tekst komunikatu o błędzie.



## Dodatek I. Obsługa języków narodowych

W tej sekcji znajdują się informacje, które mogą pomóc administratorom w skonfigurowaniu produktu ODWEK do obsługi języków DBCS.

Do określania opcji konfiguracyjnych języka narodowego (NL) służą parametry CODEPAGE i LANGUAGE w pliku ARSWWW.INI.

Parametr CODEPAGE określa stronę kodową serwera ODWEK i musi być kompatybilny z bazą danych OnDemand i serwerem bibliotecznym OnDemand. Parametr ten należy określić **tylko** wtedy, gdy strona kodowa stacji roboczej, na której uruchamiana jest aplikacja ODWEK, jest inna niż strona kodowa bazy danych OnDemand na serwerze bibliotecznym OnDemand. Jako wartości domyślnej system używa strony kodowej stacji roboczej, na której uruchamiana jest aplikacja ODWEK.

Parametr LANGUAGE określa katalog komunikatów, którego produkt ODWEK używa do wyświetlania komunikatów.

Tabela 16 zawiera listę stron kodowych i języków DBCS obsługiwanych przez produkt OnDemand. W kolumnie **CODEPAGE=** znajdują się wartości określające stronę kodową; podaje się je **tylko** wtedy, gdy strona kodowa stacji roboczej, na której uruchamiana jest aplikacja ODWEK, różni się od strony kodowej bazy danych OnDemand. W kolumnie **LANGUAGE=** znajdują się wartości określające katalogi z przetłumaczonymi komunikatami.

**Uwaga:** W tabeli brak systemu Linux, ponieważ nie może on być serwerem docelowym aplikacji ODWEK. Innymi słowy, aplikacja ODWEK może działać w systemie Linux, ale baza danych OnDemand (serwer biblioteczny OnDemand) - nie.

Tabela 16. Strony kodowe, języki, zestawy kodów i ustawienia narodowe DBCS

Terytorium	LANGUAGE=	System operacyjny	Strona kodowa bazy danych	CODEPAGE=	Zestaw kodowy	Ustawienia narodowe
Chiny (PRC)	CHS	AIX	1383	1383	IBM_eucCN	zh_CN
		HP-UX	1383	1383	hp15CN	zh_CN. hp15CN
		Solaris	1383	1383	gb2312	zh
		Windows	1386	1386	GBK	—
		z/OS lub OS/390 (EBCDIC)	935	935	IBM-935	—
Japonia	JPN	AIX	954	954	IBM_eucJP	ja_JP
		HP-UX	954	954	eucJP	ja_JP.eucJP
		Solaris	954	954	eucJP	ja
		Windows	943	943	IBM-943	—
		z/OS lub OS/390 (EBCDIC)	939	939	IBM-939	—

Tabela 16. Strony kodowe, języki, zestawy kodów i ustawienia narodowe DBCS (kontynuacja)

Terytorium	LANGUAGE=	System operacyjny	Strona kodowa bazy danych	CODEPAGE=	Zestaw kodowy	Ustawienia narodowe
Korea Południowa	KOR	AIX	970	970	IBM_eucKR	ko_KR
		HP-UX	970	970	eucKR	ko_KR.eucKR
		Solaris	970	970	5601	ko
		Windows	1363	1363	1363	—
		z/OS lub OS/390 (EBCDIC)	933	933	IBM-933	—
Tajwan	CHT	AIX	964	964	IBM_eucTW	zh_TW
		HP-UX	964	964	eucTW	zh_TW.eucTW
		Solaris	964	964	cns11643	zh_TW
		Windows	950	950	big5	—
		z/OS lub OS/390 (EBCDIC)	937	937	IBM-937	—

Więcej informacji na temat konfigurowania systemu OnDemand do obsługi języków DBCS znajduje się w rozdziale “Obsługa języków narodowych” w publikacji *Planowanie i instalowanie*.

## Dodatek J. Narzędzia do określania problemu

Tabela 17 zawiera listę narzędzi, których można użyć do zbierania informacji o systemie i dokumentach. Informacje te mogą pomóc administratorowi w rozwiązywaniu problemów związanych z konfigurowaniem pakietu ODWEK, a innym użytkownikom w rozwiązywaniu problemów związanych z obsługą apletów i modułów dodatkowych.

Tabela 17. Narzędzia do określania problemu

Narzędzie	Zastosowanie	Jak włączyć
Dane wyjściowe HTML	Zapisanie kopii danych HTML, które aplikacja ODWEK wysłała do przeglądarki.	Z menu Plik przeglądarki wybierz opcję Zapisz jako.
Pliki protokołu serwera	Zapisanie informacji o próbach dostępu, błędach i informacjach wysłanych przez serwer.	<p>Wykonaj następujące czynności:</p> <ol style="list-style-type: none"> <li>1. W sekcji DEBUG pliku ARSWWW.INI nadaj parametrowi LOG wartość 1 (jeden). Wygenerowany przez aplikację ODWEK plik protokołu nosi nazwę ARSWWW.LOG i znajduje się w katalogu określonym przez parametr LOGDIR. (Domyślnym katalogiem jest /QIBM/UserData/OnDemand/WWW/LOG). <b>Ważne:</b> Jeśli sekcja DEBUG jest określona, musi być pierwszą instrukcją wykonywalną w pliku ARSWWW.INI.</li> <li>2. Skonfiguruj protokołowanie w serwerze HTTP. (W każdym serwerze HTTP konfigurowanie protokołowania przebiega inaczej, dostępne są także różne protokoły i opcje, które umożliwiają zebranie bardziej lub mniej szczegółowych informacji).</li> </ol> <p><b>Uwaga:</b> Ponieważ w pliku protokołu może znaleźć się znaczna ilość informacji, IBM zaleca włączanie protokołowania tylko w razie potrzeby, na przykład podczas próby odtworzenia sytuacji, w której wystąpił problem. Jeśli zachodzi konieczność włączenia protokołowania na dłuższy czas, należy upewnić się, że ścieżki do plików protokołu określają urządzenia pamięci masowej, na których jest dużo wolnego miejsca. Należy pamiętać o okresowym usuwaniu starych plików protokołu z serwera.</p>
Konsola języka Java	Wyświetlenie komunikatów wygenerowanych przez aplety.	<ul style="list-style-type: none"> <li>• Netscape: z menu przeglądarki wybierz Narzędzia, a następnie Konsola Java.</li> <li>• Internet Explorer: <ol style="list-style-type: none"> <li>1. Z menu Widok wybierz opcję Opcje internetowe.</li> <li>2. W zakładce Zaawansowane zaznacz opcję Konsola Java włączona.</li> <li>3. Zrestartuj przeglądarkę.</li> <li>4. Z menu Widok wybierz opcję Konsola Java.</li> </ol> </li> </ul>

Tabela 17. Narzędzia do określania problemu (kontynuacja)

Narzędzie	Zastosowanie	Jak włączyć
Funkcja śledzenia w przeglądarce AFP Web Viewer	Utrwalanie szczegółowych informacji dotyczących dokumentów AFP wyświetlanych za pomocą przeglądarki AFP Web Viewer.	Upewnij się, że w pliku FLDPORT2.INI na stacji roboczej użytkownika znajdują się następujące sekcje: [Misc] ViewTraceFile=d:\temp\afpplgin.log Trace=TRUE  Sprawdź ścieżkę do pliku protokołu. Pamiętaj, aby po zebraniu potrzebnych informacji wyłączyć protokołowanie.
Protokół systemowy serwera OnDemand	Zapisanie komunikatów systemowych (na przykład zalogowanie się i wylogowanie) oraz komunikatów grupy aplikacji związanych z dokumentami (na przykład zapytanie i wczytanie) i adnotacjami.	Wykonaj następujące czynności: 1. Włącz protokołowanie systemowe i grupy aplikacji dla serwera OnDemand. Za pomocą klienta administracyjnego zaktualizuj parametry systemowe dla serwera. 2. Włącz komunikaty konkretnej grupy aplikacji, którą chcesz protokołować. Za pomocą klienta administracyjnego zaktualizuj opcje protokołowania komunikatów dla tej grupy aplikacji.

---

## Uwagi

Niniejsza publikacja została przygotowana z myślą o produktach i usługach oferowanych w Stanach Zjednoczonych.

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 IBM. Jednakże cała odpowiedzialność za ocenę przydatności i sprawdzenie działania produktu, programu lub usługi, pochodzących od producenta innego niż IBM, spoczywa na użytkowniku.

IBM może posiadać patenty lub złożone wnioski patentowe na towary i usługi, o których mowa w niniejszej publikacji. Przedstawienie tej publikacji nie daje żadnych uprawnień licencyjnych do tychże patentów. Pisemne zapytania w sprawie licencji można przysyłać na adres:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
USA

Zapytania w sprawie licencji na informacje dotyczące zestawów znaków dwubajtowych (DBCS) należy kierować do lokalnych działów własności intelektualnej IBM (Intellectual Property Department) lub zgłaszać na piśmie pod adresem:

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 ZNAJDUJE ("AS IS") BEZ UDZIELANIA JAKICHKOLWIEK GWARANCJI (W TYM TAKŻE RĘKOJMI), WYRAŹNYCH 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 lub domniemanych w odniesieniu do 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. 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 dla tego produktu IBM, a użytkownik korzysta z nich na własną odpowiedzialność.

IBM ma prawo do używania i rozpowszechniania informacji przysyłanych przez użytkownika w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

Informacje na temat możliwości stosowania tego programu, takie jak: (i) wymiana informacji między niezależnie tworzonymi programami a innymi programami (włącznie z tym programem) czy (ii) wspólne używanie wymienianych informacji, można uzyskać pod adresem:

IBM Corporation  
Software Interoperability Coordinator  
3605 Highway 52 N  
Rochester, MN 55901–7829  
USA

Informacje takie mogą być udostępniane na odpowiednich warunkach, w niektórych przypadkach za opłatą.

Licencjonowany program opisany w niniejszej publikacji oraz 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 testowała tych produktów 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 W ZAKRESIE PRAW AUTORSKICH:

Publikacja ta zawiera przykładowe aplikacje w kodzie źródłowym, które ilustrują techniki programowania na różnych platformach systemowych. Użytkownik może kopiować, modyfikować i dystrybuować te programy przykładowe w dowolnej formie bez uiszczania opłat na rzecz IBM, w celu projektowania, używania, sprzedaży lub dystrybucji aplikacji zgodnych z aplikacyjnym interfejsem programowym dla tego systemu operacyjnego, dla którego napisane zostały programy przykładowe. Programy przykładowe nie zostały gruntownie przetestowane. IBM nie może zatem gwarantować lub sugerować niezawodności, użyteczności i funkcjonalności tych programów. Użytkownik może kopiować, modyfikować i dystrybuować te programy przykładowe w dowolnej formie bez uiszczania opłat na rzecz IBM w celu projektowania, używania, sprzedaży lub dystrybucji aplikacji zgodnych z aplikacyjnym interfejsem programowym IBM.

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

---

## Znaki towarowe i znaki usług

Advanced Function Presentation, AFP, AS/400, i5/OS, IBM, iSeries, Operating System/400, OS/390, Redbooks, WebSphere oraz z/OS są znakami towarowymi International Business Machines Corporation w Stanach Zjednoczonych i/lub w innych krajach.

Java oraz wszystkie znaki towarowe i logo dotyczące języka Java są znakami towarowymi lub zastrzeżonymi znakami towarowymi Sun Microsystems, Inc. w Stanach Zjednoczonych i/lub w innych krajach.

Lotus jest znakiem towarowym Lotus Development Corporation w Stanach Zjednoczonych i/lub w innych krajach.

Microsoft, Windows oraz Windows NT są zastrzeżonymi znakami towarowymi Microsoft Corporation w Stanach Zjednoczonych i/lub w innych krajach.

Nazwy innych przedsiębiorstw, produktów i usług mogą być znakami towarowymi lub znakami usług innych podmiotów.





---

# Indeks

## Znaki specjalne

@SRV@\_DEFAULT, sekcja 13

@SRV@\_serwer, sekcja 14

## A

ADDEXTENSION, parametr 31

ADDFIELDSTODOCID, parametr 31

ADDNOTES, parametr 32

adnotacje 32, 37

API 60, 84

dane wyjściowe ASCII oddzielone separatorami 140

funkcje API języka Java 120, 122

opis funkcji 6, 7

parametry 60, 84

przykładowe wywołanie funkcji 62, 85

AFP, czcionki

odzworowanie 51

AFP, dokumenty

MIME, typ zawartości 26

przeglądanie 32

przekształcanie 32

typ multimediów 26

AFP2HTML, sekcja 22

AFP2PDF Transform

instalowanie 12

konfigurowanie 133

włączanie 24

AFP2PDF, sekcja 24

AFP2WEB Transform

instalowanie 12

konfigurowanie 129

włączanie 22

AFPVIEWING, parametr 32

aktualizowanie dokumentu

API 82

funkcje API języka Java 124

opis funkcji 7

parametry 82

przykładowe wywołanie funkcji 83

anulowanie wyszukiwania 106

Apache, serwer HTTP

pliki konfiguracyjne 135

API

adnotacje 60, 84

aktualizowanie dokumentu 82

dodanie adnotacji 60

dokument, lista trażeń 65

drukowanie dokumentu 73

funkcje API CGI 59

funkcje API języka Java 89

informacje diagnostyczne 94

informacje diagnostyczne i związane ze śledzeniem 94

klasy 91

kod przykładowy 91

kryteria wyszukiwania 80

obsługa wyjątków 94

odniesienia 59, 87, 89

podgląd adnotacji 84

podręcznik programowania 91

API (*kontynuacja*)

podręcznik programowania funkcji API języka Java 91

przykładowe 91

środowisko systemowe 92

tworzenie pakietów 91

wczytanie dokumentu 77

wydruk serwerowy 73

wylogowanie się 69

zalogowanie się 71

zmiana hasła 63

zmienne środowiskowe 92

aplet AFP2HTML w języku Java

APPLETCACHEDIR, parametr 15

informacje o 1, 5

instalowanie 45

obsługa dużych obiektów 23, 131

wymagania 12, 46

aplet AFP2PDF w języku Java

katalog 24

włączanie 24

aplet danych dla drukarki wierszowej w języku Java

APPLETCACHEDIR, parametr 15

informacje o 1, 4

instalowanie 45

wymagania 46

aplety 16

APPLETCACHEDIR, parametr 15

informacje o 1

instalowanie 45

katalog 24

obsługa dużych obiektów 23, 131

włączanie 24

wymagania 46

aplety w języku Java

APPLETCACHEDIR, parametr 15

informacje o 1, 4, 5

instalowanie 45

katalog 24

obsługa dużych obiektów 23, 131

włączanie 24

wymagania 46

aplikacje WWW

przykłady 43

aplikacyjny interfejs programistyczny (API)

*Patrz* API

APPLETCACHEDIR, parametr 15

APPLETDIR, parametr 16

ARSWWW.INI, plik

@SRV@\_DEFAULT, sekcja 13

@SRV@\_serwer, sekcja 14

ADDEXTENSION, parametr 31

ADDFIELDSTODOCID, parametr 31

ADDNOTES, parametr 32

AFP2HTML, sekcja 22

AFP2PDF Transform 24

AFP2PDF, sekcja 24

AFP2WEB Transform 22

AFPVIEWING, parametr 32

APPLET, parametr 16

APPLETCACHEDIR, parametr 15

ATTACHMENT IMAGES, sekcja 29

ARSWWW.INI, plik *(kontynuacja)*  
 AUTODOCRETRIEVAL, parametr 33  
 BEGIN, parametr 30  
 CACHEDIR, parametr 16  
 CACHEDOCS, parametr 16  
 CACHERMAXTHRESHOLD, parametr 17  
 CACHERMINTHRESHOLD, parametr 17  
 CACHERSIZE, parametr 17  
 CACHERUSERIDS, parametr 18  
 CODEPAGE, parametr 18  
 CONFIGFILE, parametr 23, 24  
 CONFIGURATION, sekcja 15  
 debug, sekcja 38  
 DEFAULT BROWSER, sekcja 31  
 DOCSIZE, parametr 19  
 EMAILVIEWING, parametr 33  
 ENCRYPTCOOKIES, parametr 34  
 ENCRYPTURL, parametr 34  
 END, parametr 30  
 FOLDERDESC, parametr 34  
 HOST, parametr 14  
 IMAGEDIR, parametr 19  
 INSTALLDIR, parametr 23, 24  
 konfigurowanie 13  
 LANGUAGE, parametr 20  
 LINEVIEWING, parametr 35  
 LOG, parametr 39, 143  
 LOGDIR, parametr 39  
 MAXHITS, parametr 35  
 MIMETYPES, sekcja 25  
 NOHTML, sekcja 30  
 NOLINKS, parametr 36  
 ODApplet.jre.path.IE, parametr 36  
 ODApplet.jre.path.NN, parametr 36  
 ODApplet.jre.version, parametr 36  
 ODApplet.version, parametr 36  
 określanie 13  
 PORT, parametr 14, 15  
 PROTOCOL, parametr 14, 15  
 przeglądarka, opcje 38  
 przeglądarka, sekcja 38  
 REPORTSERVERTIMEOUT, parametr 22  
 SECURITY, sekcja 21  
 SEPARATOR, parametr 31  
 SERVERACCESS, parametr 22  
 SERVERPRINT, parametr 36  
 SERVERPRINTERS, parametr 37  
 SHOWDOCLOCATION, parametr 37  
 ShowSearchString, parametr 20  
 TEMPDIR, parametr 21  
 TEMPLATEDIR, parametr 21  
 USEEXECUTABLE, parametr 23, 25  
 VIEWNOTES, parametr 37  
 atrybut method znacznika form 7  
 ATTACHMENT IMAGES, sekcja 29  
 AUTODOCRETRIEVAL, parametr 33

## B

BEGIN, parametr 30  
 błędy 94, 143  
 BMP, dokumenty  
 MIME, typ zawartości 27  
 typ multimediów 27  
 BMP, załączniki 29  
 brak danych wyjściowych HTML 30, 137

## C

CACHEDIR, parametr 16  
 CACHEDOCS, parametr 16  
 CACHERMAXTHRESHOLD, parametr 17  
 CACHERMINTHRESHOLD, parametr 17  
 CACHERSIZE, parametr 17  
 CACHERUSERIDS, parametr 18  
 CODEPAGE, parametr 18, 141  
 CONFIGFILE, parametr 23, 24  
 CONFIGURATION, sekcja 15  
 czcionki  
 AFP 51  
 odwzorowanie 51  
 TrueType 51

## D

dane dla drukarki wierszowej  
 MIME, typ zawartości 28  
 przeglądanie 35  
 przekształcanie 35  
 typ multimediów 28  
 dane wyjściowe ASCII  
 adnotacje 140  
 dokument, lista trafień 139  
 format 137  
 generowane przez OnDemand 137  
 komunikat o błędzie 140  
 komunikaty 140  
 kryteria wyszukiwania 138  
 podgląd adnotacji 140  
 zalogowanie się 137  
 dane wyjściowe ASCII oddzielone separatorami  
 adnotacje 140  
 dokument, lista trafień 139  
 format 137  
 generowane przez OnDemand 137  
 komunikat o błędzie 140  
 komunikaty 140  
 kryteria wyszukiwania 138  
 podgląd adnotacji 140  
 separatory 30  
 zalogowanie się 137  
 DBCS 141  
 debug, sekcja 38  
 DEFAULT BROWSER, sekcja 31  
 DOCSIZE, parametr 19  
 dodanie adnotacji  
 API 60  
 opis funkcji 6  
 parametry 60  
 przykładowe wywołanie funkcji 62  
 dokument, lista trafień  
 API 65  
 dane wyjściowe ASCII oddzielone separatorami 139  
 funkcje API języka Java 100, 104, 113, 115  
 opis funkcji 6  
 parametry 65  
 przykładowe wywołanie funkcji 68  
 dokumenty  
 AFP 32  
 aktualizowanie za pomocą funkcji API języka Java 124  
 dane dla drukarki wierszowej 35  
 drukowanie za pomocą funkcji API języka Java 118  
 dyskowa pamięć podręczna 16  
 EMAIL 33

- dokumenty (*kontynuacja*)
  - MIME, typ zawartości 25
  - odsyłacze 36
  - przeglądanie 32, 33, 35
  - przekształcanie 32, 33, 35
  - typ multimediiów 25
  - wczytywanie 33
- dokumenty, funkcje API języka Java 113, 115
- domyślna przeglądarka, opcje 31
- drukowanie
  - funkcje API języka Java 118
  - serwer 36, 37
- drukowanie dokumentu
  - API 73
  - funkcje API języka Java 118
  - opis funkcji 7
  - parametry 73
  - przykładowe wywołanie funkcji 76
- duże obiekty 23, 131
- dyskowa pamięć podręczna 16, 17, 18

## E

- EMAIL, dokumenty
  - MIME, typ zawartości 27
  - przeglądanie 33
  - przekształcanie 33
  - typ multimediiów 27
- EMAILVIEWING, parametr 33
- ENCRYPTCOOKIES, parametr 34
- ENCRYPTURL, parametr 34
- END, parametr 30

## F

- folder, wyszukiwanie za pomocą funkcji API języka Java 100, 104, 106, 108, 113
- folder, wyświetlanie grup aplikacji używających funkcji API języka Java 98
- FOLDERDESC, parametr 34
- funkcje
  - adnotacje 6, 7
  - aktualizowanie dokumentu 7
  - dodanie adnotacji 6
  - dokument, lista trafień 6
  - drukowanie dokumentu 7
  - kryteria wyszukiwania 6
  - podgląd adnotacji 7
  - serwerowy wydruk dokumentu 7
  - wczytanie dokumentu 6
  - wylogowanie się 6
  - zalogowanie się 6
  - zmiana hasła 6
- funkcje API CGI
  - odniesienia 59
- funkcje API języka Java
  - informacje o 1
  - odniesienia 89
  - podręcznik programowania 91
  - wymagania dotyczące oprogramowania 1

## G

- GIF, dokumenty
  - MIME, typ zawartości 27
  - typ multimediiów 27

- GIF, załączniki 30
- grupy aplikacji w folderze
  - funkcje API języka Java 98

## H

- hasła
  - funkcje API języka Java 96, 126
- host, nazwa 14
- HOST, parametr 14
- httpd.conf, plik 135

## I

- i5/OS
  - instalacja 12
- identyfikatory użytkowników
  - dyskowa pamięć podręczna 18
  - funkcje API języka Java 96
- Image Web Viewer
  - informacje o 1
  - instalowanie 45
  - wymagania 46
- IMAGEDIR, parametr 19
- informacje cookie 34, 46
- informacje diagnostyczne 94
- informacje diagnostyczne i związane ze śledzeniem 94
- informacje o produkcie OnDemand Internet Connection 1
- informacje o tej publikacji 1
- instalacja
  - aplet AFP2HTML w języku Java 45
  - aplet danych dla drukarki wierszowej w języku Java 45
  - aplety 45
  - aplety w języku Java 45
  - ARSWWW.INI, plik 13
  - dostosowywanie 47
- i5/OS 12
- Image Web Viewer 45
- moduły dodatkowe 45
- oprogramowanie ODWEK 11
- pliki zdefiniowane przez użytkownika 47
- przeglądarka AFP Web Viewer 45
- serwer HTTP 11
- stacja robocza użytkownika 45
- wymagania 11

INSTALLDIR, parametr 23, 24

## J

- język 20, 141
- języki używające zestawu znaków dwubajtowych 141
- JFIF, dokumenty
  - MIME, typ zawartości 27
  - typ multimediiów 27
- JVM 46

## K

- katalog lokalny
  - funkcje API języka Java 96
- katalog obrazów 19
- klasy 91
- kod przykładowy 91
- kommunikacja, protokoły 14, 15

- komunikat o błędzie
  - dane wyjściowe ASCII oddzielone separatorami 140
- komunikaty 20
  - dane wyjściowe ASCII oddzielone separatorami 140
- konfigurowanie
  - ARSWWW.INI, plik 13
  - oprogramowanie ODWEK 11
  - plik konfiguracyjny AFP2HTML 129
  - plik konfiguracyjny AFP2PDF 133
  - serwer HTTP 11
- kryteria wyszukiwania
  - API 80
  - dane wyjściowe ASCII oddzielone separatorami 138
  - funkcje API języka Java 100, 104, 108
  - łańcuch SQL 104
  - opis funkcji 6
  - parametry 80
  - przykładowe wywołanie funkcji 81

## L

- LANGUAGE, parametr 20, 141
- limit czasu 22
- limit czasu nieaktywności 22
- LINEVIEWING, parametr 35
- lista dostępu do serwera 22
- Lista kontrolna
  - instalacji dla 9
- LOG, parametr 39, 143
- LOGDIR, parametr 39

## Ł

- łańcuch wyszukiwania SQL za pomocą funkcji API języka Java 104

## M

- maksymalna liczba trafień 35
- MAXHITS, parametr 35
- metoda GET 7
- metoda POST 7
- MIME, typ zawartości 25, 100
- MIMETYPES, sekcja 25
- moduły dodatkowe
  - informacje o 1
  - instalowanie 45

## N

- nawiązywanie połączenia z serwerem 95, 96
- nazwa aplikacji
  - funkcje API języka Java 96
- nazwa folderu, funkcje API języka Java 112
- NLS 18, 20, 141
- NOHTML, sekcja 30
- NOLINKS, parametr 36
- numer portu 14, 15

## O

- obsługa języka narodowego 141
- obsługa wyjątków 94
- ochrona 7, 21, 34
- ochrona danych 7
- ochrona serwera 7, 21

- ODApplet.jre.path.IE, parametr 36
- ODApplet.jre.path.NN, parametr 36
- ODApplet.jre.version, parametr 36
- ODApplet.version, parametr 36
- ODCallback 117
- ODCriteria
  - aktualizowanie dokumentu 124
  - dokumenty, aktualizowanie 124
  - nazwa 100
  - operandy 100, 106, 108
  - wartości wyszukiwania 100, 106, 108
- ODCriteria.getFixedValues 108
- ODCriteria.getName 100
- ODCriteria.getOperand 100, 106
- ODCriteria.getType 108
- ODCriteria.getValidOperands 108
- ODCriteria.getValues 108
- ODCriteria.setOperand 108, 124
- ODCriteria.setSearchValue 100, 124
- ODCriteria.setSearchValues 100, 106, 108
- ODFolder
  - anulowanie wyszukiwania 106
  - dokument, drukowanie 118
  - dokument, wczytanie 115
  - drukowanie dokumentów 118
  - grupy aplikacji 98
  - kolejność wyświetlania 100, 113
  - komunikat 100
  - kryteria 100, 106, 108
  - nazwa 100, 113
  - opis 100
  - wczytanie dokumentu 115
  - wyszukiwanie 100, 104, 106, 108, 113, 115
  - zamykanie 98, 100, 104, 106
- ODFolder.close 98, 100, 104, 106, 115
- ODFolder.getApplGroups 98
- ODFolder.getCriteria 100, 106, 108
- ODFolder.getDescription 100
- ODFolder.getDisplayOrder 100, 104, 113
- ODFolder.getName 100, 113
- ODFolder.getNumApplGroups 98
- ODFolder.getSearchMessage 100
- ODFolder.printDocs 118
- ODFolder.retrieve 115
- ODFolder.search 100, 104, 106, 113, 115
- ODFolder.setApplGroupForSearchWithSQL 104
- ODHit
  - adnotacje 120, 122
  - aktualizowanie dokumentów 124
  - dokument, aktualizowanie 124
  - dokument, wczytanie 115
  - lista dokumentów 113
  - MIME, typ zawartości 100
  - położenie dokumentu 100
  - typ dokumentu 100
  - uwagi 120, 122
  - wartości wyświetlania 100, 104
  - wartość wyświetlania 113
  - wczytanie dokumentu 115
- ODHit.addNote 122
- ODHit.getDisplayValue 100, 104, 113, 124
- ODHit.getDisplayValues 100
- ODHit.getDocId 100, 115
- ODHit.getDocLocation 100
- ODHit.getDocType 100
- ODHit.getMimeType 100
- ODHit.getNotes 120, 122

- ODHit.retrieve 115
- ODHit.update 124
- odłączanie od serwera 96
- odniesienia
  - API 59, 87, 89
  - funkcje API CGI 59
  - funkcje API języka Java 89
  - serwlet 87
  - serwlet w języku Java 87
- ODNote
  - adnotacje 120, 122
  - data 120
  - godzina 120
  - identyfikator użytkownika 120
  - kolor 120
  - nazwa grupy 120
  - pozycja 120
  - strona 120
  - tekst 120
- ODNote.getColor 120
- ODNote.getDateTime 120
- ODNote.getGroupName 120
- ODNote.getOffsetX 120
- ODNote.getOffsetY 120
- ODNote.getPageNum 120
- ODNote.getText 120
- ODNote.getUserid 120
- ODNote.isOkToCopy 120, 122
- ODNote.isPublic 120, 122
- ODNote.setGroupName 122
- ODNote.setText 122
- ODServer
  - anulowanie wyszukiwania 106
  - dokument, wczytanie 115
  - drukarki 118
  - drukarki serwera 118
  - folder, otwieranie 115
  - hasło 96, 126
  - identyfikator użytkownika 96
  - katalog lokalny 96
  - łączenie z 96
  - nawiązywanie połączenia z serwerem 95
  - nazwa aplikacji 96
  - nazwa folderu 112
  - odłączanie od 96
  - opis folderu 112
  - otwieranie folderu 108, 115
  - port 96
  - serwer 96
  - typ połączenia 96
  - ustawianie haseł 126
  - ustawianie i uzyskiwanie haseł 96
  - ustawianie i uzyskiwanie identyfikatorów użytkowników 96
  - wczytanie dokumentu 115
  - zmiana hasła 126
- ODServer.cancel 106
- ODServer.changePassword 126
- ODServer.getConnectType 96
- ODServer.getFolderNames 112
- ODServer.getFoldersDescription 112
- ODServer.getLocalDir 96
- ODServer.getNumFolders 112
- ODServer.getPassword 96
- ODServer.getPort 96
- ODServer.getServerName 96
- ODServer.getServerPrinters 118
- ODServer.getUserId 96
- ODServer.logoff 95, 96
- ODServer.logon 95, 96
- ODServer.openFolder 108, 115
- ODServer.retrieve 115
- ODServer.setApplicationName 96
- ODServer.setConnectType 96
- ODServer.setLocalDir 96
- ODServer.setPassword 96
- ODServer.setPort 96
- ODServer.setServer 96
- ODServer.setUserId 96
- ODServer.terminate 95, 96
- odsyłacze 36
- odwzorowywanie czcionek AFP 51
- określanie problemów 143
- OnDemand Internet Connection
  - informacje o 1
- OnDemand, opcje serwera
  - @SRV@\_DEFAULT, sekcja 13
  - @SRV@\_serwer, sekcja 14
  - HOST, parametr 14
  - parametry 14
  - PORT, parametr 14, 15
  - PROTOCOL, parametr 14, 15
  - wartości domyślne 13
- opcje serwera HTTP
  - AFP2HTML, sekcja 22
  - AFP2PDF Transform 24
  - AFP2PDF, sekcja 24
  - AFP2WEB Transform 22
  - APPLETDIR, parametr 16
  - ATTACHMENT IMAGES, sekcja 29
  - BEGIN, parametr 30
  - CACHEDIR, parametr 16
  - CACHEDOCS, parametr 16
  - CACHEMAXTHRESHOLD, parametr 17
  - CACHEMINTHRESHOLD, parametr 17
  - CACHESIZE, parametr 17
  - CACHEUSERIDS, parametr 18
  - CODEPAGE, parametr 18
  - CONFIGFILE, parametr 23, 24
  - CONFIGURATION, sekcja 15
  - debugowanie 38
  - domyślna przeglądarka 31
  - END, parametr 30
  - IMAGEDIR, parametr 19
  - INSTALLDIR, parametr 23, 24
  - LANGUAGE, parametr 20
  - MIMETYPES, sekcja 25
  - NOHTML, sekcja 30
  - przeglądarki 31, 38
  - REPORTSERVERTIMEOUT, parametr 22
  - SECURITY, sekcja 21
  - SEPARATOR, parametr 31
  - SERVERACCESS, parametr 22
  - ShowSearchString, parametr 20
  - TEMPDIR, parametr 21
  - TEMPLATEDIR, parametr 21
  - USEEXECUTABLE, parametr 23, 25
- opcje serwera WWW
  - DBCS 141
  - język 141
  - NLS 141
  - strona kodowa 141
- operandy, funkcje API języka Java 100
- opis folderu, funkcje API języka Java 112

oprogramowanie ODWEK  
  instalowanie 11  
oprogramowanie serwera HTTP  
  instalowanie 11

## P

pakiety, hierarchia w języku Java 91  
pamięć podręczna dla dokumentów 16  
pamięć podręczna, katalog 16  
pamięć tymczasowa 21  
parametry  
  @SRV@\_DEFAULT, sekcja 13  
  @SRV@\_serwer, sekcja 14  
  ADDEXTENSION 31  
  ADDFIELDSTODOCID 31  
  ADDNOTES 32  
  AFP2HTML, sekcja 22  
  AFP2PDF, sekcja 24  
  AFPVIEWING 32  
  APPLETCACHEDIR 15  
  APPLETDIR 16  
  ATTACHMENT IMAGES, sekcja 29  
  AUTODOCRETRIEVAL 33  
  BEGIN 30  
  CACHEDIR 16  
  CACHEDOCS 16  
  CACHEMAXTHRESHOLD 17  
  CACHEMINTHRESHOLD 17  
  CACHE SIZE 17  
  CACHEUSERIDS 18  
  CODEPAGE 18  
  CONFIGFILE 23, 24  
  CONFIGURATION, sekcja 15  
  DOCSIZE 19  
  EMAILVIEWING 33  
  ENCRYPTCOOKIES 34  
  ENCRYPTURL 34  
  END 30  
  FOLDERDESC 34  
  HOST 14  
  IMAGEDIR 19  
  INSTALLDIR 23, 24  
  LANGUAGE 20  
  LINEVIEWING 35  
  LOG 39, 143  
  LOGDIR 39  
  MAXHITS 35  
  NOLINKS 36  
  ODApplet.jre.path.IE 36  
  ODApplet.jre.path.NN 36  
  ODApplet.version 36  
  PORT 14, 15  
  PROTOCOL 14, 15  
  REPORTSERVERTIMEOUT 22  
  SECURITY, sekcja 21  
  SEPARATOR 31  
  SERVERACCESS 22  
  SERVERPRINT 36  
  SERVERPRINTERS 37  
  SHOWDOCLOCATION 37  
  ShowSearchString 20  
  TEMPDIR 21  
  TEMPLATEDIR 21  
  USEEXECUTABLE 23, 25  
  VIEWNOTES 37

PCX, dokumenty  
  MIME, typ zawartości 28  
  typ multimediiów 28  
PDF, dokumenty  
  MIME, typ zawartości 28  
  typ multimediiów 28  
plik konfiguracyjny AFP2HTML 129  
plik konfiguracyjny AFP2PDF 133  
pliki protokołów 39, 143  
pliki zdefiniowane przez użytkownika  
  instalowanie 47  
podgląd adnotacji  
  API 84  
  dane wyjściowe ASCII oddzielone separatorami 140  
  opis funkcji 7  
  parametry 84  
  przykładowe wywołanie funkcji 85  
podręcznik programowania  
  API 91  
  funkcje API języka Java 91  
położenie dokumentu 37  
pomoc 143  
port  
  funkcje API języka Java 96  
PORT, parametr 14, 15  
PROTOCOL, parametr 14, 15  
protokołowanie 39, 143  
protokoły 14, 15  
przegląd 1  
przeglądarka AFP Web Viewer  
  AFP, czcionki 51  
  czcionki 51  
  dostosowanie instalacji 47  
  informacje o 1  
  instalowanie 45  
  instalowanie plików zdefiniowanych przez użytkownika 47  
  konfigurowanie 51  
  odwzorowywanie czcionek AFP 51  
  pliki zdefiniowane przez użytkownika 47  
  wymagania 46  
przeglądarka danych dla drukarki wierszowej  
  konfigurowanie 52  
  ODApplet.jre.path.IE, parametr 36  
  ODApplet.jre.path.NN, parametr 36  
  ODApplet.jre.version, parametr 36  
  ODApplet.version, parametr 36  
przeglądarka danych dla drukarki wierszowej w języku Java  
  konfigurowanie 52  
  ODApplet.jre.path.IE, parametr 36  
  ODApplet.jre.path.NN, parametr 36  
  ODApplet.jre.version, parametr 36  
  ODApplet.version, parametr 36  
przeglądarka Image Web viewer  
  konfigurowanie 52  
przeglądarka, opcje  
  DEFAULT BROWSER, sekcja 31  
  przeglądarka, sekcja 38  
przeglądarka, sekcja 38  
przeglądarki  
  informacje cookie 46  
  JVM 46  
  obsługiwane 46  
  wirtualna maszyna języka Java 46  
przygotowanie do użycia produktu OnDemand Internet Connection 1  
przykładowe 91  
przykładowe aplikacje 43

## R

REPORTSERVERTIMEOUT, parametr 22

## S

SECURITY, sekcja 21  
SEPARATOR, parametr 31  
separatory 30  
separatory danych wyjściowych 30  
SERVERACCESS, parametr 22  
SERVERPRINT, parametr 36  
SERVERPRINTERS, parametr 37  
serwer  
funkcje API języka Java 96  
serwer HTTP  
httpd.conf, plik 135  
pliki konfiguracyjne 135  
serwlet  
odniesienia 87  
serwlet w języku Java  
odniesienia 87  
SHOWDOCLOCATION, parametr 37  
ShowSearchString, parametr 20  
strona kodowa 18, 141  
strony WWW  
przykłady 43  
szyfrowanie 34

## Ś

śledzenie problemów 143  
środowisko systemowe, funkcje API języka Java 92

## T

TCP/IP, protokół komunikacyjny 14, 15  
TEMPDIR, parametr 21  
TEMPLATEDIR, parametr 21  
TIFF, dokumenty  
MIME, typ zawartości 29  
typ multimediów 29  
TrueType, czcionki  
odwzorowywanie czcionek AFP na 51  
TXT, załączniki 30  
tymczasowy katalog roboczy 21  
typ dokumentu, funkcje API języka Java 100  
typ połączenia  
funkcje API języka Java 96  
typ/podtyp multimediów 25

## U

uprawnienia 12  
uprawnienia do katalogu 12  
USEEXECUTABLE, parametr 23, 25  
ustawianie haseł 126  
uwagi 32, 37

## V

VIEWNOTES, parametr 37

## W

wartości wyszukiwania, funkcje API języka Java 100, 104  
wczytanie dokumentu  
API 77  
opis funkcji 6  
parametry 77  
przykładowe wywołanie funkcji 79  
wczytywanie  
dokumenty 33  
wczytywanie dokumentu 115  
wielkość pamięci podręcznej 17  
wirtualna maszyna języka Java 46  
wydruk serwerowy  
API 73  
funkcje API języka Java 118  
opis funkcji 7  
parametry 73  
przykładowe wywołanie funkcji 76  
włączanie 36, 37  
wylogowanie się  
API 69  
opis funkcji 6  
parametry 69  
przykładowe wywołanie funkcji 70  
wymagania  
AFP2PDF Transform 12  
AFP2WEB Transform 12  
aplet AFP2HTML w języku Java 12  
dyskowa pamięć podręczna 11  
funkcje API języka Java 1  
informacje cookie 46  
pamięć podręczna dla dokumentów 11  
serwer 11  
serwer HTTP 11  
wirtualna maszyna języka Java 46  
wyniki zapytania 35  
wyszukiwanie w folderze 100, 104, 106, 108, 113  
wyświetlanie wartości, funkcje API języka Java 100  
wyświetlenie położenia dokumentu 37

## Z

zalogowanie się  
API 71  
dane wyjściowe ASCII oddzielone separatorami 137  
opis funkcji 6  
parametry 71  
przykładowe wywołanie funkcji 72  
załączniki 29, 30  
zmiana hasła 126  
API 63  
opis funkcji 6  
parametry 63  
przykładowe wywołanie funkcji 64  
zmiennne środowiskowe, funkcje API języka Java 92









Numer Programu: 5722-RD1

Printed in the USA

SC85-0120-03

