



System i

IBM Content Manager OnDemand for i5/OS  
Common Server ODWEK  
Podręcznik instalacji i konfiguracji

*Wersja 6 wydanie 1*

SC85-0120-04







System i

IBM Content Manager OnDemand for i5/OS  
Common Server ODWEK  
Podręcznik instalacji i konfiguracji

*Wersja 6 wydanie 1*

SC85-0120-04

**Uwaga**

Przed skorzystaniem z tych informacji oraz z produktu, którego dotyczą, należy przeczytać informacje zawarte w sekcji “Uwagi” na stronie 161.

To wydanie dotyczy oprogramowania IBM Content Manager OnDemand for i5/OS Common Server w wersji 6 wydanie 1 oraz wszystkich kolejnych wydań i modyfikacji, chyba że w nowych wydaniach określono inaczej. To wydanie zastępuje pozycję o numerze SC85-0120-03.

© Copyright International Business Machines Corporation 2001, 2008. Wszelkie prawa zastrzeżone.

# Spis treści

## Informacje o podręczniku dotyczącym IBM OnDemand for i5/OS Common Server Web Enablement Kit . . . . . 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
O czym należy wiedzieć . . . . .	vi
Centrum informacyjne OnDemand . . . . .	vii
Informacje o ułatwieniach dostępu produktu OnDemand . . . . .	vii
System i Navigator . . . . .	vii
Wysyłanie komentarzy . . . . .	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 . . . . .	12
Instalowanie w systemie i5/OS . . . . .	12
Dalsze kroki . . . . .	13
Określanie pliku ARSWWW.INI . . . . .	13
[@SRV@_DEFAULT] . . . . .	14
[@SRV@_server]. . . . .	14
[CONFIGURATION]. . . . .	15
[SECURITY] . . . . .	22
[AFP2HTML] . . . . .	23
[AFP2PDF]. . . . .	24
[MIMETYPES] . . . . .	26
[ATTACHMENT IMAGES]. . . . .	30
[NO HTML] . . . . .	31
[DEFAULT BROWSER]. . . . .	32
[przeładowarka] . . . . .	39
[DEBUG] . . . . .	40

Przykład pliku ARSWWW.INI . . . . .	40
Dalsze kroki . . . . .	43

## Rozdział 4. Konfigurowanie przykładowych aplikacji . . . . . 45

LOGON.HTM . . . . .	46
CREDIT.HTM . . . . .	46
TEMPLATE.HTM . . . . .	47
Dalsze kroki . . . . .	48

## Rozdział 5. Instalowanie przeglądarek WWW . . . . . 49

Przegląd. . . . .	49
Wymagania. . . . .	50
Instalowanie . . . . .	51
Przeładowarka AFP Web viewer . . . . .	52
Dystrybucja plików zdefiniowanych przez użytkownika . . . . .	52
Instalowanie plików przeglądarki AFP Web Viewer . . . . .	53
Dodawanie podkatalogów . . . . .	54
Przechowywanie plików zdefiniowanych przez użytkownika . . . . .	54
Konfigurowanie plików czcionek . . . . .	55
Tworzenie pliku instalacyjnego przeglądarki AFP Web Viewer . . . . .	55
Instalowanie przeglądarki AFP Web Viewer na stacji roboczej użytkownika. . . . .	56
Odwzorowanie czcionek AFP . . . . .	56
Wyświetlanie raportów AFP. . . . .	56
Wyświetlanie nakładek . . . . .	57
Przeładowarka Image Web viewer . . . . .	58
Przeładowarka danych dla drukarki wierszowej w języku Java . . . . .	58
Dalsze kroki . . . . .	63

## Rozdział 6. Weryfikowanie instalacji . . . . . 65

Weryfikowanie programu CGI . . . . .	65
Weryfikowanie serwletu . . . . .	66
Rozwiązywanie problemów . . . . .	66
Dalsze kroki . . . . .	67

## Dodatek A. Funkcje API CGI. . . . . 69

Dodanie adnotacji . . . . .	70
Zmiana hasła . . . . .	73
Lista trafień dokumentów . . . . .	75
Wylogowanie się . . . . .	79
Zalogowanie się . . . . .	81
Drukowanie dokumentu (serwer) . . . . .	83
Wczytanie dokumentu . . . . .	87
Kryteria wyszukiwania . . . . .	91
Aktualizowanie dokumentu . . . . .	93
Podgląd adnotacji . . . . .	95

## Dodatek B. Odniesienie serwletu w języku Java. . . . . 97

**Dodatek C. Odniesie funkcji API języka  
Java . . . . . 99**

**Dodatek D. Podręcznik  
programowania funkcji API języka  
Java . . . . . 101**

Architektura klient/serwer . . . . .	101
Tworzenie pakietów w środowisku języka Java . . . . .	101
Wskazówki dotyczące programowania . . . . .	102
Konfigurowanie parametrów systemowych . . . . .	103
Informacje diagnostyczne i związane ze śledzeniem . . . . .	104
Śledzenie . . . . .	104
Obsługa wyjątków . . . . .	105
Stałe . . . . .	106
Uruchamianie aplikacji ODWEK . . . . .	106
Nawiązywanie połączenia z serwerem OnDemand . . . . .	106
Nawiązywanie połączenia . . . . .	106
Ustawianie i uzyskiwanie haseł . . . . .	107
Praca z serwerem OnDemand . . . . .	107
Połączenie z portem innym niż domyślny za pomocą funkcji API języka Java . . . . .	109
Wyświetlanie grup aplikacji w folderze . . . . .	110
Wyszukiwanie w folderze . . . . .	111
Wyszukiwanie w folderze za pomocą łańcucha SQL . . . . .	115
Anulowanie wyszukiwania . . . . .	117
Wyświetlanie kryteriów wyszukiwania . . . . .	120
Wyświetlanie folderów i informacji o nich . . . . .	123
Wyświetlanie listy dokumentów . . . . .	124
Wczytywanie dokumentu . . . . .	126
Drukowanie dokumentu. . . . .	129
Wyświetlanie informacji o uwagach . . . . .	131
Dodawanie uwagi . . . . .	133
Aktualizowanie dokumentu. . . . .	135
Zmiana hasła . . . . .	137

**Dodatek E. Transformowanie z formatu  
AFP w HTML. . . . . 141**

Format pliku AFP2HTML.INI . . . . .	141
Opcje usługi AFP2WEB Transform . . . . .	142
Wyświetlanie przekształconych dokumentów . . . . .	143

**Dodatek F. Transformowanie z formatu  
AFP w PDF . . . . . 145**

Określanie pliku AFP2PDF.INI . . . . .	145
Wyświetlanie przekształconych dokumentów . . . . .	146

**Dodatek G. Pliki konfiguracyjne  
serwera HTTP . . . . . 147**

Serwer HTTP Apache . . . . .	147
WebSphere Application Server . . . . .	148

**Dodatek H. Brak danych wyjściowych  
HTML. . . . . 149**

Dane wyjściowe ASCII oddzielone separatorami . . . . .	149
Zalogowanie się . . . . .	149
Uwagi . . . . .	150
Kryteria wyszukiwania . . . . .	150
Uwagi . . . . .	150
Lista trafień dokumentów . . . . .	151
Uwagi . . . . .	151
Podgląd adnotacji . . . . .	152
Komunikat o błędzie. . . . .	152
Uwagi . . . . .	152

**Dodatek I. Obsługa języków  
narodowych . . . . . 153**

**Dodatek J. Narzędzia do określania  
problemu . . . . . 155**

**Dodatek K. Obsługa wielu języków w  
programie CGI za pomocą serwera  
Apache HTTP . . . . . 157**

Wymagania wstępne dotyczące oprogramowania . . . . .	157
Implementacja . . . . .	157

**Uwagi. . . . . 161**

Znaki towarowe i znaki usług . . . . .	162
--	-----

**Indeks . . . . . 165**

---

# Informacje o podręczniku dotyczącym IBM OnDemand for i5/OS Common Server Web Enablement Kit

Ten podręcznik zawiera informacje przydatne podczas planowania, instalowania, konfigurowania i używania aplikacji IBM Content Manager OnDemand for i5/OS w wersji 6 wydanie 1 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

W tym podręczniku znajdują się informacje pomocne podczas instalowania i konfigurowania aplikacji ODWEK oraz tworzenia planów dotyczących uzyskiwania przez użytkowników dostępu do danych z poziomu systemu IBM Content Manager OnDemand for i5/OS 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 45
- Rozdział 5, “Instalowanie przeglądarek WWW”, na stronie 49
- Dodatek A, “Funkcje API CGI”, na stronie 69
- Dodatek B, “Odniesienie serwletu w języku Java”, na stronie 97
- Dodatek C, “Odniesie funkcji API języka Java”, na stronie 99
- Dodatek D, “Podręcznik programowania funkcji API języka Java”, na stronie 101
- Dodatek E, “Transformowanie z formatu AFP w HTML”, na stronie 141
- Dodatek F, “Transformowanie z formatu AFP w PDF”, na stronie 145
- Dodatek G, “Pliki konfiguracyjne serwera HTTP”, na stronie 147
- Dodatek H, “Brak danych wyjściowych HTML”, na stronie 149
- Dodatek I, “Obsługa języków narodowych”, na stronie 153
- Dodatek J, “Narzędzia do określania problemu”, na stronie 155
- Dodatek K, “Obsługa wielu języków w programie CGI za pomocą serwera Apache HTTP”, na stronie 157

---

## Wymagania wstępne i informacje pokrewne

Centrum informacyjne systemu IBM i5/OS należy traktować jako punkt wyjścia przy wyszukiwaniu informacji technicznych na temat systemów System i5.

Dostęp do Centrum informacyjnego można uzyskać na dwa sposoby:

- z serwisu WWW pod adresem <http://www.ibm.com/systems/i/infocenter/>,
- z płyty CD-ROM dołączonych do zamówionego systemu i5/OS:  
*Centrum informacyjne System i5 SK3T-5495-07.*

Centrum informacyjne IBM i5/OS zawiera:

- nowe oraz zaktualizowane informacje dotyczące instalowania i aktualizowania systemu i5/OS, migracji danych, usług i rozwiązywania problemów, dostępności, integracji z serwerem System i, połączenia z serwerem System i, bazy danych, systemu Linux, oprogramowania WebSphere, języka Java, komend CL oraz systemowych interfejsów API, a także podręczniki;
- doradców oraz inne interaktywne narzędzia, które pomagają użytkownikom przy rozwiązywaniu problemów i konfigurowaniu oprogramowania i5/OS.

## Inne informacje dostępne w sieci WWW

Więcej informacji na temat systemu System i5 można znaleźć w sieci WWW. Informacje ogólne na temat systemu zawiera strona główna System i5, która znajduje się pod adresem: <http://www.ibm.com/systems/i/>.

W celu uzyskania dostępu do warsztatów dotyczących zaawansowanych funkcji systemu System i5, należy skorzystać z serwisu Technical Studio pod adresem: <http://www.redbooks.ibm.com/tstudio/>.

Użytkownicy z całego świata mogą przez Internet przeczytać o najnowszych poprawkach PTF dla systemu System i5, a także wybrać je, zamówić i odebrać. Internetowe poprawki PTF (do pobrania) oraz informacje PSP dotyczące systemu System i5 dostępne są pod następującym adresem: <http://as400service.ibm.com>.

Dokumentacja produktu została przeniesiona ze strony biblioteki na stronę wsparcia technicznego w serwisie WWW oprogramowania CM OnDemand for System i5. Pełna lista dostępnej dokumentacji oprogramowania OnDemand for System i5 znajduje się pod adresem <http://www.ibm.com/software/data/ondemand/400/support.html>. Znajduje się ona w lewej kolumnie, w sekcji "Self help", pod nagłówkiem "Learn".

## O czym należy wiedzieć

Dokumentacja aplikacji ODWEK zakłada, że użytkownik zna zasady funkcjonowania Internetu, serwerów i przeglądarek WWW, protokołów Transmission Control Protocol/Internet Protocol (TCP/IP) oraz serwera OnDemand. Ten podręcznik zakłada ponadto, że użytkownik zna podstawy języka Hypertext Markup Language (HTML), interfejsu Common Gateway Interface (CGI) i programowania w języku Java, potrafi tworzyć treści stron WWW, konfigurować i obsługiwać serwer HTTP, serwer WWW obsługujący język Java i serwer aplikacji Java oraz administrować serwerem OnDemand.

Jeśli planuje się korzystanie z przeglądarki Java AFP2HTML Viewer, należy uzyskać usługę AFP2WEB Transform z IBM, a następnie zainstalować i skonfigurować ją na serwerze, na którym rezyduje aplikacja ODWEK. Więcej informacji na temat usługi AFP2WEB Transform można uzyskać u przedstawiciela IBM. Należy również udostępnić opcje konfiguracji dla dokumentów i zasobów Advanced Function Presentation (AFP), które mają być przetwarzane przy użyciu usługi AFP2WEB Transform. Więcej informacji na temat pliku konfiguracyjnego zawiera Dodatek E, "Transformowanie z formatu AFP w HTML", na stronie 141.

Jeśli planuje się przekształcenie dokumentów AFP wczytywanych z serwera OnDemand w dokumenty PDF, które można wyświetlać w przeglądarce Adobe Acrobat, należy uzyskać usługę AFP2PDF Transform z IBM, a następnie zainstalować i skonfigurować ją na serwerze WWW. Więcej informacji na temat usługi AFP2PDF Transform można uzyskać u przedstawiciela IBM. Należy również udostępnić opcje konfiguracji dla dokumentów i zasobów AFP, które mają być przetwarzane przy użyciu usługi AFP2PDF Transform. Więcej informacji na temat pliku konfiguracyjnego zawiera Dodatek F, "Transformowanie z formatu AFP w PDF", na stronie 145.



## Centrum informacyjne OnDemand

Oprócz Centrum informacyjnego i5/OS warto odwiedzić również Centrum informacyjne OnDemand, które jest poświęcone oprogramowaniu CM OnDemand. Centrum informacyjne OnDemand zapewnia szybki, scentralizowany dostęp elektroniczny do informacji o produkcie. Stanowi ono zorientowane zadaniowo repozytorium dokumentacji, które umożliwia wyszukiwanie komend, kodów błędów i wszelkich innych tematów w całej bibliotece produktu. Użytkownik może tworzyć zakładki do interesujących i często przeglądanych stron, aby uprościć wyszukiwanie informacji w przyszłości.

Aby uzyskać dostęp do Centrum informacyjnego OnDemand, należy przejść do serwisu WWW pod adresem <http://publib.boulder.ibm.com/infocenter/cmod/v8r4m0/index.jsp>.

## Informacje o ułatwieniach dostępu produktu OnDemand

Kompletny zestaw informacji dotyczących funkcji ułatwień dostępu oferowanych przez ten produkt znajduje się w publikacji *IBM Content Manager OnDemand for i5/OS Common Server - podręcznik administrowania*.

## System i Navigator

Oprogramowanie IBM System i Navigator to rozbudowany interfejs graficzny, służący do zarządzania serwerami System i5. Funkcjonalność oprogramowania System i Navigator umożliwia nawigację po systemie, konfigurowanie go, planowanie jego możliwości oraz korzystanie z pomocy elektronicznej, która przeprowadzi użytkownika przez wykonywane zadania. Oprogramowanie System i Navigator pozwala na łatwiejszą i bardziej wydajną obsługę serwera oraz administrowanie nim, a ponadto stanowi jedyny interfejs użytkownika do wszystkich nowych, zaawansowanych funkcji systemu i5/OS. Program ten obejmuje także Centrum Zarządzania, które służy do zarządzania wieloma serwerami z systemu centralnego.

Więcej informacji na temat oprogramowania System i Navigator można znaleźć w Centrum informacyjnym IBM i5/OS, a także w serwisie WWW pod adresem: <http://www.ibm.com/servers/eserver/series/navigator/>.

---

## Wysyłanie komentarzy

Komentarze przesyłane firmie IBM pozwalają na podnoszenie jakości udostępnianych informacji. Prosimy o przesyłanie komentarzy dotyczących tej publikacji lub dowolnego elementu dokumentacji oprogramowania OnDemand. W celu wysłania komentarza należy odwiedzić stronę IBM Data Management Online Reader's Comment Form (RCF) pod adresem [www.ibm.com/software/data/rcf](http://www.ibm.com/software/data/rcf).

Prosimy o podanie nazwy i wersji produktu oraz tytułu podręcznika. Jeśli komentarz dotyczy konkretnego tekstu, prosimy o podanie miejsca w tekście (na przykład tytułu rozdziału i sekcji, numeru tabeli, numeru strony lub tytułu tematu pomocy).



---

## Podsumowanie zmian

To wydanie publikacji *IBM Content Manager OnDemand for i5/OS Common Server ODWEK: podręcznik instalowania i konfigurowania* zawiera nowe informacje techniczne. W niektórych przypadkach wprowadzone zmiany nie zostały oznaczone symbolami zmian. Poważne zmiany, na które należy zwrócić uwagę:

- Udostępniono nowy interfejs API wsadowych zadań administracyjnych (ARSXML), który wykonuje wiele funkcji administracyjnych (takich jak dodawanie użytkowników, zmiana uprawnień lub eksport/import definicji) w środowisku wsadowym. Do tej pory taką funkcję oferował wyłącznie interaktywny klient administracyjny OnDemand.
- Do produktu w wersji 6 wydanie 1 dodano obsługę oprogramowania Tivoli Storage Manager (TSM).
- Można teraz wykorzystywać niezależne pule pamięci dyskowej (IASP) jako opcję dla bibliotek instancji OnDemand oraz dla nośników archiwalnych.
- System plików podłączany do NFS może teraz działać jako pula dyskowa, w charakterze zarówno nośników podstawowych, jak i nośników składowania. Szczegółowe instrukcje konfigurowania tej funkcji znajdują się w serwisie WWW OnDemand Support pod adresem <http://www.ibm.com/software/data/ondemand/400/support.html> (w polu wyszukiwania należy wprowadzić hasło 'NFS disk pool').
- Struktura IFS pul dyskowych dla nowo archiwizowanych danych otrzymała dodatkowy poziom katalogów. Przy umieszczaniu obiektów w puli dyskowej tworzony jest podkatalog określający rok, miesiąc i dzień (RRRRMMDD) dodania obiektu do puli, i w tym podkatalogu umieszczany jest obiekt. Obiekty istniejące już w puli dyskowej pozostaną w pierwotnym położeniu i nie zostaną włączone do nowej struktury. Z czasem, gdy obiekty wygasną lub zostaną przeniesione na inny poziom, pula będzie zawierać wyłącznie obiekty zapisane w ramach nowej struktury. Nowa struktura pozwoli na tworzenie kopii zapasowych o większej granulacji. Będzie można tworzyć kopie zapasowe z konkretnego roku, miesiąca w danym roku lub nawet z wybranego dnia.
- Nowe opcje protokołowania i śledzenia zapewniają bardziej zaawansowane możliwości określania problemów. Parametry śledzenia można ustawiać przy użyciu klienta administracyjnego OnDemand.
- Do programu licencjonowanego OnDemand w wersji 6 wydanie 1 dodano dwie nowe komendy. Komenda Change Policy Level Date (CHGPLDOND) pozwala na elastyczną zmianę daty przeniesienia danych archiwalnych z jednego nośnika archiwalnego na inny. Komenda Migrate Media (MGRMEDRDAR) umożliwia przenoszenie danych środowiska Spool File Archive z jednego typu nośnika na inny.
- Do interfejsu API ARSLOAD dodano nowe parametry.
- W wersji 6 wydanie 1 oprogramowania usunięto opcjonalne parametry COPIES i PAGERANGE komendy Raport drukowania z OnDemand (Print Report from OnDemand - PRTRPTOND). Należy usunąć te parametry ze wszystkich programów w języku CL i wpisów programu do planowania zadań, gdzie są obecnie wykorzystywane.
- W wersji 6 wydanie 1 oprogramowania usunięto opcjonalny parametr VALIDATE komendy Uruchom zarządzanie pamięcią dyskową (Start Disk Storage Management - STRDSMOND). Należy usunąć ten parametr ze wszystkich programów w języku CL i wpisów programu do planowania zadań, gdzie są obecnie wykorzystywane.
- Znacznie ulepszono przykładowy kod programu postprocesora i jego dokumentację.
- Z licencjonowanego programu OnDemand w wersji 6 wydanie 1 usunięto obsługę funkcji Spool File Archive (archiwizacja zbiorów buforowych), Object Archive (archiwizacja obiektów), Record Archive (archiwizacja rekordów), AnyStore oraz klienta/serwera Spool File Archive (opcje produktu nr 1, 2, 3, 4 i 5). Zgodnie z ogłoszeniem IBM 206-030 z 14

lutego 2006 r. oraz dokumentami "Przeczytaj zanim zaczniesz" dołączonymi do oprogramowania OnDemand w wersjach 5.3 oraz 5.4 ostatnią wersją, w której zapewniono obsługę i wsparcie funkcji Spool File Archive, AnyStore, Record Archive i Object Archive, jest wersja 5.4. Poczynając od wersji 5.3 do licencjonowanego programu OnDemand dołączany jest program narzędziowy do migracji ze środowiska Spool File Archive, który umożliwia migrację definicji raportów oraz indeksów z wcześniejszego środowiska Spool File Archive do środowiska Common Server. Zachęcamy obecnych klientów korzystających ze środowiska Spool File Archive do zapoznania się z oprogramowaniem Common Server i zaplanowania migracji na tę platformę. Nowe instalacje oraz aplikacje OnDemand powinny być wdrożone za pomocą środowiska Common Server.

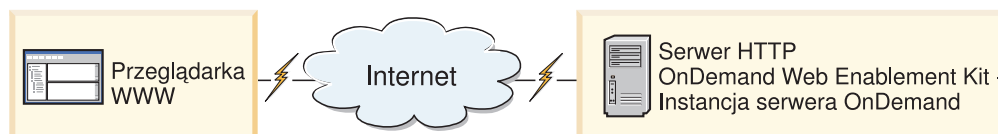
- Ponieważ wsparcie dla funkcji Spool File Archive nie jest już świadczone, z licencjonowanego programu OnDemand w wersji 6.1 i następnych usunięto narzędzie migracji ze środowiska Spool File Archive do Common Server (oraz powiązaną z nim dokumentację w dodatkach A i B podręcznika planowania i instalowania Common Server). Informacje na temat migracji znajdują się w dokumentacji wersji 5.4.

---

## 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

Aplikacja ODWEK może wyszukiwać i odtwarzać dokumenty z serwerów OnDemand, na których uruchomiony jest system IBM Content Manager OnDemand for i5/OS Common Server, IBM Content Manager OnDemand for Multiplatforms lub IBM Content Manager OnDemand for z/OS.

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. Funkcja API języka Java wymaga środowiska Java w wersji 1.4 lub nowszej.
- Przeglądarka IBM OnDemand Advanced Function Presentation (AFP) Web Viewer. Przeglądarka AFP Web Viewer umożliwia użytkownikom wyszukiwanie dokumentów AFP, ich wczytywanie, wyświetlanie, drukowanie oraz 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, ich wczytywanie, wyświetlanie i drukowanie oraz nawigację w tych dokumentach przy użyciu przeglądarki WWW.

- Aplikacja ODWEK obsługuje teraz dwie różne wersje apletu 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.

**Ważne:** 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

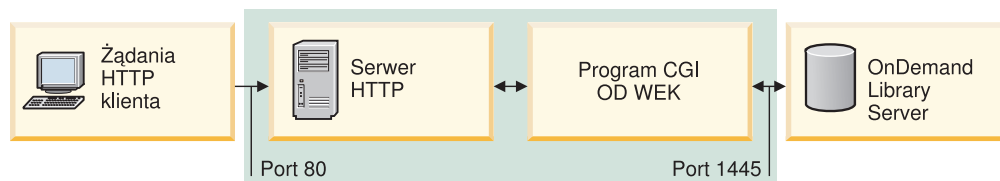
*Instancja* aplikacji ODWEK jest kodem tej aplikacji, który uzyskuje dostęp do danych na serwerze OnDemand. Instancja określa, co można zrobić z danymi, i zarządza zasobami systemowymi przypisanymi do tych danych. Każda instancja jest całym środowiskiem. Ma ona własny plik ASWWW.INI oraz interfejs programistyczny 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.

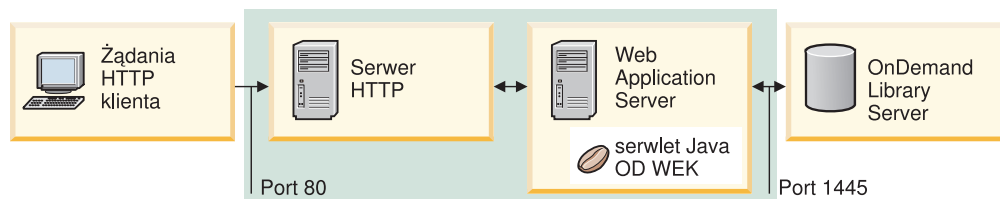
Najczęściej stosowaną implementacją pakietu ODWEK jest jedna instancja w systemie. Konfiguracja z pojedynczą instancją jest typowa dla przetwarzania programistycznego lub autonomicznego środowiska produkcyjnego, w których jedna instancja serwera aplikacji działa niezależnie od innych aplikacji.

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



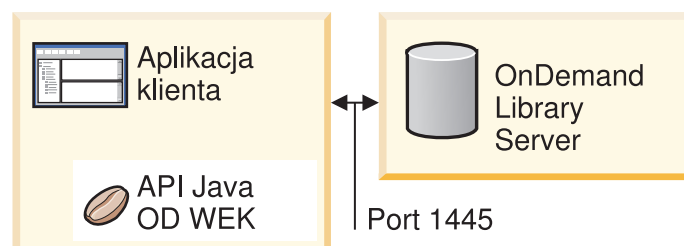
Rysunek 2. Pojedyncza instancja używająca interfejsu CGI

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



Rysunek 3. Pojedyncza instancja używająca interfejsu serwletu w języku Java

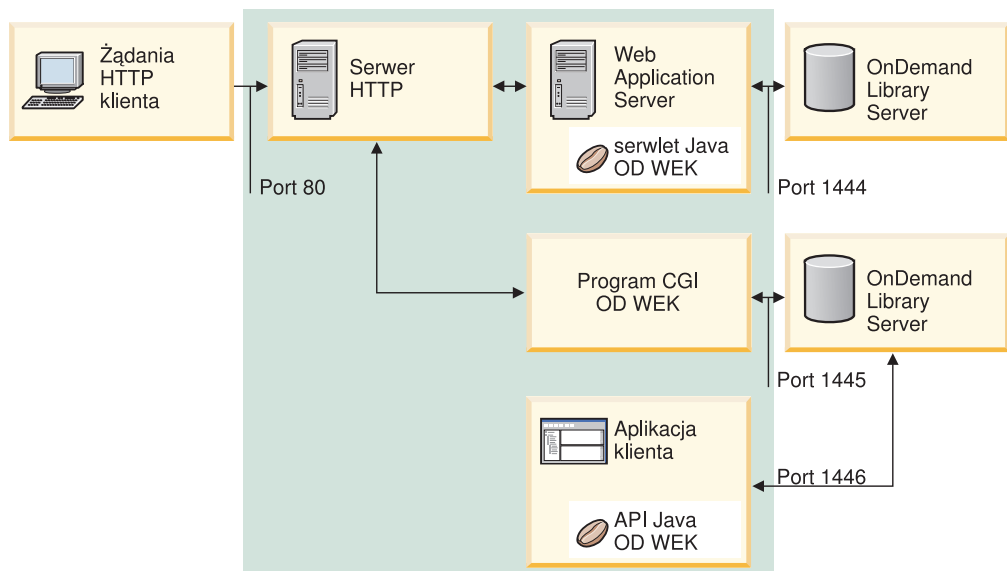
Na Rys. 4 przedstawiono przykład pojedynczej instancji używającej interfejsu API w języku Java.



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.

**Ważne:** 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:** aplikacja 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 69.

## 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.company.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.company.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.

**Ważne:** 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 35.

---

## Rozdział 2. Lista kontrolna instalacji

Aby skonfigurować środowisko OnDemand Web Enablement Kit, zazwyczaj należy wykonać następujące czynności:

1. Skontaktuj się z Centrum wsparcia IBM w celu uzyskania najnowszych poprawek PTF dla serwera OnDemand. Lista bieżących poprawek PTF znajduje się w informacyjnym raporcie APAR nr II14283.
2. Skontaktuj się z Centrum wsparcia IBM w celu uzyskania najnowszych poprawek PTF dla grupy serwerów HTTP systemu System i5. Numer produktu serwera HTTP to 5722-DG1. Aby wyświetlić listę najnowszych poprawek PTF dla 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 w celu uzyskania najnowszych poprawek 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>. Jest on dostępny w sekcji "Learn" razem z innymi dokumentami dotyczącymi wersji 6 wydanie 1 wydanie 4. Przed rozpoczęciem instalowania wydrukuj cały plik i przeczytaj go.
5. Sprawdź wymagania wstępne ODWEK. Patrz Rozdział 3, "Instalowanie i konfigurowanie serwera HTTP", na stronie 11
6. Zainstaluj oprogramowanie OnDemand na serwerze System i5. 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 147 zawiera przykład zbioru konfiguracyjnego serwera HTTP.
9. W razie konieczności zaktualizuj listy autoryzacji QONDADM i QRDARS400. Patrz rozdział 3, "Inne wymagania" na stronie 12
10. Skonfiguruj przeglądarkę stacji roboczej. W tym celu wykonaj następujące czynności:
  - a. Pobierz i zainstaluj odpowiednie wtyczki przeglądarki. Patrz Rozdział 5, "Instalowanie przeglądarek WWW", na stronie 49
  - b. Aby móc korzystać z najnowszej wersji apletu wyświetlania danych drukarki, pobierz moduł środowiska wykonawczego Java ze strony <http://www.java.com> i zainstaluj go.
  - c. Upewnij się, że przeglądarka akceptuje wszystkie informacje cookie. Wybierz menu **Narzędzia > Opcje internetowe** (Tools > Internet options), a następnie kartę Prywatność (Privacy).
  - d. Upewnij się, że dla przeglądarki Internet Explorer wybrana jest opcja UTF-8. Wybierz menu **Narzędzia > Opcje internetowe** (Tools > Internet options), a następnie kartę Zaawansowane (Advanced). Upewnij się, że zaznaczona jest opcja **Zawsze wysyłaj adresy URL jako UTF-8** (Always send URLs and UTF-8).
  - e. Upewnij się, że środowisko wykonawcze programów Java jest włączone. Wybierz menu **Narzędzia > Opcje internetowe** (Tools > Internet options), a następnie kartę Zaawansowane (Advanced). Przejdź do sekcji Java (Sun). Upewnij się, że

| zaznaczona jest opcja **Use Java n vx.y.x for <applet> (requires restart)**. 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 na systemie System i5, na którym uruchomiona jest aktualna wersja serwera HTTP IBM. Poza tym jeśli planowane jest użycie serwletu języka Java, należy się upewnić, że aktualna wersja serwera aplikacji WWW System i5 (WebSphere) jest poprawnie zainstalowana i skonfigurowana.

Aplikacja ODWEK może wyszukiwać i odtwarzać dokumenty z serwerów OnDemand, na których uruchomione jest oprogramowanie IBM Content Manager OnDemand dla systemu i5/OS w wersji 6 wydanie 1 aplikacji Common Server wydanie 4.

---

### Wymagania instalacyjne

Pakiet ODWEK wymaga:

- programu i5/OS Navigator w wersji 6.1,
- menedżera Tivoli Storage Manager (TSM) Extended Edition 5.4,
- systemu Windows XP SP2 lub nowszego bądź Windows Vista,
- jednej z następujących przeglądarek:
  - Mozilla Firefox 2.0 lub nowszej,
  - Netscape Navigator 7.1 lub nowszej,
  - Internet Explorer 6.0.1 lub 7.0.
- środowiska wykonawczego programów Java dla apletu w wersji 1.6.0 lub nowszej,
- bieżącej wersji serwera IBM Apache HTTP (5722-DG1). Ponadto:
  - Jeśli planowane jest użycie serwletu języka Java, należy się upewnić, że aktualna wersja serwera aplikacji WWW System i5 (WebSphere) w wersji 6.0.1 lub nowszej jest poprawnie zainstalowana i skonfigurowana. Serwery muszą być uruchomione w systemie i5/OS w wersji 6 wydaniu 1 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 należy zakończyć i zrestartować wszystkie uruchomione serwery Apache, na których działa ten program.
- 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 18),
- Więcej informacji na temat plików konfiguracyjnych można znaleźć w sekcji Dodatek G, “Pliki konfiguracyjne serwera HTTP”, na stronie 147.



---

## 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 17.

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 Microsoft Internet Explorer wybierz menu **Narzędzia > Opcje internetowe** (Tools > Internet options), a następnie wybierz kartę Zaawansowane (Advanced). W sekcji Przeglądanie (Browsing) ustaw opcję **Zawsze wysyłaj adresy URL jako UTF-8** (Always send URLs as 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 z formatu AFP w HTML”, na stronie 141.

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 z formatu AFP w PDF”, na stronie 145. Aby wyświetlić przekształcone dokumenty, należy pobrać i zainstalować moduł dodatkowy Adobe Acrobat viewer dla przeglądarek WWW używanych w organizacji.

---

## Instalowanie w systemie i5/OS

Konfiguracja aplikacji ODWEK zazwyczaj wymaga wykonania następujących czynności:

1. Aby zainstalować aplikację ODWEK, postępuj zgodnie z instrukcjami znajdującymi się w podręczniku o nazwie *IBM Content Manager OnDemand for i5/OS Common Server: podręcznik planowania i instalowania* (SC85-0121). Numer programu licencjonowanego to 5722RD1, a numer opcji - 11.

**Ważne:** 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). W oknie Instalowanie programów licencjonowanych (Install licensed programs) wprowadź wartość 1, aby dodać opcję, wpisz



5722RD1 dla programu licencjonowanego i wartość 11 dla opcji produktu, lub przewin listę programów licencjonowanych i opcji produktu i odszukaj aplikację ODWEK, a następnie przed jej nazwą wpisz wartość 1. *W przypadku zainstalowania serwera OnDemand w inny sposób podczas próby jego użycia mogą wystąpić błędy.*

2. IBM zaleca zamówienie, załadowanie i zastosowanie po udanym zainstalowaniu programu licencjonowanego wszystkich poprawek PTF dostępnych dla serwera OnDemand. Pełna lista poprawek PTF do oprogramowania OnDemand w wersji 6 wydanie 1 wydanie 4 znajduje się w informacyjnym raporcie APAR II14283. APAR informacyjny można zamówić elektronicznie, korzystając z komendy SNDPTFORD. Jako numer PTF należy podać II14283. Należy dokładnie przeczytać list przewodni do poprawki PTF i wykonać wszelkie instrukcje specjalne.
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 się upewnić, że w systemie System i5 zainstalowana i skonfigurowana jest aktualna wersja serwera HTTP IBM. Serwer HTTP należy skonfigurować. Dodatek G, “Pliki konfiguracyjne serwera HTTP”, na stronie 147 zawiera przykład pliku konfiguracyjnego serwera HTTP.

Jeśli planowane jest użycie serwletu języka Java, należy się upewnić, że aktualna wersja serwera aplikacji WWW System i5 (WebSphere) w wersji 6.0.1 lub nowszej jest poprawnie zainstalowana i skonfigurowana. Należy skonfigurować serwer WebSphere. Instrukcje można znaleźć w Centrum dokumentacji serwera aplikacji IBM WebSphere, które dostępne jest 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 opcjonalnym skonfigurowaniu serwera WebSphere można skonfigurować plik inicjowania aplikacji 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@_QUSROND]
HOST=Sxxxxxx.mynetwork.com
PORT=1450
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 40.

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@\_server]**

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 applety 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, applety będą próbować przechowywać dokumenty w katalogu roboczym języka Java.
3. Jeśli podany katalog nie istnieje, applety 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 applety 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 applety 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/edms1od/(.*)$ /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" na stronie 17. Domyślnie aplikacja ODWEK przechowuje dokumenty w katalogu /QIBM/UserData/OnDemand/WWW/CACHE.

**Ważne:** 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” na stronie 18).

**Ważne:** 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).

**Ważne:** informacje o tym, jak włączyć pamięć podręczną dla dokumentów, znajdują się w sekcji “CACHEDOCS” na stronie 17.

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]  
CACHESIZE=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 serwlet, katalogiem wykonawczym jest katalog, w którym znajduje się ten serwlet; 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
```

**Ważne:** 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.

**Ważne:** 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=QUSROND
```

### UPDATETIMESTAMP

Określa, że aplikacja ODWEK powinna aktualizować datę i godzinę po każdej transakcji z udziałem serwera. Jeśli wartość limitu czasu nieaktywności nie została ustawiona, zostanie ona porównana z czasem, jaki upłynął od ostatniej transakcji użytkownika. Ma to na celu

wyeliminowanie dodatkowych i zbędnych operacji wpisywania się. Aby określić, że aplikacja ODWEK ma aktualizować datę i godzinę po każdej transakcji, parametr UPDATETIMESTAMP należy ustawić na 1 (jeden).

**Ważne:** jeśli w wielu przeglądarkach aktywny jest ten sam identyfikator użytkownika, parametr UPDATETIMESTAMP może funkcjonować nieprawidłowo.

Parametr UPDATETIMESTAMP ma zasięg globalny i w sekcji SECURITY należy go określić wyłącznie jeden raz.

Parametr UPDATETIMESTAMP jest opcjonalny. Jeśli nie jest określony lub jest ustawiony na 0 (zero), a parametr REPORTSERVERTIMEOUT nie jest ustawiony na 1 (jeden), aplikacja ODWEK nie przeprowadzi aktualizacji daty i godziny po każdej transakcji z udziałem serwera. Jeśli wartość limitu czasu nieaktywności została ustawiona, zostanie ona porównana z czasem, jaki upłynął od ostatniego wpisania się użytkownika, a nie jego ostatniej transakcji. W rezultacie mogą mieć miejsce zbędne dodatkowe operacje wpisywania się.

Więcej informacji dotyczących limitu czasu nieaktywności znajduje się w pomocy elektronicznej klienta administracyjnego. Parametry UPDATETIMESTAMP i REPORTSERVERTIMEOUT są do siebie podobne. Po ustawieniu wartości 1 (jeden) każdy z nich zaktualizuje datę i godzinę po każdej transakcji z udziałem serwera. Różnica widoczna jest po upłynięciu limitu czasu nieaktywności. Parametr REPORTSERVERTIMEOUT powoduje zakończenie sesji użytkownika i wygenerowanie raportu o błędzie. Parametr UPDATETIMESTAMP tworzy nową operację wpisywania się dla użytkownika i nie generuje raportu o błędzie. Jeśli żaden z powyższych parametrów nie ma ustawionej wartości 1 (jeden), data i godzina nie będą aktualizowane, a wartość limitu czasu nieaktywności będzie porównywana z czasem, jaki upłynął od ostatniego wpisania się.

Przykład:

```
[SECURITY]
UPDATETIMESTAMP=1
```

## [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 z formatu AFP w HTML", na stronie 141.
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 33. (Planując użycie funkcji API Retrieve Document (Wczytanie dokumentu), należy określić parametr \_afp=HTML. Szczegóły znajdują się w sekcji "Wczytanie dokumentu" na stronie 87.)
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 udostępni 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 z formatu AFP w HTML", na stronie 141. 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.

**Ważne:** 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/HTMLbin
```

## USEEXECUTABLE

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

**Ważne:** aplikacja ODWEK zainstalowana na systemie System i5 musi korzystać z 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 z formatu AFP w PDF", na stronie 145.
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 33. (Planując użycie funkcji `API Retrieve Document` (Wczytanie dokumentu), należy określić parametr `_afp=PDF`. Szczegóły znajdują się w sekcji "Wczytanie dokumentu" na stronie 87.)
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 z formatu AFP w PDF", na stronie 145 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.

**Ważne:** 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/PDFbin
```

## USEEXECUTABLE

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

**Ważne:** aplikacja ODWEK zainstalowana na systemie System i5 musi korzystać z 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 40) 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 na stronie 27 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 33. 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 34.

### 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 36.

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 `odlc_att.gif` (który znajduje się w katalogu określonym przez parametr `IMAGEDIR` w sekcji `CONFIGURATION`). Pliku `odlc_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 149 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 “[przeglądarka]” na stronie 39. (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 trzyznakowe 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.

**Ważne:** 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 API Aktualizacja dokumentu (Update Document), należy określić wartość parametru ADDFIELDSTODOCID=1.
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.

**Ważne:** 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.

**Ważne:** 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 26.

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

**Ważne:** 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 26.

**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=0
```

## 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.

**Ważne:** określając parametr EMAIL=NATIVE, należy sprawdzić, czy typ zawartości MIME określa odpowiednią przeglądarkę do ich wyświetlania. Szczegóły znajdują się w sekcji “[MIMETYPES]” na stronie 26.

**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 `server`, `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 39). 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 wyodrębnia i dekompresuje dokumenty zawierające dane dla drukarki wierszowej z serwera OnDemand.                  |

**Ważne:** określając parametr LINEVIEWING=NATIVE, należy sprawdzić, czy typ zawartości MIME określa odpowiednią przeglądarkę do ich wyświetlania. Szczegóły znajdują się w sekcji “[MIMETYPES]” na stronie 26.

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. Dla funkcji API Lista trafień dokumentów (Document Hit List) i Drukowanie dokumentu (Print Document) jest to wartość parametru `_max_hits`, jeśli został określony dla danej funkcji. Wartość parametru `_max_hits` przestania 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 58.

## ODApplet.jre.path.NN

Patrz sekcja “Przeglądarka danych dla drukarki wierszowej w języku Java” na stronie 58.

## ODApplet.jre.version

Patrz sekcja “Przeglądarka danych dla drukarki wierszowej w języku Java” na stronie 58.

## ODApplet.version

Patrz sekcja “Przeglądarka danych dla drukarki wierszowej w języku Java” na stronie 58.

## 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.

3. Podczas wybierania z listy trafień dokumentów, które mają być wydrukowane, należy pamiętać, że wydrukowany zostanie tylko pierwszy wybrany dokument (nawet jeśli wybranych zostało wiele dokumentów).

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 149) określa, czy w danych wyjściowych znajdzie się położenie dokumentu w pamięci masowej. Szczegóły zawiera sekcja “Lista trafień dokumentów” na stronie 151. Wartością domyślną jest 0 (zero); oznacza ona, że położenie w pamięci masowej nie będzie uwzględnione w danych wyjściowych.

**Ważne:** 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.

**Ważne:** 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
```

## [przełglądarka]

Można określić opcje dotyczące konkretnych przełglądarek 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 przełglądarki).

### Uwagi:

1. Poniższe parametry mają zasięg globalny i można je określić wyłącznie w sekcji DEFAULT BROWSER (po ich określeniu w dowolnej innej sekcji przełglądarki zostaną one zignorowane).
  - ODApplet.jre.path.IE
  - ODApplet.jre.path.NN
  - ODApplet.jre.version
  - ODApplet.version
2. Nagłówek sekcji musi zawierać łańcuch określający przełglądarkę, której mają dotyczyć podane opcje. ODWEK wyodrębnia tę wartość ze zmiennej środowiskowej HTTP\_USER\_AGENT, określając używaną przełglądarkę. Następnie szuka w pliku ARSWWW.INI sekcji dotyczącej przełglądarki, która odpowiada tej wartości. Jeśli nie zostanie znaleziona odpowiednia sekcja, ODWEK szuka w pliku ARSWWW.INI jednej z następujących sekcji:

```
[przełglądarka wersja(główna.poboczna)/platforma]
```

```
[przełglądarka wersja(główna.poboczna)]
```

```
[przełglądarka wersja(główna)]
```

```
[przełglądarka]
```

```
[DEFAULT BROWSER]
```

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

3. Jako wartość **przełglądarka** można określić IE lub Netscape.
4. Jako wartość **platforma** można określić WinNT lub Unix.

Sekcja dotycząca przełglądarki ma zasięg globalny dla niej. Dla każdej przełglądarki 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 przełglądarki domyślnej. Patrz sekcja “[DEFAULT BROWSER]” na stronie 32.

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.

**Ważne:** 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:

### TRACE

Umożliwia aplikacji ODWEK zapisywanie komunikatów i innych informacji o programie w pliku śledzenia (plik śledzenia nosi nazwę ARSWWW.TRACE).

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

Parametr opcjonalny. Aby określić poziom śledzenia, użyj jednej z następujących wartości:

- 0** No trace (Brak śledzenia)
- 1** Errors only (Tylko błędy)
- 2** Errors and Warnings (Błędy i ostrzeżenia)
- 3** Errors, Warnings, and Information (Błędy, ostrzeżenia i informacje)
- 4** All (Wszystko)

### TRACEDIR

Określa katalog, w którym aplikacja ODWEK zapisuje plik ARSWWW.TRACE, jeśli funkcja śledzenia została włączona za pomocą parametru TRACE.

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

Parametr opcjonalny. Domyślnie przy włączonej funkcji śledzenia aplikacja ODWEK zapisuje plik śledzenia w katalogu /QIBM/UserData/OnDemand/WWW/LOG.

Przykład:

```
[DEBUG]
;Trace=None=0, Error=1, Error+Warn=2, Err+Warn+Info=3, All=4
Trace=4
TraceDir=/QIBM/UserData/OnDemand/www/logs
```

## Przykład pliku ARSWWW.INI

Poniżej widoczna jest przykładowa konfiguracja domyślnej instancji QUSROND.

Wytluszczone elementy to wiersze konfiguracji, które należy zmodyfikować lub dodać.

Czasami obok wierszy znajdują się uwagi ujęte w nawiasach i zaznaczone tłustym drukiem i kursywą. Wiersze komentarza rozpoczyna średnik. Należy upewnić się, że katalogi określone w tym zbiorze istnieją. Jeśli któryś z katalogów nie istnieje, uruchomienie aplikacji 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 widocznych poniżej trzech wierszy należy usunąć średniki. Należy pamiętać o ponownym dodaniu średników po zakończeniu korzystania z opcji DEBUG).
[DEBUG]
;Trace=None=0, Error=1, Error+Warn=2, Err+Warn+Info=3, All=4
;Trace=4
;TraceDir=/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/HTMLbin
;ConfigFile=/QIBM/UserData/OnDemand/www/HTMLbin/afp2html.ini

;[afp2pdf]
;InstallDir=/QIBM/UserData/OnDemand/www/PDFbin
;ConfigFile=/QIBM/UserData/OnDemand/www/PDFbin/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/afp
LINE=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-wordpro
SAM=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
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 45.





---

## Rozdział 4. Konfigurowanie przykładowych aplikacji

Aby skonfigurować środowisko OnDemand Web Enablement Kit, zazwyczaj należy wykonać następujące czynności:

1. Skontaktuj się z Centrum wsparcia IBM w celu uzyskania najnowszych poprawek PTF dla serwera OnDemand. Lista bieżących poprawek PTF znajduje się w informacyjnym raporcie APAR nr II14283.
2. Skontaktuj się z Centrum wsparcia IBM w celu uzyskania najnowszych poprawek PTF dla grupy serwerów HTTP systemu System i5. Numer produktu serwera HTTP to 5722-DG1. Aby wyświetlić listę najnowszych poprawek PTF dla 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 w celu uzyskania najnowszych poprawek 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>. Jest on dostępny w sekcji "Learn" razem z innymi dokumentami dotyczącymi wersji 6 wydanie 1 wydanie 4. Przed rozpoczęciem instalowania wydrukuj cały plik i przeczytaj go.
5. Sprawdź wymagania wstępne ODWEK. Patrz Rozdział 3, "Instalowanie i konfigurowanie serwera HTTP", na stronie 11
6. Zainstaluj oprogramowanie OnDemand na serwerze System i5. 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 147 zawiera przykład zbioru konfiguracyjnego serwera HTTP.
9. W razie konieczności zaktualizuj listy autoryzacji QONDADM i QRDARS400. Patrz rozdział 3, "Inne wymagania" na stronie 12
10. Skonfiguruj przeglądarkę stacji roboczej. W tym celu wykonaj następujące czynności:
  - a. Pobierz i zainstaluj odpowiednie wtyczki przeglądarki. Patrz Rozdział 5, "Instalowanie przeglądarek WWW", na stronie 49
  - b. Aby móc korzystać z najnowszej wersji apletu wyświetlania danych drukarki, pobierz moduł środowiska wykonawczego Java ze strony <http://www.java.com> i zainstaluj go.
  - c. Upewnij się, że przeglądarka akceptuje wszystkie informacje cookie. Wybierz menu **Narzędzia > Opcje internetowe** (Tools > Internet options), a następnie kartę Prywatność (Privacy).
  - d. Upewnij się, że dla przeglądarki Internet Explorer wybrana jest opcja UTF-8. Wybierz menu **Narzędzia > Opcje internetowe** (Tools > Internet options), a następnie kartę Zaawansowane (Advanced). Upewnij się, że zaznaczona jest opcja **Zawsze wysyłaj adresy URL jako UTF-8** (Always send URLs and UTF-8).
  - e. Upewnij się, że środowisko wykonawcze programów Java jest włączone. Wybierz menu **Narzędzia > Opcje internetowe** (Tools > Internet options), a następnie kartę Zaawansowane (Advanced). Przejdź do sekcji Java (Sun). Upewnij się, że

zaznaczona jest opcja **Use Java n vx.y.x for <applet> (requires restart)**. Wersja wyświetlona w panelu odzwierciedla wersję zainstalowaną obecnie dla przeglądarki.

**Ważne:** jeśli do sterowania aplikacją ODWEK używana jest funkcja API języka Java, należy zapoznać się z informacjami w sekcji Dodatek D, "Podręcznik programowania funkcji API języka Java", na stronie 101 na temat konfiguracji środowiska systemowego i uruchamiania aplikacji ODWEK. Ten rozdział opisuje, w jaki sposób konfigurować przykładowe aplikacje udostępniane razem z aplikacją ODWEK i przeznaczone do programu CGI i serwletu Java.

- LOGON.HTM. Ta aplikacja obsługuje użytkowników, którzy mają dostęp do wielu folderów. Każdy użytkownik zdefiniowany jest według parametrów serwera biblioteki OnDemand. Po wpisaniu się użytkownika na serwerze, aplikacja ODWEK wyświetla listę folderów, które mogą być otworzone. Aplikacja "LOGON.HTM" zawiera instrukcje konfiguracji tej aplikacji.
- CREDIT.HTM. Ta aplikacja obsługuje codzienne użycie serwera OnDemand. Użytkownicy mają do dyspozycji kryteria wyszukiwania dla określonego folderu. Nazwa serwera OnDemand, identyfikator i hasło użytkownika, nazwa folderu i jego pola są kodowane w aplikacji. Informacje na temat konfiguracji aplikacji CREDIT.HTM można znaleźć w sekcji "CREDIT.HTM" na stronie 46.
- FCREDIT.HTM. Ta aplikacja jest wersją aplikacji CREDIT, która prezentuje użycie ramek HTML.

Po zmodyfikowaniu przykładowych aplikacji należy opublikować adres URL każdego pliku, aby użytkownicy mogli tworzyć odsyłacze do nich z tym adresem i uzyskiwać dostęp do serwera OnDemand. Każdy przykład wymaga innego poziomu konfiguracji. Poniżej znajdują się kompletne instrukcje dotyczące konfiguracji przykładowej aplikacji CREDIT.HTM. Instrukcje te należy potraktować jako pomoc podczas konfigurowania innych aplikacji, które mogą okazać się potrzebne.

**Ważne:** IBM zaleca, aby poza modyfikowaniem przykładowych aplikacji dostosować również plik TEMPLATE.HTM do potrzeb danej organizacji. Plik TEMPLATE.HTM zawiera treść zdefiniowaną przez użytkownika, z której korzysta aplikacja ODWEK w celu wyświetlania stron WWW. Ważne informacje dotyczące modyfikacji tego pliku znajdują się w sekcji "TEMPLATE.HTM" na stronie 47.

---

## LOGON.HTM

1. Skopiuj plik logon.htm z katalogu instalacyjnego do katalogu elementu głównego dokumentu serwera HTTP (na przykład: /www/HttpInstanceName/htdocs).
2. W przypadku programu CGI sprawdź, czy plik logon.htm zawiera następujące wiersze:

```
<h4>Please enter your logon information:</h4>
<FORM METHOD=POST ACTION="/arswww.cgi">
```
3. W przypadku serwletu sprawdź, czy plik logon.htm zawiera następujące wiersze:

```
<h4>Please enter your logon information:</h4>
<FORM METHOD=POST ACTION="/od/odservlet">
```

---

## CREDIT.HTM

Przykładową aplikację CREDIT.HTM należy skonfigurować poprzez utworzenie kopii pliku dla każdego folderu, do którego mają mieć dostęp użytkownicy. Nazwa pliku powinna być taka sama, jak nazwa folderu.

1. Edytuj plik CREDIT.HTM (domyślnie plik ten znajduje się w katalogu /QIBM/UserData/OnDemand/www/samples).
2. Zmień lub usuń obrazek w tle określony w instrukcji <body> (wiersz 11).

3. Jeśli chcesz, możesz usunąć kolor tła określony w instrukcji <body> (wiersz 11).
4. Zmień lub usuń obrazek produktu określony w instrukcji <img> (wiersz 12).
5. Zastąp nazwę folderu nazwą określoną w instrukcji <h1> (wiersz 15).
6. Zastąp tekst określony w instrukcjach <p> (wiersze od 17 do 25). Wprowadź ogólne instrukcje dla użytkownika.
7. Zastąp nazwę katalogu CGI-BIN określoną w instrukcji <FORM> (wiersz 29). W przypadku CGI zastąp ją nazwą "/scripts/arswww.cgi".
8. Zastąp wartość określoną w instrukcji <input> (wiersz 30). Jest to łańcuch zawierający pola widoku folderu oddzielone przecinkami.
9. Zastąp wartość określoną w instrukcji <input> (wiersz 31). Jest to nazwa folderu.
10. Zastąp wartość określoną w instrukcji <input> (wiersz 33). Jest to maksymalna liczba elementów wyświetlonych na liście dokumentów, bez względu na liczbę elementów spełniających kryteria zapytania.
11. Zastąp nazwę serwera określoną w instrukcji <input> (wiersz 35). Jest to nazwa serwera OnDemand, z którym ma się komunikować aplikacja ODWEK. Podana nazwa serwera to QUSROND.
12. Jeśli chcesz sortować elementy na liście dokumentów, sprawdź wartość określoną w instrukcji <input> (wiersz 36). W przeciwnym razie usuń wiersz 36.
13. Jeśli chcesz sortować elementy na liście dokumentów, sprawdź wartość określoną w instrukcji <input> (wiersz 37). W przeciwnym razie usuń wiersz 37.
14. Zastąp wartość określoną w instrukcji <input> (wiersz 38). Jest to identyfikator użytkownika serwera OnDemand. Określony identyfikator użytkownika musi mieć uprawnienie do otworzenia folderu i uzyskania dostępu do danych grupy aplikacji.
15. Opcjonalnie zastąp nazwę pliku szablonu określoną w instrukcji <input> (wiersz 39). Serwer OnDemand używa pliku szablonu do generowania kolejnych stron WWW. Podana nazwa szablonu to template.htm.
16. Zmodyfikuj wiersze od 40 do 43 dla pola wyszukiwania pierwszego folderu:
  - a. Wpisz nazwę pola folderu w instrukcji <font>.
  - b. Zastąp wartość określoną w polu nazwy instrukcji <input> właściwą nazwą pola folderu.
  - c. Zastąp wartość określoną w polu wartości instrukcji <input> domyślną wartością wyszukiwania.
17. Skopiuj wiersze od 40 do 43 i powtórz krok 16 dla każdego dodatkowego pola wyszukiwania folderu.
18. Zapisz zmiany i zamknij edytor tekstu.

---

## TEMPLATE.HTM

Plik TEMPLATE.HTM to domyślny plik szablonu używany przez aplikację ODWEK do generowania stron WWW w odpowiedzi na różne funkcje produktu (na przykład Logon). Plik TEMPLATE.HTM należy zastąpić kopią zawierającą treść zdefiniowaną przez użytkownika. Jednakże plik szablonu musi zawierać następujący wiersz komentarza HTML <!--AOI#Marker-->. Położenie wiersza komentarza określa miejsce umieszczenia danych wyjściowych programu ODWEK. 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. Domyślnie plik szablonu znajduje się w katalogu o nazwie określonej przez parametr TEMPLATEDIR w pliku arswww.ini. Więcej informacji znajduje się w sekcji "TEMPLATEDIR" na stronie 21.

---

## | Dalsze kroki

| Po skonfigurowaniu przykładowych aplikacji przejdź do sekcji Rozdział 5, “Instalowanie  
| przeglądarek WWW”, na stronie 49.

---

## 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.

Jeśli na serwerze OnDemand ma być zdefiniowany jeden identyfikator użytkownika dla wielu użytkowników, którzy chcą zalogować się na serwerze, a każdy z nich ma posiadać dostęp wyłącznie do własnych informacji, system należy skonfigurować w następujący sposób:

1. Proces sprawdzania wpisywania się należy udostępnić jako część aplikacji WWW.
2. Sprawdzanie wpisywania się musi mieć miejsce przed wysłaniem zapytania do serwera OnDemand.
3. Należy użyć wyniki udanego wpisania się w celu udostępnienia serwerowi OnDemand numeru konta.
4. Należy użyć funkcje API aplikacji ODWEK w celu wysłania zapytania SQL do serwera OnDemand. Zapytanie SQL powinno zawierać numer określonego konta.

Na przykład: użytkownik otwiera stronę główną "Witamy w serwisie firmowym". Aby uzyskać dostęp do informacji o koncie, użytkownik musi podać identyfikator użytkownika i numer PIN. Wartości te są sprawdzane przez aplikację WWW firmy, a nie serwer OnDemand. Po udanym wpisaniu się aplikacja WWW wyświetla stronę z podsumowaniem informacji o koncie. Na tej stronie znajduje się łącze umożliwiające wyświetlanie instrukcji konta. Po kliknięciu przez użytkownika opcji "wyświetl instrukcję konta" (view account statement) aplikacja WWW wywołuje funkcję API aplikacji ODWEK, włącznie z zapytaniem SQL zawierającym numer konta pobrany podczas procesu wpisywania się. Funkcje API wpisują się na serwer OnDemand za pomocą identyfikatora i hasła użytkownika utworzonego na potrzeby połączeń aplikacji ODWEK z serwerem, wyszukują i wczytują instrukcje użytkownika, a następnie przesyłają dokument z powrotem do przeglądarki. Przeglądarka internetowa uruchamia przeglądarkę odpowiednią dla typu danych zawartych w instrukcji.

---

## Wymagania

Przeglądarki udostępniane przez IBM wymagają oprogramowania Netscape Navigator w wersji 7.1 lub nowszej lub Microsoft Internet Explorer w wersji 6.0 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 umieścić 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 58 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

**Ważne:** jeśli planujesz udostępniać pliki zdefiniowane przez użytkownika za pomocą przeglądarki AFP Web Viewer, skonfiguruj plik instalacyjny przeglądarki AFP Web Viewer w taki sposób, aby wstrzymywał pliki zdefiniowane przez użytkownika przed rozpoczęciem instalacji przeglądarki AFP Web Viewer przez użytkowników. Więcej informacji znajduje się w sekcji “Dystrybucja plików zdefiniowanych przez użytkownika” na stronie 52.

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:

- 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.

**Ważne:** 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.

---

## Przeglądarka AFP Web viewer

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

- Kolor tła. Obsługiwane są następujące kolory (żaden inny kolor nie jest obsługiwany):
  - zielony pasek (wyświetlany z białym tłem),
  - zielony,
  - czerwony,
  - żółty,
  - czarny,
  - biały,
  - szary.
- Kolor obrazu. Obsługiwane są następujące kolory (żaden inny kolor nie jest obsługiwany):
  - żółty,
  - niebieski,
  - czerwony,
  - purpurowy,
  - zielony,
  - niebieskozielony,
  - domyślny (powinien być wyświetlany jako czarny).
- Powiększenie.

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

---

## 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 tego 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.



**Ważne:** 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 54 i “Przechowywanie plików zdefiniowanych przez użytkownika” na stronie 54.
  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 55.
  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 55.
  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
```

**Ważne:** 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.

---

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

## Konfigurowanie plików czcionek

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

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 54.
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 54.

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.

**Ważne:** 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.

**Ważne:** 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.

---

## Odwzorowanie 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 łańciski) 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 odwzorowania czcionek AFP, plików definicji czcionek oraz inne informacje techniczne związane z czcionkami AFP i TrueType znajdują się w publikacji *AFP Workbench Technical Reference*.

---

## Wyświetlanie raportów AFP

Plik FTDPOR2.INI rezydujący w katalogu instalacyjnym przeglądarki AFP Web Viewer zawiera modyfikowalne parametry, które mogą wpływać na sposób wyświetlania raportów AFP. Ta sekcja opisuje te parametry i ich wartości.

- Reguły i wiersze

Jeśli podczas wyświetlania raportu AFP reguły lub wiersze nie są wyświetlone prawidłowo, problem może być spowodowany różnicami pomiędzy sterownikami monitora. Do wyświetlenia reguł należy użyć innej metody. W sekcji Misc pliku FTDPOR2.INI zmień:

```
RuleFix=FALSE
```

na:

```
RuleFix=TRUE.
```

- Dokładność tekstu

Nieprawidłowe zastępowanie czcionek oraz wyrównywanie tekstu, w szczególności gdy parametr Dokładność tekstu (Text Fidelity) ustawiony jest na wartość Znak (Character), może wynikać z faktu, że raport został utworzony z miarą 300 pel, zamiast 240 pel. Po ustawieniu parametru 240Fidelity na FALSE raport zostanie wyświetlony z miarą 300 pel. Po ustawieniu parametru 240Fidelity na TRUE raport zostanie wyświetlony z miarą 240 pel. Domyślnie wybrana jest miara 240 pel.

- Domyślne ustawienia okna dialogowego Drukuj (Print)

Po wyświetleniu okna dialogowego Drukuj domyślnie drukowana będzie bieżąca strona raportu. Można zmienić ustawienie domyślne, aby drukować wszystkie strony raportu, określając parametr PrintAllPages=TRUE w sekcji ustawień pliku FTDPORT2.INI.

- Wielkości strony zdefiniowane przez użytkownika

Można zdefiniować dwie wielkości strony w celu wyświetlania raportów zawierających niestandardowe wielkości strony. Te dwie wielkości strony zdefiniowane przez użytkownika zostaną dodane do listy pozostałych wielkości strony, które można wybrać podczas wyświetlania raportu. Aby zdefiniować dwie wielkości strony, należy zmodyfikować następujące dwa wiersze w pliku FTDPORT2.INI:

```
PaperSize1=szerokość, długość  
PaperSize2=szerokość, długość
```

Określ szerokość w długość każdej strony w raporcie. Wszystkie wartości muszą być wyrażone jako wielokrotność 1/1440 cala.

- Jeśli wielkość strony wyrażona jest w calach, należy pomnożyć ją przez 1440.
- Jeśli wielkość strony wyrażona jest w milimetrach, należy pomnożyć ją przez 56,7, a następnie zaokrąglić do najbliższej liczby całkowitej.

Jeśli dla parametrów PaperSize1 i PaperSize2 nie są ustawione żadne wartości, domyślna wielkość strony raportów zawierających niestandardowe wielkości strony wynosi 8,5 x 11 cali.

- Czcionki True Type

Jeśli raporty mają być wyświetlone przy wykorzystaniu czcionek True Type:

1. Jeśli na stacji roboczej zainstalowany jest program Adobe Type Manager (ATM), należy go wyłączyć lub usunąć. Jeśli zainstalowane są czcionki Type 1, je również należy usunąć.
2. Do sekcji Misc pliku FTDPORT2.INI należy dodać następujący wiersz:  
TTONLY=TRUE.

---

## Wyświetlanie nakładek

Jeśli w standardowym widoku strumienia danych AFP w kliencie użytkownika serwera OnDemand wyświetlona jest nakładka, a przeglądarka AFP Web Viewer aplikacji ODWEK nie wyświetla nakładki, najprawdopodobniej przeglądarka nie odnalazła zasobu nakładki.

Aby skonfigurować przeglądarkę AFP Web Viewer tak, by wyświetlała nakładkę, określ katalog zasobu w pliku FTDPORT2.INI. Otwórz plik za pomocą edytora tekstu ASCII, na przykład aplikacji Windows Notepad, i wyszukaj pozycję o nazwie ResourceDataPath w sekcji [Preferences]. Na przykład:

```
[Preferences]  
DefaultView=DEFAULT  
ViewDataPath=C:\Program Files\IBM\OnDemand AFP Web Viewer\Data  
ResourceDataPath=C:\Program Files\IBM\OnDemand AFP Web Viewer\Resource  
FontDataPath=C:\Program Files\IBM\OnDemand AFP Web Viewer\Font
```

Pozycja ResourceDataPath używana dla aplikacji OnDemand Client powinna odpowiadać tej samej pozycji używanej dla przeglądarki AFP Web Viewer. Zarówno aplikacja OnDemand Client, jak i przeglądarka AFP Web Viewer powinny posiadać plik FTDPORT2.INI.

**Ważne:** zewnętrzne zasoby nakładek nie są pobierane razem z dokumentem AFP. W przypadku zewnętrznego zasobu (który nie jest przechowywany w tym samym pliku, co dokument AFP), musi on zostać pobrany razem z dokumentem AFP. Zasoby zewnętrzne należy przechowywać w katalogu określonym przez parametr ResourceDataPath.

Przeglądarka AFP Web Viewer nie pobiera nakładek do katalogu zasobów określonego przez parametr ResourceDataPath. W związku z tym jeśli zasób nie może być pobrany do klienckiej stacji roboczej przy użyciu innej metody, należy zmodyfikować strumień danych AFP, aby uwzględnić zasób i umieścić dokument AFP wraz z zasobem w tym samym pliku.

---

## Przeglądarka Image Web viewer

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

**Ważne:** 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.

IBM udostępnia elementy sterujące ActiveX, które umożliwiają wyświetlanie dokumentów Advanced Function Presentation (AFP) i Image przy użyciu przeglądarki. Jeśli używana jest przeglądarka Internet Explorer 5.5 z pakietem poprawek Service Pack 2 lub nowszym lub Internet Explorer 6.0, a planowane jest wyświetlanie dokumentów AFP lub Image, należy zaktualizować przeglądarkę klienta ODWEK do wersji 7.1.0.5 lub nowszej. Najnowsza wersja oprogramowania ODWEK dostępna jest pod adresem <ftp://service.software.ibm.com/software/ondemand/fixes/v71>.

---

## 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 umieścić 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 java.sun.com. 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. 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. **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.



---

## Rozdział 6. Weryfikowanie instalacji

Na tym etapie wszystkie kroki podstawowej instalacji aplikacji ODWEK powinny być zakończone.

Możesz sprawdzić, czy aplikacja ODWEK jest zainstalowana prawidłowo, wpisując się do serwera biblioteki OnDemand i otwierając folder. Jeśli używasz programu CGI, przejdź do sekcji “Weryfikowanie programu CGI” na stronie 65. Jeśli używasz serwletu Java, przejdź do sekcji “Weryfikowanie serwletu” na stronie 66.

**Ważne:** jeśli używasz funkcji API języka Java, przejdź do sekcji Dodatek D, “Podręcznik programowania funkcji API języka Java”, na stronie 101, która zawiera informacje o konfigurowaniu systemu i korzystaniu z interpretera języka Java do uruchamiania aplikacji ODWEK.

---

### Weryfikowanie programu CGI

Aby zweryfikować instalację, wykonaj następujące czynności.

**Ważne:** przed rozpoczęciem zrestartuj serwer HTTP, aby zainicjować system ze zmianami wprowadzonymi do plików konfiguracyjnych.

1. Sprawdź parametry HOST, PORT i PROTOCOL w sekcji [`@SVR@_default`] pliku `arswww.ini`. Domyślne położenie pliku `arswww.ini` to `/QIBM/UserData/OnDemand/www`. Jeśli korzystasz z obsługi wielu języków, zapoznaj się z sekcją “Implementacja” na stronie 157 w celu uzyskania bardziej szczegółowych informacji.

**Ważne:** wartość parametru PORT w pliku `arswww.ini` określa numer portu, na którym uruchomiony jest serwer biblioteki OnDemand, a nie numer portu, na którym serwer HTTP IBM oczekuje na żądania klienta.

2. Uruchom klienta (przeglądarkę).
3. W wierszu adresu przeglądarki wpisz adres URL zawierający serwer biblioteki OnDemand, port HTTP i funkcję wpisywania się. Na przykład:  
`http://odserver1.xyz.com:80/logon.htm`  
gdzie `odserver1.xyz.com` to wartość parametru HOST w pliku `arswww.ini`, `80` to numer portu HTTP, a `logon.htm` określa funkcję, którą aplikacja ODWEK ma wywołać. W tym przykładzie aplikacja ODWEK wywołuje funkcję logowania na określonym serwerze biblioteki OnDemand (plik `logon.htm` jest jedną z przykładowych aplikacji dostarczanych z aplikacją ODWEK. Sekcja Rozdział 4, “Konfigurowanie przykładowych aplikacji”, na stronie 45 zawiera instrukcje dotyczące wdrażania przykładowych aplikacji).
4. Jeśli system został skonfigurowany poprawnie, aplikacja ODWEK wyświetli ekran wpisywania się.
5. Jeśli ekran wpisywania się nie został wyświetlony, przejdź do sekcji “Rozwiązywanie problemów” na stronie 66.
6. Na ekranie wpisywania się wpisz identyfikator użytkownika i hasło odpowiednie dla serwera biblioteki OnDemand. Kliknij opcję Wyślij (Submit), aby przejść do ekranu Otwórz folder (Open a Folder).
7. Podstawowa instalacja została pomyślnie zakończona. Jednakże możesz kontynuować proces sprawdzania, wczytując różne typy dokumentów w celu utworzenia wersji tekstowej transformacji zintegrowanych z aplikacją ODWEK.

---

## Weryfikowanie serwletu

Przed przejściem dalej zatrzymaj i zrestartuj serwer aplikacji WWW, jeśli czynność ta nie została wykonana wcześniej. Szczegółowe informacje na temat konfiguracji serwletu znajdują się pod adresem <http://www.ibm.com/software/data/ondemand/400/support.html>, gdzie należy wyszukać informacje związane z produktami "ODWEK" i "WebSphere."

Aby sprawdzić, czy serwlet działa poprawnie, uruchom przeglądarkę WWW i otwórz serwlet. Określ położenie serwletu. Na przykład: `http://server/od/odservlet`, gdzie *server* to nazwa hosta systemu, na którym wdrożony został serwlet, *od* to główny element treści, a *odservlet* to odwzorowanie serwletu skonfigurowane na serwerze aplikacji WebSphere.

Jeśli wyświetlona zostanie strona WWW zawierająca tekst Internet Connection Version x.x.x.x, a argument `'_function'` nie został określony, wdrożenie powiodło się.

---

## Rozwiązywanie problemów

Ta sekcja opisuje najczęściej spotykane problemy występujące podczas weryfikowania instalacji i możliwe rozwiązania tych problemów.

Problem	Rozwiązanie
Ekran wpisywania się nie jest wyświetlany.	<p>Jeśli ekran wpisywania się nie jest wyświetlany, najbardziej prawdopodobną przyczyną są nieprawidłowe reguły odwzorowania pliku <code>logon.htm</code>. Reguły odwzorowania określone są w pliku <code>httpd.conf</code>. Aby uzyskać więcej szczegółów na ten temat, należy zapoznać się z informacjami o serwerze HTTP.</p> <p>Jeśli reguły odwzorowania są poprawne:</p> <ol style="list-style-type: none"><li>1. Sprawdź, czy uprawnienia do katalogu z przykładami <code>/QIBM/UserData/OnDemand/www/samples</code> lub do dowolnego innego katalogu, w którym rezyduje skrypt wpisywania się, są poprawne.</li><li>2. Sprawdź, czy uprawnienia do pliku <code>logon.htm</code> są poprawne.</li><li>3. Sprawdź, czy serwer HTTP działa poprawnie.</li></ol> <p>Wprowadź wszelkie niezbędne poprawki, a następnie zrestartuj serwer lub serwer aplikacji WWW i ponów próbę wpisania się.</p>
Błąd 404. Nie znaleziono pliku.	<p>Jeśli ekran wpisywania się został wyświetlony, ale podczas próby wpisania się na serwer wyświetlony został również komunikat o wystąpieniu błędu 404, sprawdź odwzorowania plików w pliku <code>httpd.conf</code>. Aby uzyskać więcej szczegółów na ten temat, należy zapoznać się z dokumentacją serwera HTTP.</p> <p>Wprowadź wszelkie niezbędne poprawki, a następnie zrestartuj serwer lub serwer aplikacji WWW i ponów próbę logowania.</p>

<b>Problem</b>	<b>Rozwiązanie</b>
Błąd 500. Błąd serwera	<p>Jeśli ekran wpisywania się został wyświetlony, ale po wprowadzeniu informacji niezbędnych do wpisania się wyświetlony został komunikat o wystąpieniu błędu 500, zapoznaj się z protokołami błędów serwera HTTP lub serwera aplikacji WWW, aby uzyskać więcej informacji. Najprawdopodobniej jest to błąd konfiguracji (na przykład brak katalogów w lokalizacji/QIBM/UserData/OnDemand/www).</p> <p>Wprowadź wszelkie niezbędne poprawki, a następnie zrestartuj serwer lub serwer aplikacji WWW i ponów próbę logowania.</p>

## Dalsze kroki

Ta sekcja opisuje sposób uzyskania informacji, które mogą być przydatne po zakończeniu instalacji aplikacji ODWEK. Zawiera listę opcjonalnych czynności konfiguracyjnych, które opisane są w tym podręczniku, informacje dla administratorów odpowiedzialnych za dystrybucję oprogramowania klienckiego ODWEK i pracujących z czcionkami AFP, informacje dla programistów, których zadaniem jest integracja aplikacji biznesowych z aplikacją ODWEK oraz wskazówki dotyczące określania problemów.

Następujące sekcje zawierają informacje na temat opcjonalnych zadań konfiguracyjnych:

- Dodatek E, “Transformowanie z formatu AFP w HTML”, na stronie 141
- Dodatek F, “Transformowanie z formatu AFP w PDF”, na stronie 145

Następujące sekcje zawierają informacje dla administratorów:

- “Odwzorowanie czcionek AFP” na stronie 56
- “Przeglądarka AFP Web viewer” na stronie 52
- “Przeglądarka Image Web viewer” na stronie 58
- “Przeglądarka danych dla drukarki wierszowej w języku Java” na stronie 58

Następujące sekcje zawierają informacje dla programistów:

- Dodatek A, “Funkcje API CGI”, na stronie 69
- Dodatek B, “Odniesienie serwletu w języku Java”, na stronie 97
- Dodatek C, “Odniesie funkcji API języka Java”, na stronie 99
- Dodatek D, “Podręcznik programowania funkcji API języka Java”, na stronie 101
- Dodatek H, “Brak danych wyjściowych HTML”, na stronie 149

Informacje na temat narzędzi do określania problemów i wskazówki znajdują się w sekcji Dodatek J, “Narzędzia do określania problemu”, na stronie 155.





---

## 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.

**Ważne:** wartości parametrów są standardowym tekstem. Istnieje możliwość, że będą one zawierać znaki niewłaściwe 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 149 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 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
- \_text
- \_docid

Poniższe parametry są opcjonalne:

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

## Przykładowe wywołanie funkcji

```
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 149 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`

## Przykładowe wywołanie funkcji

```
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. Więcej informacji na temat formatów daty obsługiwanych przez serwer OnDemand znajduje się w publikacji <i>IBM Content Manager OnDemand for i5/OS V5R1 Common Server Administration Guide</i>, sekcja 27–1161.</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 oddzielnym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 149 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. Dowlolna podana tu wartość powoduje przesłonięcie parametru PORT z pliku ARSWWW.INI.</p>



Tabela 5. Funkcja Lista trafień dokumentów (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_codepage</b> =wartość	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>_sql</b> =łańcuch	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.  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.  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.
<b>_date1</b> =wartość	Służy do określania początkowej daty zakresu dat do wyszukiwania. W przypadku określenia parametrów <b>_date1</b> i <b>_date2</b> 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).
<b>_date2</b> =wartość	Służy do określania końcowej daty zakresu dat do wyszukiwania. W przypadku określenia parametrów <b>_date1</b> i <b>_date2</b> 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).
<b>_cgibin</b> =program	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.  Wartość <i>program</i> może określać katalog względem dyrektywy <b>ServerRoot</b> lub <i>alias</i> zdefiniowany w pliku konfiguracyjnym serwera HTTP. Domyślnie ODWEK wczytuje program CGI z katalogu /QIBM/Proddata/OnDemand/www/bin.
<b>_or</b> =wartość	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>_logoff</b> =1	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*

## Przykładowe wywołanie funkcji

```
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 ARSWWW.INI. 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 149 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. Dowlona podana tu wartość powoduje przesłonięcie parametru <code>PORT</code> z pliku ARSWWW.INI.

### Sposób użycia

Następujące parametry są wymagane:

`_function`

Poniższe parametry są opcjonalne:

`_html`  
`_nohtml`  
`_port`

## **Przykładowe wywołanie funkcji**

`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 uprawnienia 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
<code>_function=logon</code>	Zalogowuje użytkownika do serwera OnDemand.
<code>_server=wartość</code>	Nazwa serwera OnDemand.
<code>_user=wartość</code>	Identyfikator użytkownika OnDemand.
<code>_password=wartość</code>	Hasło użytkownika.
<code>_new_password=wartość</code>	Nowe hasło użytkownika. Umożliwia zmianę hasła po udanym zalogowaniu się do serwera OnDemand. Jest to parametr opcjonalny.
<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 LOGON.HTML znajdującego się w katalogu określonym przez parametr <code>TEMPLATEDIR</code> w pliku ARSWWW.INI. 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 dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji zalogowania.</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. Więcej informacji na temat formatów daty obsługiwanych przez serwer OnDemand znajduje się w publikacji <i>IBM Content Manager OnDemand for i5/OS V5R1 Common Server Administration Guide</i> , sekcja 27–1161.

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 149 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>

## Sposób użycia

Następujące parametry są wymagane:

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

Poniższe parametry są opcjonalne:

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

## Przykładowe wywołanie funkcji

```
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 149 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>Ważne:</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>Ważne:</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`

## Przykładowe wywołanie funkcji

```
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, ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 149 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.

Tabela 9. Funkcja Wczytanie dokumentu (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_port</b> =wartość	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</b> =wartość	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</b> =program	<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 serwlet do tego programu lub serwletu 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</b> =wartość	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</b> =wartość	<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>Ważne:</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 26).</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>

Tabela 9. Funkcja Wczytanie dokumentu (kontynuacja)

Nazwa=Wartość	Zastosowanie
<b>_email</b> =wartość	<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>Ważne:</b> Jeśli zostanie podana wartość <b>_email=NATIVE</b>, należy sprawdzić, czy typ zawartości MIME określa żądaną przeglądarkę (więcej informacji znajduje się w sekcji “[MIMETYPES]” na stronie 26).</p> <p><b>HTML</b> ODWEK przekształca dokument EMAIL do formatu HTML.</p>
<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>Ważne:</b> Jeśli zostanie podana wartość <b>_line=NATIVE</b>, należy sprawdzić, czy typ zawartości MIME określa żądaną przeglądarkę (więcej informacji znajduje się w sekcji “[MIMETYPES]” na stronie 26).</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 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  
 \_or  
 \_afp  
 \_email  
 \_line  
 \_html  
 \_nohtml  
 \_port  
 \_codepage

\_logoff  
\_cgibin

## Przykładowe wywołanie funkcji

```
http://www.company.com/cgi-bin/arswww.cgi?_function=retrieve  
&_server=od400&_user=WWW&_password=WWW  
&_folder=wyciągi%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 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 zmiennej 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 dostarczanym 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. Więcej informacji na temat formatów daty obsługiwanych przez serwer OnDemand znajduje się w publikacji <i>IBM Content Manager OnDemand for i5/OS V5R1 Common Server Administration Guide</i> , sekcja 27–1161.

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, że ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 149 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>	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.  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.
<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`

## Przykładowe wywołanie funkcji

```
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
<code>_function=updatedoc</code>	Aktualizuje bazę danych.
<code>_server=wartość</code>	Nazwa serwera OnDemand.
<code>_user=wartość</code>	Identyfikator użytkownika OnDemand. Użytkownik musi mieć uprawnienia do aktualizowania dokumentów dla grupy aplikacji.
<code>_password=wartość</code>	Hasło użytkownika.
<code>_folder=wartość</code>	Nazwa folderu.
<code>nazwa pola folderu=wartość</code>	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.
<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 UPDATE.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 dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji aktualizacji.</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 149 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<code>_docid=ID_dokumentu</code>	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

## Przykładowe wywołanie funkcji

```

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
<code>_function=getnotes</code>	Wyświetla adnotacje.
<code>_server=wartość</code>	Nazwa serwera OnDemand.
<code>_user=wartość</code>	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).
<code>_password=wartość</code>	Hasło użytkownika.
<code>_folder=wartość</code>	Nazwa folderu.
<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 GETNOTES.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 dostarczonym z pakietem ODWEK. Można go użyć w celu utworzenia własnego szablonu dla funkcji podglądu adnotacji.</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 oddzielonymi separatorami. Dodatek H, "Brak danych wyjściowych HTML", na stronie 149 zawiera szczegóły dotyczące danych wyjściowych ASCII oddzielonych separatorami.
<code>_docid=ID_dokumentu</code>	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. Dowlona 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. Dowlona 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

## Przykładowe wywołanie funkcji

```
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. Odniesienie serwletu 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 69.

Przykłady konfiguracji serwletu Java znajdują się pod adresem <http://www-306.ibm.com/software/data/ondemand/400/support.html>, gdzie należy wyszukać informacje związane z oprogramowaniem "ODWEK" i "WebSphere."



---

## Dodatek C. Odniesie funkcji API języka Java

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

Zanim będzie można wyświetlić dokumentację, należy zainstalować oprogramowanie ODWEK na komputerze, a następnie wyodrębnić pliki dokumentacji z pliku ODAPiDoc.zip znajdującego się w katalogu /QIBM/ProdData/OnDemand/www/api. 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 języka 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,

**Wskazówka:** w sekcji Rozdział 1, “Przegląd”, na stronie 1 opisane są ograniczenia występujące podczas uzyskiwania dostępu do serwera OnDemand for OS/390 Version 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.ODApplication`**

| Klasa ta reprezentuje aplikację OnDemand. Instancja obiektu `ODApplication`  
| zapewnia programiście aplikacji dostęp do informacji określonych dla aplikacji  
| OnDemand.

| **`com.ibm.edms.od.ODApplicationGroup`**

| Klasa ta reprezentuje grupę aplikacji OnDemand. Instancja obiektu  
| `ODApplicationGroup` zapewnia programiście aplikacji dostęp do informacji  
| określonych dla grupy aplikacji OnDemand.

| **`com.ibm.edms.od.ODApplicationGroupFields`**

| Klasa ta reprezentuje pole grupy aplikacji OnDemand. Zawiera informacje o polu  
| grupy aplikacji.

| **Ostrzeżenie:** dostęp do wszystkich obiektów ODWEK może być uzyskany  
| wyłącznie w środowisku jednowątkowym.

| **`com.ibm.edms.od.LogicalView`**

| Klasa ta reprezentuje widok logiczny OnDemand. Umożliwia zbieranie informacji o  
| widoku logicznym, na przykład o liczbie wierszy nagłówka i zablokowanych polach  
| oraz funkcji powiększenia.

| **`com.ibm.edms.od.NamedQuery`**

| Klasa ta reprezentuje zapytanie OnDemand o określonej nazwie. Zawiera szczegóły

zapytania o określonej nazwie i umożliwia odtwarzanie istniejących zapytań o określonej nazwie oraz składowanie nowych zapytań o określonej nazwie na serwerze OnDemand.

#### **com.ibm.edms.od.NamedQueryCriteria**

Klasa ta reprezentuje kryteria zapytania OnDemand o określonej nazwie. Zawiera szczegóły dotyczące kryteriów wyszukiwania przechowywanych w zapytaniu o określonej nazwie.

#### **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.ODConfig**

Obiekt ODConfig języka Java to preferowana metoda konfiguracji parametrów systemowych. W następnych wersjach nie będzie można używać pliku arswwww.ini. Do tego momentu obiekt ODConfig języka Java i plik arswwww.ini są obsługiwane.

#### **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.ODHitProperties**

Klasa ta zawiera właściwości dla wartości trafienia OnDemand. Należy jej użyć do uzyskania wartości wewnętrznej właściwości OnDemand dla trafienia.

#### **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.

#### **com.ibm.edms.od.ODUser**

Klasa ta reprezentuje użytkownika OnDemand. Z tej klasy można zebrać informacje o użytkowniku, na przykład adres i numer telefonu, które są przechowywane na serwerze OnDemand.

## **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. Uruchamiając interpreter języka Java w aplikacji, podaj nazwę katalogu zdefiniowanego przez użytkownika. Przykład znajduje się w sekcji “Uruchamianie aplikacji ODWEK” na stronie 106.

Współ użyt kowaną bibliotek ą (programem us ługowym) funkcji API j ęzyka Java jest biblioteka ARS3WAPI, kt ora znajduje si e w bibliotece QRDARS.

## Konfigurowanie parametr ów systemowych

Preferowan ą metod ą konfigurowania parametr ów systemowych jest użycie obiektu `ODConfig` j ęzyka Java.

Działaj ąca instancja `ODServer` wymaga skonfigurowania ośmiu parametr ów systemowych. Parametry mo żna ustawić na warto ści domyślne za pomoc ą konstruktora domyślnego `ODConfig`:

```
<pre>
    try{
        ODConfig cfg = new ODConfig();
        ODServer srvr = new ODServer(cfg);
        srvr.initialize(null, "MyCustomApp");
        cfg.printConfig();
    }
    catch(ODException e) {
        System.out.println("Wyj ątek " + e);
    }
</pre>
```

Ten przykładowy kod konfiguruje następuj ące parametry domyślne:

```
<pre>
AfpViewOpt      PLUGIN
LineViewOpt     APPLET
MaxHits         200
MetaViewOpt     NATIVE
AppletDir       /applets
Language        ENU
TempDir         Ścieżka tymczasowa zdefiniowana metod ą Java System.getProperty("java.io.tmpdir").
TraceDir        Ścieżka tymczasowa zdefiniowana metod ą Java System.getProperty("java.io.tmpdir").
TraceLevel      0
</pre>
```

Informacje na temat tych parametr ów mo żna znaleźć w sekcji “Określanie pliku ARSWWW.INI” na stronie 13.

Następuj ące parametry odwzorowuj ą odpowiednie parametry w pliku ARSWWW.INI:

Parametr	Odwzorowuje parametr w pliku ARSWWW.INI
AfpViewOpt	AfpViewing
LineViewOpt	LineViewing
MetaViewOpt	MetaViewing

Parametr `TraceLevel` odzwierciedla poziom śledzenia wykorzystywany podczas określania problemów z aplikacj ą ODWEK. Wi ęcej informacji na temat śledzenia mo żna znaleźć w sekcji “Śledzenie” na stronie 104.

Parametry te mo żna równie ż wyraźnie ustawić za pomoc ą następuj ącego kodu przykładowego. W tym przykładowym kodzie zastosowany został inny konstruktor `ODConfig`:

```

<pre>
    try{
        ODConfig cfg = new ODConfig(ODConstant.PLUGIN, //AfpViewer
                                    ODConstant.APPLET, //LineViewer
                                    null, //MetaViewer
                                    500, //MaxHits
                                    "c:\\applets", //AppletDir
                                    "ENU", //Language
                                    "c:\\temp", //TempDir
                                    "c:\\temp\\trace", //TraceDir
                                    1); //TraceLevel

        ODServer svr = new ODServer(cfg);
        svr.initialize(null, "MyCustomApp");
        cfg.printConfig();
    }
    catch(ODException e){
        System.out.println("Wyjątek " + e);
    }
}
</pre>

```

Ten konstruktor przywraca domyślne ustawienia parametrów o wartości zerowej lub parametrów bez wartości.

**Ważne:** ten obiekt umożliwia ustawienie parametrów wyłącznie podczas budowy. Po zakończeniu budowy nie można modyfikować obiektu.

---

## 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

Funkcja śledzenia ODWEK zapisuje instrukcje śledzenia w pliku arswwww.trace określonym w pliku arswwww.ini. Funkcja ta ma pomóc przy określaniu problemów. Podobnie jak w przypadku innych form śledzenia, po włączeniu funkcji śledzenia ODWEK występuje spadek wydajności.

Plik śledzenia należy okresowo czyścić ręcznie. Plik śledzenia nie jest plikiem cyklicznym i wymaga odpowiednio dużej przestrzeni.

Aby włączyć funkcję śledzenia ODWEK, wykonaj następujące czynności:

1. Zmodyfikuj domyślnego konstruktora ODConfig. Postępuj zgodnie z instrukcjami w sekcji “Konfigurowanie parametrów systemowych” na stronie 103.
2. Zmodyfikuj poniższą sekcję debugowania w następujący sposób:

```

[DEBUG]
;Trace=None=0, Error=1, Error+Warn=2, Err+Warn+Info=3, All=4
Trace=4
TraceDir=/QIBM/UserData/OnDemand/www/logs

```

**Wskazówka:** parametr Śledzenie (Trace) umożliwia ustawienie wielu różnych poziomów funkcji śledzenia. Podczas rozwiązywania problemów związanych z aplikacją ODWEK należy ustawić najwyższy poziom funkcji śledzenia, chyba że dział wsparcia IBM zaleci inne ustawienie. Następnie w celu monitorowania aplikacji ODWEK znajdującej się w stanie ustalonym funkcję śledzenia można ustawić na niższym poziomie. Na przykład ustawienie Trace=1 generuje minimalny narzut, informując wyłącznie o warunkach błędu. Jeżeli używany jest plik arswwww.ini pochodzący z poprzedniej wersji aplikacji ODWEK, należy usunąć

nieaktualną sekcję debugowania. Wiele sekcji debugowania może negatywnie wpłynąć na działanie funkcji śledzenia ODWEK.

3. Plik `arswww.trace` tworzony jest w katalogu, do którego odnosi się parametr `TraceDir`.
4. Aby zastosować zmiany wprowadzone do pliku `arswww.ini`, należy zrestartować aplikację ODWEK.
5. Jeśli funkcja śledzenia włączona została w celu rozwiązania problemu, po włączeniu jej należy odtworzyć problem, a następnie przesłać plik `arswww.trace` do działu wsparcia IBM.

Następujące parametry w pliku `ARSWWW.INI` zapisują informacje o śledzeniu do pliku `arswww.trace` w określonym katalogu:

```
[DEBUG]
;Trace=None=0, Error=1, Error+Warn=2, Err+Warn+Info=3, All=4
TraceDir=/QIBM/UserData/OnDemand/www/logs
```

**Ważne:** 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 155 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” na stronie 104 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, "Odniesie funkcji API języka Java", na stronie 99.

---

## Uruchamianie aplikacji ODWEK

Do uruchamiania aplikacji ODWEK można używać interpretera języka Java. Podczas tworzenia, kompilacji i uruchamiania aplikacji ODWEK należy pamiętać o następujących czynnościach:

1. Funkcje eksportowania należy skonfigurować przed kompilacją i uruchomieniem aplikacji:

```
export LIBPATH=/QSYS.LIB/QRDARS.LIB
export PATH=/usr/bin:/sciezka_do_progr
export -s CLASSPATH=/QIBM/ProdData/OnDemand/www/api/ODApi.jar:/sciezka_do_progr
```

2. 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[] )
    {
        .
        .
        .
    }
}
```

3. 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.

---

## 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, "Odniesie funkcji API języka Java", na stronie 99 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;
}
}

```

---

## Połączenie z portem innym niż domyślny za pomocą funkcji API języka Java

W niektórych przypadkach może zająć konieczność nawiązania połączenia z innym portem niż domyślny za pomocą funkcji API języka Java w aplikacji ODWEK. Na przykład na serwerze OnDemand mogą istnieć dwie instancje. Jedna z nich wykorzystuje port domyślny, natomiast druga korzysta z innego portu. Jeśli system nie zostanie poprawnie skonfigurowany, po uruchomieniu programu w języku Java wyświetlony zostanie następujący błąd: "Nie można nawiązać połączenia z serwerem instancji2." (A connection cannot be established to the instance2 server.).

Aby zastosować to rozwiązanie, w źródle języka Java użyj metody ODServer.setPort() tuż przed wpisaniem się. Następnie określ nazwę hosta dla nazwy serwera (zamiast aliasu instancji). Metody nie korzystają z tej części pliku arswww.ini w celu przetłumaczenia instancji.

---

## 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 arswwww.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 <server> <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.getNumAppGroups( ) );
appl_grps = odFolder.getAppGroups( );
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_uzytkownika> <hasło> <folder> <kryteria> <operator> <wartość1>
            <wartość2> <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 arswww.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 czyszczenia
//-----
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( "*** Znalezione co najmniej jedno niedopasowanie" );
    System.out.println( "*** pomiędzy wierszami 1 i 2 trafienia" );
    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 getDocTypeString( char type )
{
    String str;

    switch( type )
    {
        case ODConstant.FileTypeAFP:
            str = "AFP";
            break;
        case ODConstant.FileTypeBMP:
            str = "BMP";
            break;
    }
}

```

```

        case ODConstant.FileTypeEMAIL:
            str = "EMAIL";
            break;
        case ODConstant.FileTypeGIF:
            str = "GIF";
            break;
        case ODConstant.FileTypeJFIF:
            str = "JFIF";
            break;
        case ODConstant.FileTypeLINE:
            str = "LINE";
            break;
        case ODConstant.FileTypeMETA:
            str = "META";
            break;
        case ODConstant.FileTypeNONE:
            str = "NONE";
            break;
        case ODConstant.FileTypePCX:
            str = "PCX";
            break;
        case ODConstant.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 arswww.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
//   setApplGroupForSearchWithSQL
//   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 prace
        //-----
        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 ( 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( );
}
}

```

## 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 zbyt 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 ( 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( );
}
}

```

---

## 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 `arswww.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 field_type;

        //-----
        // Jeśli podano zbyt mało parametrów, wyświetla składnię i kończy pracę
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "składnia: java TcListCriteria <serwer> <ID_uzytkownika> <haslo> <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], ODCConstant.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 ODCConstant.InputTypeChoice:
        case ODCConstant.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 ( 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 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 arswwww.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 arswwww.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 zwrotnego)..." );
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 arswww.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 arswwww.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_użytkownika> <hasło> <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( "Wczytywanie 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 ( 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( );
}
}
}

```

---

## Wyświetlanie informacji o uwagach

W poniższym przykładzie użyto metod `ODNote` 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 `ODNote`:

- `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 <serwer> <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 );

//-----
// Tworzenie 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`,
- `login`,
- `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 z formatu AFP w 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.

**Ważne:** w tym dokumencie nazwa AFP2HTML.INI oznacza plik konfiguracyjny. Aby określić plik zawierający opcje konfiguracyjne, zapoznaj się z sekcją "CONFIGFILE" na stronie 24.

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.

**Ważne:** 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 33. (Planując użycie funkcji API Retrieve Document (Wczytanie dokumentu), należy określić parametr \_afp=HTML. Szczegóły znajdują się w sekcji "Wczytanie dokumentu" na stronie 87.) Należy także określić katalog, w którym znajdują się programy AFP2WEB Transform (patrz "CONFIGFILE" na stronie 24).

---

### 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 którejś 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 13 zawiera listę opcji, które można określić w pliku AFP2HTML.INI w celu przekształcenia dokumentów za pomocą usługi AFP2WEB Transform.

Tabela 13. 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 13. 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 zacięniowanych 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.

**Ważne:** 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 z formatu AFP w 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.

**Ważne:** w tym dokumencie nazwa AFP2PDF.INI określa plik konfiguracyjny. Aby określić plik zawierający opcje konfiguracyjne, należy zapoznać się z sekcją “CONFIGFILE” na stronie 25.

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.

**Ważne:** 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 33. (Planując użycie funkcji API Retrieve Document (Wczytanie dokumentu), należy określić parametr \_afp=PDF. Szczegóły znajdują się w sekcji “Wczytanie dokumentu” na stronie 87.)

---

### 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 25. 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

**Ważne:** pomoc podczas konfigurowania można uzyskać, zapoznając się z dokumentacją serwera HTTP.

---

### Serwer HTTP Apache

Zamieszczony poniżej przykład przedstawia zbiór konfiguracyjny serwera HTTP (należy najpierw utworzyć konfigurację serwera HTTP za pomocą klienta administracyjnego HTTP). Przykładowa konfiguracja dotyczy instancji QUSROND i serwera HTTP ODAPACHE. Opcje konfiguracyjne zapisane pogrubioną czcionką są wymagane dla lub pokrewne z aplikacją ODWEK. Notatki dotyczące konfiguracji zaznaczone są pogrubioną kursywą i ujęte w nawiasy. 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. Aplikacja ODWEK wymaga, aby przeglądarka użytkownika końcowego akceptowała format UTF-8. W przeglądarce Microsoft Internet Explorer wybierz menu **Narzędzia > Opcje internetowe** (Tools > Internet Options), a następnie kartę Zaawansowane (Advanced). 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  
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 zbiór konfiguracyjny serwera WebSphere można uzyskać ze strony WWW IBM  
| Content Manager OnDemand for i5/OS pod adresem <http://www.ibm.com/software/data/ondemand/400/support.html>. W kategorii Uwagi techniczne (Technotes) należy wyszukać  
| przykład dla aplikacji ODWEK WebSphere.  
|



---

## Dodatek H. Brak danych wyjściowych HTML

ODWEK używa dyrektywy `_nohtml` w celu określenia typu danych wyjściowych generowanych przez dowolną funkcję (na przykład Wpisanie się). Domyślnie ODWEK generuje dane wyjściowe w formacie HTML. Podanie wartości `_nohtml=1` oznacza, ODWEK ma generować dane wyjściowe w formacie ASCII oddzielonym separatorami.

---

### Dane wyjściowe ASCII oddzielone separatorami

Dane wyjściowe ASCII oddzielone separatorami i wygenerowane przez aplikację ODWEK to zestaw rekordów wyjściowych. Rekordy te zawierają wartości łańcuchów znaków, słowa kluczowe oraz ograniczniki i separatory funkcji, rekordów i łańcuchów:

- Wartości łańcuchów znaków to dane wyjściowe funkcji, inne niż słowa kluczowe, ograniczniki i separatory. Na przykład wartości łańcucha znaków obejmują kolejną funkcję, która ma zostać wywołana, nazwę folderu, nazwy pól folderu, operatory wyszukiwania i wartości pól.
- Słowa kluczowe to określone łańcuchy znaków. Słowami 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.
- Domyślnymi ogranicznikami i separatorami łańcuchów są daszek (`^`), lewy nawias (`[`) i prawy nawias (`]`). Na przykład:

```
[folderName^folderDesc]
```

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 łańcucha znaków mogą być przechowywane na liście, oddzielone daszkiem i zamknięte znakiem lewego i prawego nawiasu. Na przykład, lista poprawnych operatorów wyszukiwania dla pola może być wyświetlona jako:

```
[1^2^4^8^16^32]
```

Domyślne znaki ograniczników i separatorów łańcuchów można przesłonić. Szczegóły znajdują się w sekcji “[NO HTML]” na stronie 31.

- Pojedyncza wartość pustego łańcucha znaków oznaczona jest brakiem wartości pomiędzy dwoma znakami cudzysłowu (“”). 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. Nazwa nie znajduje się pomiędzy podwójnym znakiem cudzysłowu.
3. Łańcuch `opis_folderu` jest opisem folderu. Opis nie znajduje się pomiędzy podwójnym znakiem cudzysłowu.

---

## Kryteria wyszukiwania

Poniższy przykład prezentuje dane ASCII oddzielone separatorami i wygenerowane przez funkcję Kryteria wyszukiwania (Search Criteria):

```
[BEGIN]\nACTION=hitListUrl\nDISPLAY_ORDER=[pole1^pole2^...poleN]\nNUMROWS=liczba_wierszy\nROW=[nazwa_kryterium^[[poprawne_operatory]^operator_domyślny]^ [typ^powiązane_dane]\n:\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 znajdują się pomiędzy podwójnym znakiem cudzysłowu.
5. Łańcuch `poprawny_operator` jest listą liczb całkowitych, które odpowiadają poprawnym operatorom wyszukiwania dla danego pola:

<b>1</b>	Równe (Equal)
<b>2</b>	Nie równe (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>	Nie w (Not in)
<b>256</b>	Podobne (Like)
<b>512</b>	Nie podobne (Not like)
<b>1024</b>	Pomiędzy (Between)
<b>2048</b>	Nie pomiędzy (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
<b>C</b>	Wybór
<b>N</b>	Normalne
<b>S</b>	Segment

- T** Wyszukiwanie tekstowe
- Z** Wyszukiwanie koloru adnotacji

8. Łańcuch `powiazane_dane` jest listą powiązaną z wartościami `operator_domyślny` i `typ`. Patrz Tabela 14.

Tabela 14. Operator domyślny i typ danych wejściowych powiązane z wartością `powiazane_dane`

operator_domyślny	typ	powiazane_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_wybór] Na przykład: [["JFIF"^"TIFF"^"PCX"]^"TIFF"] [["JFIF"^"TIFF"^"PCX"]^""]

## Lista trafień dokumentów

Poniższy przykład prezentuje dane ASCII oddzielone separatorami i wygenerowane przez funkcję Lista trafień dokumentu (Document Hit List):

```
[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. Wartości nie znajdują się pomiędzy podwójnym znakiem cudzysłowu.
4. Łańcuch `ID_dokumentu` jest identyfikatorem dokumentu.
5. Łańcuch `typ_pliku` określa typ danych dokumentu:
  - A** AFP
  - B** BMP
  - E** e-mail
  - F** JFIF
  - G** 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

Poniższy przykład prezentuje dane ASCII oddzielone separatorami i wygenerowane przez funkcję Podgląd adnotacji (View Annotations):

```
[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

Poniższy przykład prezentuje dane ASCII oddzielone separatorami i wygenerowane podczas wystąpienia błędów:

```
[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

Do określania opcji konfiguracyjnych języka narodowego (NL) służą parametry CODEPAGE i LANGUAGE w pliku ARSWWW.INI.

Parametr CODEPAGE identyfikuje stronę kodową serwera ODWEK i musi być zgodny z bazą danych OnDemand na serwerze biblioteki OnDemand. Parametr CODEPAGE musi być określony wyłącznie w sytuacji, gdy strona kodowa stacji roboczej, na której uruchomiona jest aplikacja ODWEK, jest inna niż strona kodowa bazy danych OnDemand na serwerze biblioteki 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 15 zawiera listę stron kodowych i języków DBCS obsługiwanych przez produkt OnDemand. Kolumna **CODEPAGE=** zawiera listę wartości strony kodowej i musi być określona wyłącznie w sytuacji, gdy strona kodowa stacji roboczej, na której uruchomiona jest aplikacja ODWEK, jest inna niż strona kodowa bazy danych OnDemand. W kolumnie **LANGUAGE=** znajdują się wartości określające katalogi z przetłumaczonymi komunikatami.

**Ważne:** w tabeli brak systemu Linux, ponieważ nie może on być serwerem docelowym aplikacji ODWEK. Pomimo tego że aplikacja ODWEK może być uruchomiona w systemie Linux, baza danych OnDemand (serwer biblioteki OnDemand) nie może być uruchomiona w tym systemie.

Tabela 15. 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 15. 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 16 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 16. Narzędzia do określania problemów

Narzędzie	Zastosowanie	Sposób włączenia
Dane wyjściowe HTML	Zapisanie kopii danych HTML, które aplikacja ODWEK wysłała do przeglądarki.	Z menu Plik (File) w przeglądarce wybierz opcję <b>Zapisz jako</b> (Save As).
Pliki śledzenia 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 ustaw parametr TRACE na wartość 1, 2, 3 lub 4, w zależności od wymaganego poziomu śledzenia. Plik śledzenia utworzony przez aplikację ODWEK nosi nazwę ARSWWW.TRACE i jest zapisany w katalogu określonym przez parametr TRACEDIR. (Katalog domyślny to /QIBM/UserData/OnDemand/WWW/LOGS.) <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. (Każdy serwer HTTP może pozwalać na konfigurowanie protokołowania w inny sposób i może udostępniać różne protokoły i opcje umożliwiające zbieranie bardziej lub mniej szczegółowych informacji).</li> </ol> <p><b>Ważne:</b> ponieważ w pliku protokołu można zapisać znaczną ilość informacji, IBM zaleca, aby protokołowanie włączać tylko w sytuacjach, gdy jest ono niezbędne - na przykład podczas odtwarzania problemu. 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>

Tabela 16. Narzędzia do określania problemów (kontynuacja)

Narzędzie	Zastosowanie	Sposób włączenia
Konsola języka Java	Wyświetlenie komunikatów wygenerowanych przez aplety.	<ul style="list-style-type: none"> <li>• Netscape: z menu Komunikator (Communicator) wybierz opcję <b>Narzędzia</b> (Tools), a następnie <b>Konsola Java</b> (Java Console).</li> <li>• Internet Explorer: <ol style="list-style-type: none"> <li>1. Z menu Narzędzia (Tools) wybierz opcję <b>Opcje internetowe</b> (Internet Options).</li> <li>2. Na karcie Zaawansowane (Advanced) wybierz opcję <b>Konsola Java</b> (Java Console).</li> <li>3. Zrestartuj przeglądarkę.</li> <li>4. Z menu Narzędzia (Tools) wybierz opcję Sun Java Console.</li> </ol> </li> </ul>
Funkcja śledzenia w przeglądarce AFP Web Viewer	Utrwalanie szczegółowych informacji dotyczących dokumentów AFP wyświetlanych za pomocą przeglądarki AFP Web Viewer.	<p>Upewnij się, że w pliku FLDPORT2.INI na stacji roboczej użytkownika znajdują się następujące sekcje:</p> <pre>[Misc] ViewTraceFile=d:\temp\afpplugin.log Trace=TRUE</pre> <p>Sprawdź ścieżkę do pliku protokołu. Pamiętaj, aby po zebraniu potrzebnych informacji wyłączyć protokołowanie.</p>
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.	<p>Wykonaj następujące czynności:</p> <ol style="list-style-type: none"> <li>1. Włącz protokołowanie systemowe i grupy aplikacji dla serwera OnDemand. Za pomocą klienta administracyjnego zaktualizuj parametry systemowe dla serwera.</li> <li>2. Włącz komunikaty konkretnej grupy aplikacji, którą chcesz protokołować. Zaktualizuj opcje protokołowania komunikatów dla grupy aplikacji za pomocą klienta administracyjnego.</li> </ol>



---

## Dodatek K. Obsługa wielu języków w programie CGI za pomocą serwera Apache HTTP

Do programu OnDemand Web Enablement Kit (ODWEK) Common Gateway Interface (CGI) dodana została obsługa wielu języków. Obsługa wielu języków umożliwia korzystanie z serwerów Apache HTTP obsługujących instancje OnDemand Common Server uruchomione w różnych językach na jednym serwerze i5.

---

### Wymagania wstępne dotyczące oprogramowania

Funkcja obsługi wielu języków aplikacji ODWEK w programie CGI wymaga następującego oprogramowania:

- Środowisko OnDemand Common Server (należy pamiętać, że aplikacja ODWEK nie obsługuje archiwum zbioru buforowego OnDemand).

---

### Implementacja

**Ważne:** aby zapoznać się z podobną wersją tych informacji zawierającą zrzuty ekranów, należy przejść na stronę WWW OnDemand Support dostępną pod adresem <http://www.ibm.com/software/data/ondemand/400/support.html>, a następnie wyszukać łańcuch "ODWEK multilingual".

Aby zaimplementować funkcję obsługi wielu języków, wykonaj następujące czynności (w poniższym przykładzie utworzony zostanie serwer HTTP działający w języku francuskim z identyfikatorem CCSID 1147).

1. Utwórz nowy serwer HTTP Apache.
  - a. Otwórz interfejs HTTP Admin dostępny na stronie `http://i5_hostname.company.com:2001/HTTPAdmin`, gdzie `i5_hostname.company.com` to nazwa używanego serwera i5/OS.
  - b. Kliknij opcję Tworzenie serwera HTTP (Create HTTP Server).
  - c. Określ nazwę i opis dla nowego serwera HTTP, a następnie kliknij przycisk Dalej.
  - d. Określ katalog główny dla nowego serwera HTTP, a następnie kliknij przycisk Dalej.
  - e. Określ katalog elementu głównego dokumentu dla nowego serwera HTTP, a następnie kliknij przycisk Dalej.
  - f. Określ numer portu, który nowy serwer HTTP będzie używać do nasłuchiwania (port ten nie może być używany przez inne usługi TCP/IP w systemie). Kliknij przycisk Dalej (Next).
  - g. Określ, czy ma być prowadzony protokół dostępu.
  - h. Określ, przez ile dni mają być przechowywane pliki protokołu.
  - i. Sprawdź wszystkie ustawienia, a następnie kliknij przycisk Zakończ (Finish), aby utworzyć nowy serwer HTTP (jeśli chcesz zmienić dowolne ustawienia, kliknij przycisk Wstecz (Back), aby powrócić do poprzedniego ekranu).

2. Uruchom program, aby utworzyć katalog ODWEK:

```
CALL PGM(QRDARS/QRLMINSTW) PARM('ccsid')
```

Na przykład komenda

```
CALL PGM(QRDARS/QRLMINSTW) PARM('1147')
```

powoduje utworzenie katalogu /QIBM/UserData/OnDemand/www/1147. Będzie on zawierał następujące elementy:

- pamięć podręczną,
- protokoły,
- pamięć tymczasową,
- aplety,
- obrazy,
- przykłady,
- plik arswww.ini.

3. Zaktualizuj plik httpd.conf nowego serwera HTTP, aby:

- a. Identyfikator CCSID był zgodny z identyfikatorem instancji OnDemand, do której uzyskany zostanie dostęp (na przykład jeśli instancja OnDemand używa identyfikatora CCSID 1147, identyfikator DefaultFsCCSID w pliku httpd.conf powinien mieć wartość 1147).
- b. Inne elementy w pliku httpd.conf odpowiadały potrzebom aplikacji ODWEK.

W poniższym przykładzie identyfikator CCSID serwera HTTP ma wartość 1147. Katalog główny serwera HTTP to /www/odfra. Wiersze dodane w celu zapewnienia obsługi aplikacji ODWEK opatrzone są komentarzami. Tekst zaznaczony ***pogrubioną kursywą*** należy dostosować do określonego środowiska.

```
# Configuration originally created by Create HTTP Server wizard on Thu Oct 26 09:31:27 EDT 2006
Listen *:2114
DocumentRoot /www/odfra/htdocs
Options -ExecCGI -FollowSymLinks -SymLinksIfOwnerMatch -Includes -IncludesNoExec -Indexes -MultiViews
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
LogMaint logs/access_log 7 0
LogMaint logs/error_log 7 0
# Added Lines for ODWEK
DefaultFsCCSID 1147
DefaultNetCCSID 1208
CGIConvMode EBCDIC
# End Added Lines for ODWEK
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
# Added Lines for ODWEK
LimitRequestBody 102400
ServerName iseries_hostname.company.com
UseCanonicalName Off
AccessFileName .htaccess
ScriptLogLength 200
ScriptLog /www/odfra/logs/cgi_log
Alias /logon /QIBM/UserData/OnDemand/www/1147/samples/logon.htm
AliasMatch ~/images/(.*)$ /QIBM/UserData/OnDemand/www/1147/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
# End Added Lines for ODWEK
<Directory />
Order Deny,Allow
Deny From all
# Added Lines for ODWEK
Options None
Options +ExecCGI
<LimitExcept GET HEAD OPTIONS TRACE POST PUT>
</LimitExcept>
# End Added Lines for ODWEK
</Directory>
<Directory /www/odfra/htdocs>
Order Allow,Deny
Allow From all
</Directory>
# Added Lines for ODWEK
<Directory /QIBM/ProdData/OnDemand/www/applets/>
Order allow,deny
Allow from ALL
</Directory><Directory /QIBM/ProdData/OnDemand/www/1147/samples/>
Order allow,deny
Allow from All
```

```
Options +FollowSymLinks
</Directory><Directory /QIBM/ProdData/OnDemand/www/1147/images/>
Order allow,deny
Allow from All
Options +FollowSymLinks
</Directory>
</Directory><Directory /QIBM/ProdData/OnDemand/www/1147/>
Order allow,deny
Allow from All
Options +FollowSymLinks
</Directory>
<Directory /www/odfra/>
Order allow,deny
Allow from All
</Directory>
<Directory /QSYS.LIB/QRDARS.LIB>
Order allow,deny
Allow from ALL
Options +ExecCGI
</Directory>
# End Added Lines for ODWEK
```

4. Edytuj plik arswwww.ini w katalogu /QIBM/UserData/OnDemand/www/ccsid:

```
edtf ' /QIBM/UserData/OnDemand/www/ccsid/arswww.ini '
```

gdzie *ccsid* jest żądanym identyfikatorem CCSID. Na przykład:

```
edtf ' /QIBM/UserData/OnDemand/www/1147/arswww.ini '
```

Zmień następujące wiersze w odpowiedni sposób:

```
TraceDir=/QIBM/UserData/OnDemand/www/ccsid/logs
Language=lang
CodePage=ccsid
TemplateDir=/QIBM/UserData/OnDemand/www/ccsid/samples
CacheDir=/QIBM/UserData/OnDemand/www/ccsid/CACHE
```

Na przykład:

```
TraceDir=/QIBM/UserData/OnDemand/www/1147/logs
Language=FRA
CodePage=1147
TemplateDir=/QIBM/UserData/OnDemand/www/1147/samples
CacheDir=/QIBM/UserData/OnDemand/www/1147/CACHE
```

Zmień następujące wiersze, aby określić instancję, do której ma uzyskać dostęp serwer HTTP:

```
[@SRV@_QUSROND]
HOST=Sxxxxxxx
PROTOCOL=0
PORT=1450
```

Na przykład:

```
[@SRV@_ONDFRA]
HOST=i5_hostname.company.com
PROTOCOL=0
PORT=1514
```

gdzie *i5\_hostname.company.com* to nazwa systemu i5/OS.

5. W razie konieczności zmodyfikuj inne sekcje konfiguracji pliku arswwww.ini.  
6. Edytuj plik logon.htm, aby określić nazwę domyślnej instancji:

```
edtf ' /QIBM/UserData/OnDemand/www/ccsid/samples/logon.htm '
```

Na przykład:

```
edtf ' /QIBM/UserData/OnDemand/www/1147/samples/logon.htm '
```

Wprowadź następujące zmiany:

```
<b>Server Name:</b><input type=text name=_server value=QUSROND>
```

Na przykład:

```
<b>Server Name:</b><input type=text name=_server value=ONDFRA>
```

7. W razie konieczności zmodyfikuj inne sekcje konfiguracji pliku logon.htm.
8. Uruchom nowy serwer HTTP.
9. Przetestuj konfigurację.
  - a. Otwórz stronę wpisywania się:  
`http://iseries_hostname.company.com:port_number/logon`

Na przykład:

```
http://rdr400m.raleigh.ibm.com:2114/logon
```

- b. Wpisz się do systemu.
- c. Otwórz folder.
- d. Wyszukaj element.
- e. Wyświetl zarchiwizowane dane w odpowiedni sposób:
  - dane dla drukarki wierszowej,
  - AFPDS,
  - obrazy,
  - PDF.

---

## 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 niniejszej 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 (IBM Intellectual Property Department) lub zgłaszać na piśmie pod adresem:

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

**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), WYRAŻNYCH LUB DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ, PRZYDATNOŚCI DO OKREŚLONEGO CELU ORAZ GWARANCJI, ŻE PUBLIKACJA TA NIE NARUSZA PRAW OSÓB TRZECICH.

Ustawodawstwa niektórych krajów nie dopuszczają zastrzeżeń dotyczących gwarancji wyraźnych 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 użytkowników 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 korzystania i rozpowszechniania informacji przysłanych przez użytkownika w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

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

IBM Corporation  
J46A/G4  
555 Bailey Avenue  
San Jose, CA 95141-1003  
USA

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

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 możliwości 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 uiszczenia 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 ani sugerować niezawodności, użyteczności i funkcjonalności tych programów.

W przypadku przeglądania niniejszych informacji w formie elektronicznej, zdjęcia i kolorowe ilustracje mogą nie być wyświetlane.

---

## Znaki towarowe i znaki usług

Advanced Function Presentation, AFP, AS/400, DB2, DB2 Universal Database, i5/OS, IBM, iSeries, Lotus, Lotus Notes, Operating System/400, OS/390, OS/400, Redbooks, System i, System i5, Tivoli, WebSphere oraz z/OS są znakami towarowymi International Business Machines Corporation w Stanach Zjednoczonych i/lub w innych krajach.

Adobe, Acrobat, Portable Document Format (PDF) oraz PostScript są znakami towarowymi lub zastrzeżonymi znakami towarowymi Adobe Systems Incorporated w Stanach Zjednoczonych i/lub w innych krajach.

Intel i Pentium są znakami towarowymi Intel Corporation in the United States 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.

Linux jest zastrzeżonym znakiem towarowym Linusa Torvaldsa 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.

UNIX jest zastrzeżonym znakiem towarowym the Open Group w Stanach Zjednoczonych i w innych krajach.

Nazwy innych przedsiębiorstw, produktów i usług mogą być znakami towarowymi lub znakami usług innych podmiotów.





---

# Indeks

## Znaki specjalne

@SRV@\_DEFAULT, sekcja 14

@SRV@\_serwer, sekcja 14

## A

ADDEXTENSION, parametr 32

ADDFIELDSTODOCID, parametr 32

ADDNOTES, parametr 33

adnotacje 33, 38

API 70, 95

dane wyjściowe ASCII oddzielone separatorami 152

funkcje API języka Java 131, 133

opis funkcji 6, 7

parametry 70, 95

przykładowe wywołanie funkcji 72, 96

AFP, czcionki

odwzorowanie 56

AFP, dokumenty

MIME, typ zawartości 27

przeglądanie 33

przekształcanie 33

typ multimediów 27

AFP2HTML, sekcja 23

AFP2PDF Transform

instalowanie 12

konfigurowanie 145

włączanie 24

AFP2PDF, sekcja 24

AFP2WEB Transform

instalowanie 12

konfigurowanie 141

włączanie 23

AFPVIEWING, parametr 33

aktualizowanie dokumentu

API 93

funkcje API języka Java 135

opis funkcji 7

parametry 93

przykładowe wywołanie funkcji 94

anulowanie wyszukiwania 117

Apache, serwer HTTP

pliki konfiguracyjne 147

API

adnotacje 70, 95

aktualizowanie dokumentu 93

dodanie adnotacji 70

dokument, lista trafień 75

drukowanie dokumentu 83

funkcje API CGI 69

funkcje API języka Java 99

informacje diagnostyczne 104

informacje diagnostyczne i związane ze śledzeniem 104

klasy 101

kod przykładowy 101

kryteria wyszukiwania 91

obsługa wyjątków 105

odniesienia 69, 97, 99

podgląd adnotacji 95

podręcznik programowania 101

API (*kontynuacja*)

podręcznik programowania funkcji API języka Java 101

przykładowe 101

tworzenie pakietów 101

wczytanie dokumentu 87

wydruk serwerowy 83

wylogowanie się 79

zalogowanie się 81

zmiana hasła 73

aplet AFP2HTML w języku Java

APPLETCACHEDIR, parametr 16

informacje o 1, 5

instalowanie 49

obsługa dużych obiektów 23, 143

wymagania 12, 50

aplet AFP2PDF w języku Java

katalog 25

włączanie 25

aplet danych dla drukarki wierszowej w języku Java

APPLETCACHEDIR, parametr 16

informacje o 1, 4

instalowanie 49

wymagania 50

aplety 16

APPLETCACHEDIR, parametr 16

informacje o 1

instalowanie 49

katalog 25

obsługa dużych obiektów 23, 143

włączanie 25

wymagania 50

aplety w języku Java

APPLETCACHEDIR, parametr 16

informacje o 1, 4, 5

instalowanie 49

katalog 25

obsługa dużych obiektów 23, 143

włączanie 25

wymagania 50

aplikacje WWW

przykłady 45

aplikacyjny interfejs programistyczny (API)

*Patrz* API

APPLETCACHEDIR, parametr 16

APPLETDIR, parametr 16

ARSWWW.INI, plik

@SRV@\_DEFAULT, sekcja 14

@SRV@\_serwer, sekcja 14

ADDEXTENSION, parametr 32

ADDFIELDSTODOCID, parametr 32

ADDNOTES, parametr 33

AFP2HTML, sekcja 23

AFP2PDF Transform 24

AFP2PDF, sekcja 24

AFP2WEB Transform 23

AFPVIEWING, parametr 33

APPLET, parametr 16

APPLETCACHEDIR, parametr 16

ATTACHMENT IMAGES, sekcja 30

AUTODOCRETRIEVAL, parametr 34

BEGIN, parametr 31

## ARSWWW.INI, plik (kontynuacja)

- CACHEDIR, parametr 16
- CACHEDOCS, parametr 17
- CACHEMAXTHRESHOLD, parametr 17
- CACHEMINTHRESHOLD, parametr 17
- CACHESIZE, parametr 18
- CACHEUSERIDS, parametr 18
- CODEPAGE, parametr 19
- CONFIGFILE, parametr 24, 25
- CONFIGURATION, sekcja 15
  - debug, sekcja 40
- DEFAULT BROWSER, sekcja 32
- DOCSIZE, parametr 19
- EMAILVIEWING, parametr 34
- ENCRYPTCOOKIES, parametr 35
- ENCRYPTURL, parametr 35
- END, parametr 31
- FOLDERDESC, parametr 35
- HOST, parametr 15
- IMAGEDIR, parametr 19
- INSTALLDIR, parametr 24, 25
  - konfigurowanie 13
- LANGUAGE, parametr 20
- LINEVIEWING, parametr 36
- LOG, parametr 155
- MAXHITS, parametr 36
- MIMETYPES, sekcja 26
- NOHTML, sekcja 31
- NOLINKS, parametr 37
- ODApplet.jre.path.IE, parametr 37
- ODApplet.jre.path.NN, parametr 37
- ODApplet.jre.version, parametr 37
- ODApplet.version, parametr 37
  - określanie 13
- PORT, parametr 14, 15
- PROTOCOL, parametr 14, 15
  - przeglądarka, opcje 39
  - przeglądarka, sekcja 39
- REPORTSERVERTIMEOUT, parametr 22
- SECURITY, sekcja 22
- SEPARATOR, parametr 32
- SERVERACCESS, parametr 22
- SERVERPRINT, parametr 37
- SERVERPRINTERS, parametr 38
- SHOWDOCLOCATION, parametr 38
- ShowSearchString, parametr 20
- TEMPDIR, parametr 21
- TEMPLATEDIR, parametr 21
- TRACE, parametr 40
- TRACEDIR, parametr 40
- USEEXECUTABLE, parametr 24, 26
- VIEWNOTES, parametr 38

atrybut method znacznika form 7

- ATTACHMENT IMAGES, sekcja 30
- AUTODOCRETRIEVAL, parametr 34

## B

- BEGIN, parametr 31
- błędy 104, 155
- BMP, dokumenty
  - MIME, typ zawartości 27
  - typ multimediów 27
- BMP, załączniki 30
- brak danych wyjściowych HTML 31, 149

## C

- CACHEDIR, parametr 16
- CACHEDOCS, parametr 17
- CACHEMAXTHRESHOLD, parametr 17
- CACHEMINTHRESHOLD, parametr 17
- CACHESIZE, parametr 18
- CACHEUSERIDS, parametr 18
- CODEPAGE, parametr 19, 153
- CONFIGFILE, parametr 24, 25
- CONFIGURATION, sekcja 15
  - czcionki
    - AFP 56
    - odzworowanie 56
    - TrueType 56

## D

- dane dla drukarki wierszowej
  - MIME, typ zawartości 29
  - przeglądanie 36
  - przekształcanie 36
  - typ multimediów 29
- dane wyjściowe ASCII
  - adnotacje 152
  - dokument, lista trafień 151
  - format 149
  - generowane przez OnDemand 149
  - komunikat o błędzie 152
  - komunikaty 152
  - kryteria wyszukiwania 150
  - podgląd adnotacji 152
  - zalogowanie się 149
- dane wyjściowe ASCII oddzielone separatorami
  - adnotacje 152
  - dokument, lista trafień 151
  - format 149
  - generowane przez OnDemand 149
  - komunikat o błędzie 152
  - komunikaty 152
  - kryteria wyszukiwania 150
  - podgląd adnotacji 152
  - separatory 31
  - zalogowanie się 149
- Datowanie
  - transakcji 22
- DBCS 153
- debug, sekcja 40
- DEFAULT BROWSER, sekcja 32
- DOCSIZE, parametr 19
- dodanie adnotacji
  - API 70
  - opis funkcji 6
  - parametry 70
  - przykładowe wywołanie funkcji 72
- dokument, lista trafień
  - API 75
  - dane wyjściowe ASCII oddzielone separatorami 151
  - funkcje API języka Java 111, 115, 124, 126
  - opis funkcji 6
  - parametry 75
  - przykładowe wywołanie funkcji 78
- dokumenty
  - AFP 33
  - aktualizowanie za pomocą funkcji API języka Java 135
  - dane dla drukarki wierszowej 36
  - drukowanie za pomocą funkcji API języka Java 129

- dokumenty (*kontynuacja*)
  - EMAIL 34
  - MIME, typ zawartości 26
  - odsyłacze 37
  - pamięć podręczna 17
  - przeglądanie 33, 34, 36
  - przekształcanie 33, 34, 36
  - typ multimediiów 26
  - wczytywanie 34
- dokumenty, funkcje API języka Java 124, 126
- domyślna przeglądarka, opcje 32
- drukowanie
  - funkcje API języka Java 129
  - serwer 37, 38
- drukowanie dokumentu
  - API 83
  - funkcje API języka Java 129
  - opis funkcji 7
  - parametry 83
  - przykładowe wywołanie funkcji 86
- duże obiekty 23, 143

## E

- EMAIL, dokumenty
  - MIME, typ zawartości 28
  - przeglądanie 34
  - przekształcanie 34
  - typ multimediiów 28
- EMAILVIEWING, parametr 34
- ENCRYPTCOOKIES, parametr 35
- ENCRYPTURL, parametr 35
- END, parametr 31

## F

- folder, wyszukiwanie za pomocą funkcji API języka Java 111, 115, 117, 120, 124
- folder, wyświetlanie grup aplikacji używających funkcji API języka Java 110
- FOLDERDESC, parametr 35
- 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 69
- funkcje API języka Java
  - informacje o 1
  - odniesienia 99
  - podręcznik programowania 101
  - wymagania dotyczące oprogramowania 1

## G

- GIF, dokumenty
  - MIME, typ zawartości 28

- GIF, dokumenty (*kontynuacja*)
  - typ multimediiów 28
- GIF, załączniki 31
- grupy aplikacji w folderze
  - funkcje API języka Java 110

## H

- hasła
  - funkcje API języka Java 107, 137
- host, nazwa 15
- HOST, parametr 15
- httpd.conf, plik 147

## I

- i5/OS
  - instalacja 12
- identyfikatory użytkowników
  - funkcje API języka Java 107
  - pamięć podręczna 18
- Image Web Viewer
  - informacje o 1
  - instalowanie 49
  - wymagania 50
- IMAGEDIR, parametr 19
- informacje cookie 35, 50
- informacje diagnostyczne 104
- informacje diagnostyczne i związane ze śledzeniem 104
- informacje o produkcie OnDemand Internet Connection 1
- informacje o tej publikacji 1
- instalacja
  - aplet AFP2HTML w języku Java 49
  - aplet danych dla drukarki wierszowej w języku Java 49
  - aplety 49
  - aplety w języku Java 49
  - ARSWWW.INI, plik 13
  - dostosowywanie 52
  - i5/OS 12
  - Image Web Viewer 49
  - moduły dodatkowe 49
  - oprogramowanie ODWEK 11
  - pliki zdefiniowane przez użytkownika 52
  - przeglądarka AFP Web Viewer 49
  - serwer HTTP 11
  - stacja robocza użytkownika 49
  - wymagania 11
- INSTALLDIR, parametr 24, 25

## J

- język 20, 153
- języki używające zestawu znaków dwubajtowych 153
- JFIF, dokumenty
  - MIME, typ zawartości 28
  - typ multimediiów 28
- JVM 50

## K

- katalog lokalny
  - funkcje API języka Java 107
- katalog obrazów 19
- klasy 101
- kod przykładowy 101

komunikacja, protokoły 14, 15  
komunikat o błędzie  
  dane wyjściowe ASCII oddzielone separatorami 152  
komunikaty 20  
  dane wyjściowe ASCII oddzielone separatorami 152  
konfigurowanie  
  ARSWWW.INI, plik 13  
  oprogramowanie ODWEK 11  
  plik konfiguracyjny AFP2HTML 141  
  plik konfiguracyjny AFP2PDF 145  
  serwer HTTP 11  
kryteria wyszukiwania  
  API 91  
  dane wyjściowe ASCII oddzielone separatorami 150  
  funkcje API języka Java 111, 115, 120  
  łańcuch SQL 115  
  opis funkcji 6  
  parametry 91  
  przykładowe wywołanie funkcji 92

## L

LANGUAGE, parametr 20, 153  
limit czasu 22  
limit czasu nieaktywności 22  
Limit czasu nieaktywności  
  i datowanie 22  
LINEVIEWING, parametr 36  
lista dostępu do serwera 22  
Lista kontrolna  
  instalacji dla 9  
LOG, parametr 155

## Ł

łańcuch wyszukiwania SQL za pomocą funkcji API języka Java 115

## M

maksymalna liczba trafień 36  
MAXHITS, parametr 36  
metoda GET 7  
metoda POST 7  
MIME, typ zawartości 26, 111  
MIMETYPES, sekcja 26  
moduły dodatkowe  
  informacje o 1  
  instalowanie 49

## N

nawiązywanie połączenia z serwerem 106, 107  
nazwa aplikacji  
  funkcje API języka Java 107  
nazwa folderu, funkcje API języka Java 123  
NLS 19, 20, 153  
NOHTML, sekcja 31  
NOLINKS, parametr 37  
numer portu 14, 15

## O

obsługa języków narodowych 153  
obsługa wyjątków 105  
ochrona 7, 22, 35

ochrona danych 7  
ochrona serwera 7, 22  
ODApplet.jre.path.IE, parametr 37  
ODApplet.jre.path.NN, parametr 37  
ODApplet.jre.version, parametr 37  
ODApplet.version, parametr 37  
ODCallback 128  
ODCriteria  
  aktualizowanie dokumentu 135  
  dokumenty, aktualizowanie 135  
  nazwa 111  
  operandy 111, 117, 120  
  wartości wyszukiwania 111, 117, 120  
ODCriteria.getFixedValues 120  
ODCriteria.getName 111  
ODCriteria.getOperand 111, 117  
ODCriteria.getType 120  
ODCriteria.getValidOperands 120  
ODCriteria.getValues 120  
ODCriteria.setOperand 120, 135  
ODCriteria.setSearchValue 111, 135  
ODCriteria.setSearchValues 111, 117, 120  
ODFolder  
  anulowanie wyszukiwania 117  
  dokument, drukowanie 129  
  dokument, wczytanie 126  
  drukowanie dokumentów 129  
  grupy aplikacji 110  
  kolejność wyświetlania 111, 124  
  komunikat 111  
  kryteria 111, 117, 120  
  nazwa 111, 124  
  opis 111  
  wczytanie dokumentu 126  
  wyszukiwanie 111, 115, 117, 120, 124, 126  
  zamykanie 110, 111, 115, 117  
ODFolder.close 110, 111, 115, 117, 126  
ODFolder.getAppGroups 110  
ODFolder.getCriteria 111, 117, 120  
ODFolder.getDescription 111  
ODFolder.getDisplayOrder 111, 115, 124  
ODFolder.getName 111, 124  
ODFolder.getNumAppGroups 110  
ODFolder.getSearchMessage 111  
ODFolder.printDocs 129  
ODFolder.retrieve 126  
ODFolder.search 111, 115, 117, 124, 126  
ODFolder.setAppGroupForSearchWithSQL 115  
ODHit  
  adnotacje 131, 133  
  aktualizowanie dokumentów 135  
  dokument, aktualizowanie 135  
  dokument, wczytanie 126  
  lista dokumentów 124  
  MIME, typ zawartości 111  
  położenie dokumentu 111  
  typ dokumentu 111  
  uwagi 131, 133  
  wartości wyświetlania 111, 115  
  wartość wyświetlania 124  
  wczytanie dokumentu 126  
ODHit.addNote 133  
ODHit.getDisplayValue 111, 115, 124, 135  
ODHit.getDisplayValues 111  
ODHit.getDocId 111, 126  
ODHit.getDocLocation 111  
ODHit.getDocType 111

ODHit.getMimeType 111  
 ODHit.getNotes 131, 133  
 ODHit.retrieve 126  
 ODHit.update 135  
 odłączanie od serwera 107  
 odniesienia  
   API 69, 97, 99  
   funkcje API CGI 69  
   funkcje API języka Java 99  
   serwlet 97  
   serwlet w języku Java 97  
 ODNNote  
   adnotacje 131, 133  
   data 131  
   godzina 131  
   identyfikator użytkownika 131  
   kolor 131  
   nazwa grupy 131  
   pozycja 131  
   strona 131  
   tekst 131  
 ODNNote.getColor 131  
 ODNNote.getDateTime 131  
 ODNNote.getGroupName 131  
 ODNNote.getOffsetX 131  
 ODNNote.getOffsetY 131  
 ODNNote.getPageNum 131  
 ODNNote.getText 131  
 ODNNote.getUserid 131  
 ODNNote.isOkToCopy 131, 133  
 ODNNote.isPublic 131, 133  
 ODNNote.setGroupName 133  
 ODNNote.setText 133  
 ODServer  
   anulowanie wyszukiwania 117  
   dokument, wczytanie 126  
   drukarki 129  
   drukarki serwera 129  
   folder, otwieranie 126  
   hasło 107, 137  
   identyfikator użytkownika 107  
   katalog lokalny 107  
   łączenie z 107  
   nawiązywanie połączenia z serwerem 106  
   nazwa aplikacji 107  
   nazwa folderu 123  
   odłączanie od 107  
   opis folderu 123  
   otwieranie folderu 120, 126  
   port 107  
   serwer 107  
   typ połączenia 107  
   ustawianie haseł 137  
   ustawianie i uzyskiwanie haseł 107  
   ustawianie i uzyskiwanie identyfikatorów użytkowników 107  
   wczytanie dokumentu 126  
   zmiana hasła 137  
 ODServer.cancel 117  
 ODServer.changePassword 137  
 ODServer.getConnectType 107  
 ODServer.getFolderNames 123  
 ODServer.getFoldersDescription 123  
 ODServer.getLocalDir 107  
 ODServer.getNumFolders 123  
 ODServer.getPassword 107  
 ODServer.getPort 107  
 ODServer.getServerName 107  
 ODServer.getServerPrinters 129  
 ODServer.getUserId 107  
 ODServer.logoff 106, 107  
 ODServer.logon 106, 107  
 ODServer.openFolder 120, 126  
 ODServer.retrieve 126  
 ODServer.setApplicationName 107  
 ODServer.setConnectType 107  
 ODServer.setLocalDir 107  
 ODServer.setPassword 107  
 ODServer.setPort 107  
 ODServer.setServer 107  
 ODServer.setUserId 107  
 ODServer.terminate 106, 107  
 odsyłacze 37  
 odwzorowanie czcionek AFP 56  
 określanie problemów 155  
 OnDemand Internet Connection  
   informacje o 1  
 OnDemand, opcje serwera  
   @SRV@\_DEFAULT, sekcja 14  
   @SRV@\_server, sekcja 14  
   HOST, parametr 15  
   parametry 14  
   PORT, parametr 14, 15  
   PROTOCOL, parametr 14, 15  
   wartości domyślne 14  
 opcje serwera HTTP  
   AFP2HTML, sekcja 23  
   AFP2PDF Transform 24  
   AFP2PDF, sekcja 24  
   AFP2WEB Transform 23  
   APPLETDIR, parametr 16  
   ATTACHMENT IMAGES, sekcja 30  
   BEGIN, parametr 31  
   CACHEDIR, parametr 16  
   CACHEDOCS, parametr 17  
   CACHEMAXTHRESHOLD, parametr 17  
   CACHEMINTHRESHOLD, parametr 17  
   CACHESIZE, parametr 18  
   CACHEUSERIDS, parametr 18  
   CODEPAGE, parametr 19  
   CONFIGFILE, parametr 24, 25  
   CONFIGURATION, sekcja 15  
   debugowanie 40  
   domyślna przeglądarka 32  
   END, parametr 31  
   IMAGEDIR, parametr 19  
   INSTALLDIR, parametr 24, 25  
   LANGUAGE, parametr 20  
   MIMETYPES, sekcja 26  
   NOHTML, sekcja 31  
   przeglądarki 32, 39  
   REPORTSERVERTIMEOUT, parametr 22  
   SECURITY, sekcja 22  
   SEPARATOR, parametr 32  
   SERVERACCESS, parametr 22  
   ShowSearchString, parametr 20  
   TEMPDIR, parametr 21  
   TEMPLATEDIR, parametr 21  
   USEEXECUTABLE, parametr 24, 26  
 opcje serwera WWW  
   DBCS 153  
   język 153  
   NLS 153  
   strona kodowa 153  
 operandy, funkcje API języka Java 111

- opis folderu, funkcje API języka Java 123
- oprogramowanie ODWEK
  - instalowanie 11
- oprogramowanie serwera HTTP
  - instalowanie 11

## P

- pakiety, hierarchia w języku Java 101
- pamięć podręczna 16, 17, 18
- pamięć podręczna dla dokumentów 17
- pamięć podręczna, katalog 16
- pamięć tymczasowa 21
- parametry
  - @SRV@\_DEFAULT, sekcja 14
  - @SRV@\_serwer, sekcja 14
  - ADDEXTENSION 32
  - ADDFIELDSTODOCID 32
  - ADDNOTES 33
  - AFP2HTML, sekcja 23
  - AFP2PDF, sekcja 24
  - AFPVIEWING 33
  - APPLETCACHEDIR 16
  - APPLETDIR 16
  - ATTACHMENT IMAGES, sekcja 30
  - AUTODOCRETRIEVAL 34
  - BEGIN 31
  - CACHEDIR 16
  - CACHEDOCS 17
  - CACHEMAXTHRESHOLD 17
  - CACHEMINTHRESHOLD 17
  - CACHESIZE 18
  - CACHEUSERIDS 18
  - CODEPAGE 19
  - CONFIGFILE 24, 25
  - CONFIGURATION, sekcja 15
  - DOCSIZE 19
  - EMAILVIEWING 34
  - ENCRYPTCOOKIES 35
  - ENCRYPTURL 35
  - END 31
  - FOLDERDESC 35
  - HOST 15
  - IMAGEDIR 19
  - INSTALLDIR 24, 25
  - LANGUAGE 20
  - LINEVIEWING 36
  - LOG 155
  - MAXHITS 36
  - NOLINKS 37
  - ODApplet.jre.path.IE 37
  - ODApplet.jre.path.NN 37
  - ODApplet.version 37
  - PORT 14, 15
  - PROTOCOL 14, 15
  - REPORTSERVERTIMEOUT 22
  - SECURITY, sekcja 22
  - SEPARATOR 32
  - SERVERACCESS 22
  - SERVERPRINT 37
  - SERVERPRINTERS 38
  - SHOWDOCLOCATION 38
  - ShowSearchString 20
  - TEMPDIR 21
  - TEMPLATEDIR 21
  - TRACE 40
  - TRACEDIR 40

- parametry (*kontynuacja*)
  - USEEXECUTABLE 24, 26
  - VIEWNOTES 38
- PCX, dokumenty
  - MIME, typ zawartości 29
  - typ multimediów 29
- PDF, dokumenty
  - MIME, typ zawartości 29
  - typ multimediów 29
- plik konfiguracyjny AFP2HTML 141
- plik konfiguracyjny AFP2PDF 145
- pliki protokołów 40, 155
- pliki zdefiniowane przez użytkownika
  - instalowanie 52
- podgląd adnotacji
  - API 95
  - dane wyjściowe ASCII oddzielone separatorami 152
  - opis funkcji 7
  - parametry 95
  - przykładowe wywołanie funkcji 96
- podręcznik programowania
  - API 101
  - funkcje API języka Java 101
- położenie dokumentu 38
- pomoc 155
- port
  - funkcje API języka Java 107
- PORT, parametr 14, 15
- PROTOCOL, parametr 14, 15
- protokołowanie 40, 155
- protokoły 14, 15
- przegląd 1
- przeglądarka AFP Web Viewer
  - AFP, czcionki 56
  - czcionki 56
  - dostosowanie instalacji 52
  - informacje o 1
  - instalowanie 49
  - instalowanie plików zdefiniowanych przez użytkownika 52
  - konfigurowanie 52
  - odzworowanie czcionek AFP 56
  - pliki zdefiniowane przez użytkownika 52
  - wymagania 50
- przeglądarka danych dla drukarki wierszowej
  - konfigurowanie 58
  - ODApplet.jre.path.IE, parametr 37
  - ODApplet.jre.path.NN, parametr 37
  - ODApplet.jre.version, parametr 37
  - ODApplet.version, parametr 37
- przeglądarka danych dla drukarki wierszowej w języku Java
  - konfigurowanie 58
  - ODApplet.jre.path.IE, parametr 37
  - ODApplet.jre.path.NN, parametr 37
  - ODApplet.jre.version, parametr 37
  - ODApplet.version, parametr 37
- przeglądarka Image Web viewer
  - konfigurowanie 58
- przeglądarka, opcje
  - DEFAULT BROWSER, sekcja 32
  - przeglądarka, sekcja 39
- przeglądarka, sekcja 39
- przeglądarki
  - informacje cookie 50
  - JVM 50
  - obsługiwane 50
  - wirtualna maszyna języka Java 50
- przygotowanie do użycia produktu OnDemand Internet Connection 1

przykładowe 101  
przykładowe aplikacje 45

## R

REPORTSERVERTIMEOUT, parametr 22

## S

SECURITY, sekcja 22  
SEPARATOR, parametr 32  
separatory 31  
separatory danych wyjściowych 31  
SERVERACCESS, parametr 22  
SERVERPRINT, parametr 37  
SERVERPRINTERS, parametr 38  
serwer  
funkcje API języka Java 107  
serwer HTTP  
httpd.conf, plik 147  
pliki konfiguracyjne 147  
serwlet  
odniesienia 97  
serwlet w języku Java  
odniesienia 97  
SHOWDOCLOCATION, parametr 38  
ShowSearchString, parametr 20  
strona kodowa 19, 153  
strony WWW  
przykłady 45  
szyfrowanie 35

## Ś

śledzenie problemów 155

## T

TCP/IP, protokół komunikacyjny 14, 15  
TEMPDIR, parametr 21  
TEMPLATEDIR, parametr 21  
TIFF, dokumenty  
MIME, typ zawartości 30  
typ multimediów 30  
TRACE, parametr 40  
TRACEDIR, parametr 40  
TrueType, czcionki  
odwzorowanie czcionek AFP na 56  
TXT, załączniki 31  
tymczasowy katalog roboczy 21  
typ dokumentu, funkcje API języka Java 111  
typ połączenia  
funkcje API języka Java 107  
typ/podtyp multimediów 26

## U

UPDATETIMESTAMP, sekcja 22  
uprawnienia 12  
uprawnienia do katalogu 12  
USEEXECUTABLE, parametr 24, 26  
ustawianie haseł 137  
uwagi 33, 38

## V

VIEWNOTES, parametr 38

## W

wartości wyszukiwania, funkcje API języka Java 111, 115  
wczytanie dokumentu  
API 87  
opis funkcji 6  
parametry 87  
przykładowe wywołanie funkcji 90  
wczytywanie  
dokumenty 34  
wczytywanie dokumentu 126  
wielkość pamięci podręcznej 18  
wirtualna maszyna języka Java 50  
wydruk serwerowy  
API 83  
funkcje API języka Java 129  
opis funkcji 7  
parametry 83  
przykładowe wywołanie funkcji 86  
włączanie 37, 38  
wylogowanie się  
API 79  
opis funkcji 6  
parametry 79  
przykładowe wywołanie funkcji 80  
wymagania  
AFP2PDF Transform 12  
AFP2WEB Transform 12  
aplet AFP2HTML w języku Java 12  
funkcje API języka Java 1  
informacje cookie 50  
pamięć podręczna 12  
pamięć podręczna dla dokumentów 12  
serwer 11  
serwer HTTP 11  
wirtualna maszyna języka Java 50  
wyniki zapytania 36  
wyszukiwanie w folderze 111, 115, 117, 120, 124  
wyświetlanie wartości, funkcje API języka Java 111  
wyświetlenie położenia dokumentu 38

## Z

zalogowanie się  
API 81  
dane wyjściowe ASCII oddzielone separatorami 149  
opis funkcji 6  
parametry 81  
przykładowe wywołanie funkcji 82  
załączniki 30, 31  
zmiana hasła 137  
API 73  
opis funkcji 6  
parametry 73  
przykładowe wywołanie funkcji 74









Numer Programu: 5761-RD1

SC85-0120-04

