

Natural for z/OS

Natural Web I/O Interface

Version 9.2.5

Oktober 2026

This document applies to Natural for z/OS Version 9.2.5 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 1979-2026 Software GmbH, Darmstadt, Germany and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software GmbH product names are either trademarks or registered trademarks of Software GmbH and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software GmbH and/or its subsidiaries is located at <https://softwareag.com/licenses>.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software GmbH Products / Copyright and Trademark Notices of Software GmbH Products". These documents are part of the product documentation, located at <https://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use of this software is subject to adherence to Software GmbH's licensing conditions and terms. These terms are part of the product documentation, located at <https://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Document ID: NATMF-NATWEBIO-925-20261013DE

Inhaltsverzeichnis

Vorwort	vii
1 Über diese Dokumentation	1
Dokumentationskonventionen	2
Online-Informationen und Support	2
Datenschutz	3
I Einführung	5
2 Einführung	7
Wozu dient das Natural Web I/O Interface?	8
Komponenten des Natural Web I/O Interface	8
Ausführen einer Natural-Anwendung in einem Webbrowser	9
Client-Server-Kompatibilität	10
Terminologie	11
Unterschiede in einer SPoD-Entwicklungsumgebung	11
Einschränkungen bei der Verwendung des Natural Web I/O Interface mit Natural-Anwendungen	12
Unterschiede zwischen dem Natural Web I/O Interface Client und einer Terminal-Emulation	13
II Vorstellung des Natural Web I/O Interface Server CICS Adapter	15
3 Vorstellung des Natural Web I/O Interface Server CICS Adapter	17
Zweck des Natural Web I/O Interface Server CICS Adapter	18
CICS-Unterstützung	18
Zusammenwirken der Produkte	19
III Vorstellung des Natural Web I/O Interface Server IMS Adapter	21
4 Vorstellung des Natural Web I/O Interface Server IMS Adapter	23
Zweck des Natural Web I/O Interface Server IMS Adapter	24
IMS TM-Unterstützung	24
Zusammenwirken der Produkte	25
IV Installation und Konfiguration des Natural Web I/O Interface Server	27
5 Konzept und Struktur des Natural Web I/O Interface Server	29
Konzept des Web I/O Interface Server	30
Frontend Stub NATRNWO	30
Frontend	32
Server-Monitoring	32
6 Voraussetzungen	33
Allgemeine Voraussetzungen für die Installation des Web I/O Interface Server	34
Voraussetzungen für den Web I/O Interface Server für z/OS	34
7 Installation des Natural Web I/O Interface Server unter z/OS	35
Voraussetzungen für z/OS	36
Inhalt des Distributionsmediums des Web I/O Interface Server	36
Installationsschritte	36
8 Konfigurieren des Natural Web I/O Interface Server	41
Konfigurationserfordernisse für z/OS	42

Web I/O Interface Server-Konfigurationsdatei für z/OS	50
Konfigurationsparameter für den Web I/O Interface Server	51
Beispiel für eine Web I/O Interface Server-Konfigurationsdatei	62
Web I/O Interface Server Datasets für z/OS	62
Web I/O Interface Server User Exits	63
9 Installation des Natural Web I/O Interface Server CICS Adapter unter z/OS	65
Voraussetzungen für den CICS Adapter	66
Installationsvorgang	66
10 Konfiguration des Natural Web I/O Interface Server CICS Adapter	71
Konfigurationsdatei für den CICS Adapter	72
Konfigurationsparameter für den CICS Adapter	72
11 Installation des Natural Web I/O Interface Server IMS Adapter	77
Voraussetzungen für den IMS Adapter	78
Beispieljobs	78
Installationsverfahren	78
12 Konfiguration des Natural Web I/O Interface Server IMS Adapter	83
Konfigurationsdatei für den IMS Adapter	84
Konfigurationsparameter für den IMS Adapter	84
V Installation des Natural Web I/O Interface Client	87
13 Voraussetzungen für den Natural Web I/O Interface Client	89
Servlet Container	90
Natural for z/OS	90
Natural for Linux	90
Natural for Windows	91
Browser-Voraussetzungen	91
14 Installation des Natural Web I/O Interface Client auf Apache Tomcat	93
Installationsschritte	94
Überprüfung der Installation	95
15 Migration des Natural Web I/O Interface Client von IIS nach Apache Tomcat	97
Bevor Sie den Natural Web I/O Interface Client installieren	98
Installieren des Natural Web I/O Interface Client auf Apache Tomcat	99
Konfigurieren des Natural Web I/O Interface Client auf Apache Tomcat	99
VI Konfiguration des Client	103
16 Informationen zur Anmeldeseite (Logon Page)	105
Starten einer Natural-Anwendung über die Anmeldeseite (Logon Page)	106
Beispiele für Anmeldeseiten	106
Dynamisches Ändern des CICS-Transaktionsnamens, des Hostnamens und der Portnummer beim Starten einer Sitzung	107
Angabe eines Passworts auf der Anmeldeseite (Logon Page)	108
Ändern des Passworts auf der Anmeldeseite	108
Browser-Einschränkungen	109
17 Natural Client-Konfigurationstool	111

Aufrufen des Konfigurationstools	112
Session Configuration - Sitzungskonfiguration	114
Protokollierungskonfiguration (Logging Configuration-Seite)	122
Anmeldeseite (Logon Page)	122
Abmelden (Logout)	123
18 Natural Web I/O-Stylesheets	125
Name und Speicherort der Stylesheets	126
Stylesheets bearbeiten	126
Position der Hauptausgabe (Main Output) und der PF-Tasten ändern	126
Schriftgröße ändern (Font Size)	127
Schriftart ändern (Font Type)	128
Unterstrichenen und blinkenden Text definieren	129
Kursiven Text definieren (Italic Text)	130
Fettgedruckten Text definieren (Bold Text)	130
Verschiedene Stile für Ausgabefelder definieren	131
Natural-Fenster ändern	131
Meldungszeile ändern (Message Line)	132
Hintergrundfarbe ändern (Background Color)	132
Farbattribute ändern (Color Attributes)	133
Stil der PF-Tasten ändern	134
JavaScript und XSLT-Dateien	135
19 Starten einer Natural-Anwendung mit einer URL	137
20 Konfigurieren der containerverwalteten Security	139
Informationen zur containerverwalteten Security	140
Name und Speicherort der Konfigurationsdatei für die Security	140
Security aktivieren	140
Security-Beschränkungen definieren	141
Rollen definieren	141
Authentifizierungsmethode auswählen	142
UserDatabaseRealm konfigurieren	142
21 Konfigurieren von SSL	143
Informationen zum Konfigurieren von SSL	144
Eigene Trust-Datei erstellen	144
SSL-Verwendung in der Konfigurationsdatei definieren	145
22 Protokollierung (Logging)	147
Informationen zur Protokollierung	148
Name und Speicherort der Konfigurationsdatei für die Protokollierung	148
Aufrufen der Seite Logging Configuration	148
Übersicht über die Optionen für die Ausgabedatei	150
VII Betrieb und Überwachung des Natural Web I/O Interface Server	151
23 Betrieb des Natural Web I/O Interface Server	153
Starten des Natural Web I/O Interface Server	154
Beenden des Natural Web I/O Interface Server	155
Überwachen (Monitoring) des Natural Web I/O Interface Server	155
Laufzeit-Trace-Funktion	156

Trace-Filter	158
24 Monitor Client NATMOPI	159
Über den Monitor Client NATMOPI	160
Syntax der Kommando-Schnittstelle	160
Verfügbare Kommando-Optionen	160
Monitor-Kommandos	161
Verzeichnis-Kommandos	161
Kommando-Beispiele	162
25 HTML Monitor Client	165
Über den HTML Monitor Client	166
Voraussetzungen für den HTML Monitor-Client	166
Server-Liste	166
Server-Monitor	168

Vorwort

Diese Dokumentation ist in die folgenden Themenbereiche gegliedert:

Einführung	Was ist das Natural Web I/O Interface?
Vorstellung des Natural Web I/O Interface Server CICS-Adapter	Spezielle Informationen für den Einsatz des Natural Web I/O Interface Server in einer CICS-Umgebung unter z/OS.
Vorstellung des Natural Web I/O Interface Server IMS-Adapter	Spezielle Informationen für den Einsatz des Natural Web I/O Interface Server in einer IMS TM-Umgebung unter z/OS.
Installation und Konfiguration des Natural Web I/O Interface Server	Wie Sie den Natural Web I/O Interface Server unter z/OS installieren und konfigurieren. Anmerkung: Informationen zur Installation und Konfiguration in einer Linux- oder Windows-Umgebung finden Sie in der Natural-Dokumentation für die entsprechende Plattform. Die folgenden Themen gelten zusätzlich, wenn Sie den Natural Web I/O Interface Server mittels entsprechendem Adapter in einer CICS-Umgebung oder einer IMS TM-Umgebung einsetzen wollen: ■ Installation des Natural Web I/O Interface Server CICS Adapter ■ Konfiguration des Natural Web I/O Interface Server CICS Adapter ■ Installation des Natural Web I/O Interface Server IMS Adapter ■ Konfiguration des Natural Web I/O Interface Server IMS Adapter
Installation des Natural Web I/O Interface Client	Wie Sie den Natural Web I/O Interface Client auf einem Anwendungsserver oder in einem Servlet Container installieren, damit er mit dem Natural Web I/O Interface Server verwendet werden kann.
Konfiguration des Natural Web I/O Interface Client	Wie Sie die Informationen definieren, die auf der Anmeldeseite (Logon Page) erscheinen sollen.
Betrieb und Überwachung des Natural Web I/O Interface Server	Wie Sie den Natural Web I/O Interface Server in einer z/OS-Umgebung betreiben und wie Sie ihn mit dem Monitor Client NATMOPI oder dem HTML Monitor Client überwachen.

1 Über diese Dokumentation

■ Dokumentationskonventionen	2
■ Online-Informationen und Support	2
■ Datenschutz	3

Dokumentationskonventionen

Konvention	Beschreibung
Fettschrift	>Kennzeichnet Elemente auf einem Bildschirm.
Nichtproportionale Schrift	Kennzeichnet Namen und Orte von Diensten im Format <i>Ordner.Unterordner.Dienst</i> , Programmierschnittstellen (APIs), Namen von Klassen, Methoden und Properties in Java.
<i>Kursivschrift</i>	Kennzeichnet: Variablen, für die Sie situations- oder umgebungsspezifische Werte angeben müssen. Neue Begriffe, wenn sie erstmals im Text auftreten. Verweise auf andere Dokumentationsquellen.
Nichtproportionale Schrift	Kennzeichnet: Text, den Sie eingeben müssen. Meldungen, die vom System angezeigt werden. Programmcode.
{ }	Zeigt eine Reihe von Auswahlmöglichkeiten an, von denen Sie eine auswählen müssen. Geben Sie nur die innerhalb der geschweiften Klammern vorhandenen Informationen ein. Geben Sie nicht die Klammersymbole { } ein.
	Trennt zwei sich gegenseitig ausschließende Auswahlmöglichkeiten in einer Syntaxzeile voneinander ab. Geben Sie eine der Auswahlmöglichkeiten ein. Geben Sie nicht das Symbol ein.
[]	Zeigt eine oder mehrere Optionen an. Geben Sie nur die innerhalb der eckigen Klammern vorhandenen Informationen ein. Geben Sie nicht die Klammersymbole [] ein.
...	Zeigt an, dass Sie mehrere Auswahlmöglichkeiten desselben Typs eingeben können. Geben Sie nur die Informationen ein. Geben Sie nicht die drei Auslassungspunkte (...) ein.

Online-Informationen und Support

Produktdokumentation

Sie finden die Produktdokumentation auf unserer Dokumentationswebsite unter <https://documentation.softwareag.com>.

Zusätzlich können Sie auch über <https://www.softwareag.cloud> auf die Dokumentation für die Cloud-Produkte zugreifen. Navigieren Sie zum gewünschten Produkt und gehen Sie dann, je nach Produkt, zu „Developer Center“, „User Center“ oder „Documentation“.

Produktschulungen

Sie finden hilfreiches Produktschulungsmaterial auf unserem Lernportal unter <https://knowledge.softwareag.com>.

Tech Community

Auf der Website unserer Tech Community unter <https://techcommunity.softwareag.com> können Sie mit Experten der Software AG zusammenarbeiten. Von hier aus können Sie zum Beispiel:

- Unsere umfangreiche Wissensdatenbank durchsuchen.
- In unseren Diskussionsforen Fragen stellen und Antworten finden.
- Die neuesten Nachrichten und Ankündigungen der Software AG lesen.
- Unsere Communities erkunden.
- Unsere öffentlichen Repositories auf GitHub and Docker unter <https://github.com/softwareag> und <https://hub.docker.com/publishers/softwareag> besuchen und weitere Ressourcen der Software AG entdecken.

Produktsupport

Support für die Produkte der Software AG steht lizenzierten Kunden über unser Empower-Portal unter <https://empower.softwareag.com> zur Verfügung. Für viele Dienstleistungen auf diesem Portal benötigen Sie ein Konto. Wenn Sie noch keines haben, dann können Sie es unter <https://empower.softwareag.com/register> beantragen. Sobald Sie ein Konto haben, können Sie zum Beispiel:

- Produkte, Aktualisierungen und Programmkorrekturen herunterladen.
- Das Knowledge Center nach technischen Informationen und Tipps durchsuchen.
- Frühwarnungen und kritische Alarmer abonnieren.
- Supportfälle öffnen und aktualisieren.
- Anfragen für neue Produktmerkmale einreichen.

Datenschutz

Die Produkte der Software AG stellen Funktionen zur Verarbeitung von personenbezogenen Daten gemäß der Datenschutz-Grundverordnung (DSGVO) der Europäischen Union zur Verfügung. Gegebenenfalls sind in der betreffenden Systemverwaltungsdokumentation entsprechende Schritte dokumentiert.

I Einführung

2 Einführung

■ Wozu dient das Natural Web I/O Interface?	8
■ Komponenten des Natural Web I/O Interface	8
■ Ausführen einer Natural-Anwendung in einem Webbrowser	9
■ Client-Server-Kompatibilität	10
■ Terminologie	11
■ Unterschiede in einer SPoD-Entwicklungsumgebung	11
■ Einschränkungen bei der Verwendung des Natural Web I/O Interface mit Natural-Anwendungen	12
■ Unterschiede zwischen dem Natural Web I/O Interface Client und einer Terminal-Emulation	13

Dieses Kapitel beschreibt den Zweck und die Funktionen des Natural Web I/O Interface.



Anmerkung: In dieser Einführung wird schwerpunktmäßig beschrieben, wie das Natural Web I/O Interface in einer Laufzeitumgebung (Produktionsumgebung) funktioniert. Der Abschnitt [Unterschiede in einer SPoD-Entwicklungsumgebung](#) erläutert kurz die spezielle Version, die in einer SPoD-Entwicklungsumgebung verwendet wird.

Wozu dient das Natural Web I/O Interface?

Das Natural Web I/O Interface dient der Ausführung von Natural-Anwendungen in einem Webbrowser. Es unterstützt die folgenden Funktionen in vollem Umfang:

- Die Anzeige und Eingabe von Unicode-Zeichen. Siehe *Behandlung von Unicode-Ein-/Ausgaben in Natural-Anwendungen* in der *Unicode- und Codepage-Unterstützung-Dokumentation*.
- Rich-Internet-Anwendungen, die mit Natural for Ajax entwickelt wurden.

Komponenten des Natural Web I/O Interface

Die Natural Web I/O Interface besteht aus einem Server und einem Client.

- [Server](#)
- [Client](#)

Server

Der Natural Web I/O Interface Server ermöglicht es Ihnen, einen Browser als Ein-/Ausgabe-Vorrichtung für Natural-Anwendungen zu verwenden. Der Server führt die Benutzerauthentifizierung durch, eröffnet die Natural-Sitzung und verwaltet die Eingabe/Ausgabe zwischen Natural und dem Client. Der Natural Web I/O Interface Server wird auf demselben Rechner installiert wie die Natural-Anwendung.

Client

Der Client übernimmt die Kommunikation zwischen dem Webbrowser des Benutzers und dem Natural Web I/O Interface Server. Er wandelt die Ausgaben der Natural-Anwendung in Webseiten um und gibt die Benutzereingaben an Natural zurück.

Es werden zwei Arten von Clients unterstützt:

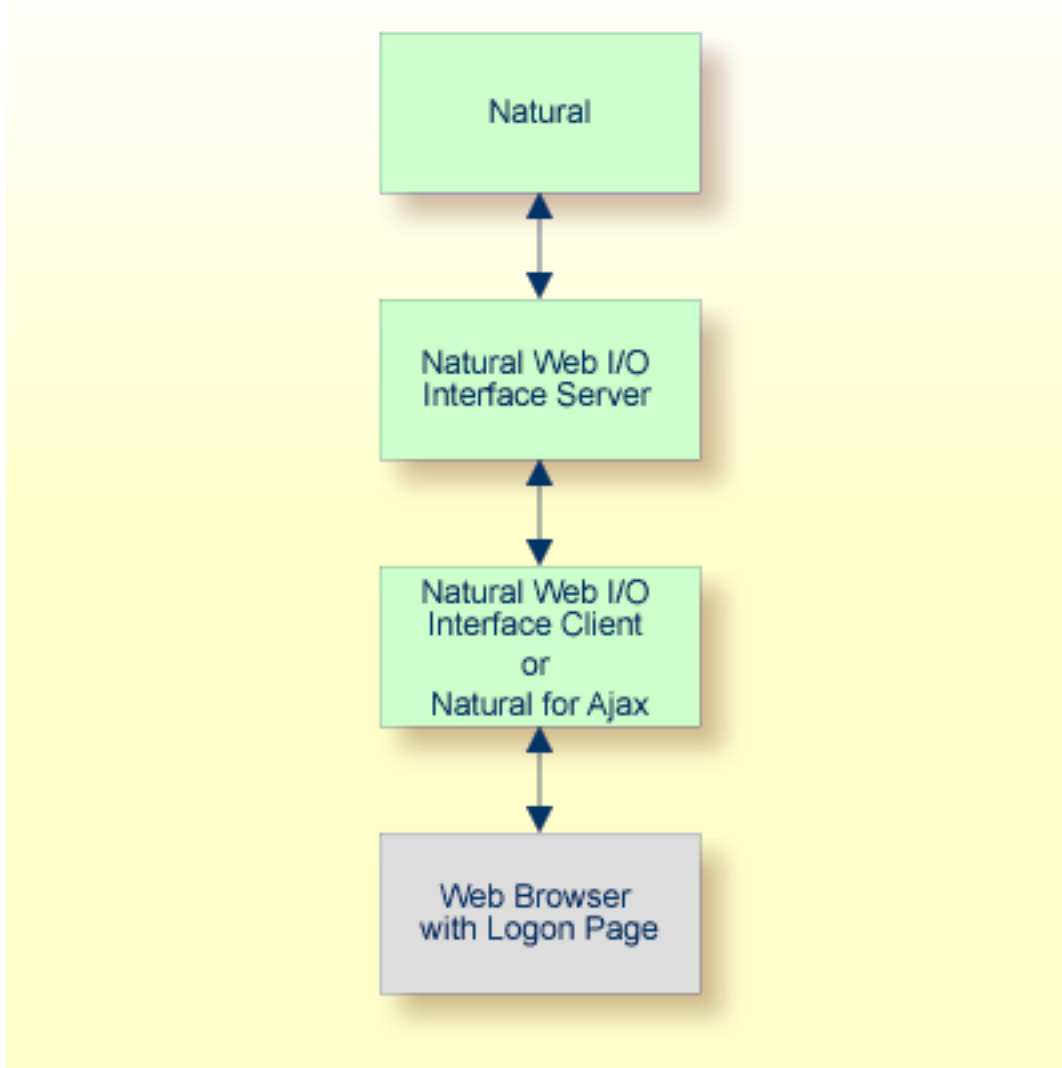
- Natural Web I/O Interface Client für die Anzeige zeichenbasierter Anwendungen im Webbrowser. Masken (Maps) mit GUI Controls werden in diesem Fall nicht unterstützt.

- Natural for Ajax für die Anzeige von Rich-Internet-Anwendungen im Webbrowser. Weitere Informationen zu dieser Art von Client finden Sie in der *Natural for Ajax*-Dokumentation.

Der Client wird auf einem Web-/Anwendungsserver installiert. Dies kann auf einem beliebigen Rechner im Netzwerk erfolgen.

Ausführen einer Natural-Anwendung in einem Webbrowser

Das Natural Web I/O Interface empfängt Daten aus einer Natural-Anwendung und liefert Webseiten an den Webbrowser des Benutzers. Dies wird in der folgenden Grafik veranschaulicht:



Die Kommunikationsschritte zum Ausführen einer Natural-Anwendung im Webbrowser sind:

1. Der Benutzer gibt die Adresse (URL) einer Anmeldeseite in den Webbrowser ein. Der Client zeigt dann die Anmeldeseite im Webbrowser an.



Anmerkung: Informationen zum Aufrufen und Konfigurieren der Anmeldeseite finden Sie *Konfiguration des Client*.

2. Der Benutzer gibt alle erforderlichen Informationen zum Starten einer Natural-Anwendung auf der Anmeldeseite ein. Diese Informationen werden an den Client gesendet.
3. Der Client fordert den Natural Web I/O Interface Server auf, die gewünschte Natural-Anwendung für diesen Benutzer zu starten.
4. Der Natural Web I/O Interface Server überprüft die angegebene Benutzerkennung (User ID) und das Passwort, erstellt eine Natural-Sitzung für den Benutzer und startet die Natural-Anwendung.
5. Die Natural-Anwendung gibt den ersten Anwendungsbildschirm zurück, der dann über den Natural Web I/O Interface Server an den Client und schließlich als Webseite an den Webbrowser übertragen wird.

Es werden verschiedene Webbrowser unterstützt. Beachten Sie, dass Cookies und JavaScript im Webbrowser aktiviert sein müssen. Eine Liste der derzeit unterstützten Webbrowser finden Sie unter *Browser-Voraussetzungen* in der Dokumentation für den von Ihnen verwendeten Client-Typ:

- **Natural Web I/O Interface Client**

Siehe *Browser-Voraussetzungen* in dieser Dokumentation.

- **Natural for Ajax**

Siehe *Browser Prerequisites* in der *Natural for Ajax*-Dokumentation.

Client-Server-Kompatibilität

Es gelten die folgenden Regeln:

- Der Natural Web I/O Interface Server kann mit jedem Client arbeiten, der über dieselbe oder eine höhere Protokollversion verfügt.

Wenn der Server feststellt, dass der Client eine niedrigere Version als die Server-Version verwendet, antwortet der Server, dass die Client-Version zu alt ist, und die Verbindung wird geschlossen.

- Der Client kann mit jedem Server arbeiten, der über dieselbe oder eine niedrigere Protokollversion verfügt.

Wenn der Client feststellt, dass der Server eine Version verwendet, die niedriger ist als die Client-Version, wechselt der Client zur Server-Version. Neue Client-Funktionalität wird in diesem Fall jedoch nicht unterstützt.

- Der Natural Web I/O Interface Server muss die gleiche Protokollversion haben wie der Natural-Prozess, der vom Server gestartet wird. Wenn Natural feststellt, dass der Server eine andere Protokollversion verwendet, wird eine Fehlermeldung an den Benutzer gesendet und die Verbindung beendet.

Terminologie

Daemon, Server, Service

Auf den verschiedenen Natural-Plattformen, für die das Natural Web I/O Interface unterstützt wird, werden unterschiedliche Techniken für die Implementierung des Serverteils des Natural Web I/O Interface verwendet:

- Auf Natural for Linux ist er als Daemon implementiert.
- Auf Natural for Windows ist er als Dienst (Service) implementiert.
- Unter z/OS ist er als Server implementiert.

In dieser Dokumentation wird der allgemeine Begriff „Server“ für alle verschiedenen Implementierungsarten verwendet.

SMARTS

SMARTS ist ein Akronym für „Software AG Multi-Architecture Runtime System“. Es stellt eine Laufzeitschicht dar, die es ermöglicht, POSIX-ähnliche Anwendungen auf Mainframe-Betriebssystemen auszuführen. Die Produkte der Software AG kommunizieren mit dem Betriebssystem über die SMARTS-Schicht.

Unterschiede in einer SPoD-Entwicklungsumgebung

In den vorangegangenen Abschnitten dieser Einführung wurde beschrieben, wie das Natural Web I/O Interface in einer Laufzeit-(Produktions-)Umgebung funktioniert. In diesem Abschnitt werden die Unterschiede in einer SPoD-Entwicklungsumgebung kurz erläutert.

Für die Arbeit in einer Remote-Entwicklungsumgebung mit Natural for Windows (SPoD) wird eine spezielle Version des Natural Web I/O Interface verwendet. In diesem Fall ist das Natural Web I/O Interface eine integrierte Komponente, die keine separate Installation erfordert. Der Server ist Teil des Natural Development Server (NDV), der Client ist Teil von Natural Studio. Anders als in der Laufzeitumgebung wird der Bildschirm nicht in einem Browser, sondern in einem normalen Fenster angezeigt. Rich-GUI-Seiten, die mit Natural for Ajax erstellt wurden, werden in der Entwicklungsumgebung nicht unterstützt.

Es ist wichtig, dass die Ein-/Ausgabe über das Natural Web I/O Interface auf dem Natural-Host aktiviert ist. Andernfalls kann das Natural Web I/O Interface nicht aufgerufen werden. Siehe auch

Behandlung von Unicode-Ein-/Ausgaben in Natural-Anwendungen in der Unicode- und Codepage-Unterstützung-Dokumentation.

Einschränkungen bei der Verwendung des Natural Web I/O Interface mit Natural-Anwendungen

Bei der Verwendung des Natural Web I/O Interface mit Natural-Anwendungen auf Linux-, z/OS- oder Windows-Hosts gibt es einige Einschränkungen.



Anmerkung: Der Begriff „Anwendung“ bezieht sich auf Anwendungssoftware. Er bezieht sich nicht auf Systemsoftware oder Software für die Entwicklung.

Es gelten die folgenden Einschränkungen:

■ GUI Controls

GUI Controls werden nicht unterstützt: Dialoge, Schaltflächen, Optionsfelder, Listenfelder, Listenansichten, Kontrollkästchen usw. Das Natural Web I/O Interface unterstützt nur Natural-Anwendungen, die ohne GUI-Controls entwickelt wurden.

■ Dateiübertragung

Die Dateiübertragung (z. B. mit dem Natural-Statement `DOWNLOAD`) wird vom Natural Web I/O Interface nicht unterstützt.

■ Laufzeitfehler

Diese Einschränkung gilt für ältere Natural-Versionen unter Linux und Windows. Ab Version 6.3.3 gilt diese Einschränkung nicht mehr.

Laufzeitfehler in Natural-Anwendungen werden vom Natural Web I/O Interface nicht behandelt. Dies führt zu einem Verlust der Sitzung. Sie können den Fehler mit Hilfe der Natural-Systemvariablen `*ERROR-TA` umgehen.

Beispiel einer Natural-Fehlertransaktion:

```
DEFINE DATA  
LOCAL  
1 ERR_INFO  
  2 ERR_NR(N5)  
  2 ERR_LINE(N4)  
  2 ERR_STAT(A1)  
  2 ERR_PNAM(A8)  
  2 ERR_LEVEL(N2)  
END-DEFINE  
INPUT ERR_INFO  
DISPLAY ERR_INFO  
TERMINATE  
END
```



Anmerkung: Einige der folgenden Einschränkungen funktionieren möglicherweise auf z/OS, sind jedoch für eine reale Anwendung nicht sinnvoll.

■ **Rückkehr zum Natural-Hauptbildschirm**

Sie können keine Natural-Anwendungen verwenden, die zum Natural-Hauptbildschirm zurückkehren. Dies führt immer zu einer falschen Bildschirmanzeige und zum Verlust der Sitzung.

■ **Natural-Editoren und -Dienstprogramme (Utilities)**

Sie können keine Natural-Dienstprogramme wie `SYSMAIN` oder `SYSDDM` und keine Editoren wie den Programmeditor verwenden. Dies führt immer zu einer falschen Bildschirmanzeige und zum Verlust der Sitzung.

■ **Natural-Systemkommandos**

Sie können keine Natural-Systemkommandos verwenden, zum Beispiel: `CATALL`, `FIND`, `GLOBALS`, `HELP`, `KEY`, `LIST`, `SCAN` oder `XREF`. Dies führt immer zu einer falschen Bildschirmanzeige und zum Verlust der Sitzung.

■ **Natural-Kommandos `SETUP` und `RETURN`**

Sie sollten die Natural-Kommandos `SETUP` und `RETURN` nicht verwenden, weil dies zum Verlust der Sitzung führen kann.

■ **Terminalkommandos**

Terminalkommandos werden nicht unterstützt. Sie funktionieren nicht, wenn sie über den Natural Web I/O Interface Client eingegeben werden.

■ **Natural-Systemvariable `*INIT-ID`**

Wenn Sie den Natural Web I/O Interface Client mit Natural-Anwendungen auf Linux-, z/OS- oder Windows-Hosts verwenden, wird die Natural-Systemvariable `*INIT-ID` nicht mit einem Wert für den Terminaltyp gefüllt.

Unter Linux und Windows enthält sie den Wert `notty`. Auf Mainframes enthält sie eine Sitzungskennung, die auf dem jeweiligen Server eindeutig ist.

Unterschiede zwischen dem Natural Web I/O Interface Client und einer Terminal-Emulation

Der Natural Web I/O Interface Client wird als HTML-Terminal-Emulator innerhalb eines Browser-Controls ausgeführt. Die Anzeige des Natural Web I/O Interface Client ähnelt der des regulären Terminals (Emulation), aber es gibt einige Unterschiede aufgrund der Browser-Funktionalität:

- Ein Doppelklick mit dem Mauszeiger auf ein beliebiges Feld simuliert die `ENTER`-Taste.
- Mit dem Mauszeiger kann der Cursor auf einem nur als Ausgabefeld fungierenden Feld positioniert werden, jedoch befindet sich die zurückgegebene Cursorposition (Systemvariable `*CURSOR`) immer auf dem ersten Zeichen im Feld.

- Es ist nicht möglich, den Cursor außerhalb des Bereichs der Eingabe- und Ausgabefelder zu positionieren.
- Der Cursor kann nur innerhalb eines Eingabefeldes mit den Pfeiltasten nach links und rechts bewegt werden. Andere Cursorbewegungen mit den anderen Pfeiltasten (z. B. innerhalb von Ausgabefeldern oder zum nächsten Eingabefeld oder vertikale Bewegungen) sind nicht möglich.
- Der Einfügemodus kann mit der `INSERT`-Taste ein- und ausgeschaltet werden.
- Bei Unicode-Zeichensätzen (Typ U, z.B. Chinesisch) kann ein Zeichen mehr Platz benötigen als ein gewöhnliches alphanumerisches Zeichen, da die Unicode-Zeichendarstellung proportional ist. Dies muss bei der Gestaltung der Anwendung berücksichtigt werden, da Natural auf Zeichen mit fester Breite basiert. Bei Eingabefeldern ist es möglich, innerhalb des Feldes zu blättern, aber bei Ausgabefeldern kann der Platz für die Anzeige der Unicode-Zeichen nicht ausreichen. Die Anzeigelänge für ein Feld kann über den Session-Parameter `DL` gesteuert werden.
- Der Vorauswahlmodus (Type-ahead mode) wird nicht unterstützt.
- Einfügen (Paste) im Überschreibmodus wird nicht unterstützt.
- Die Tastaturbelegung ist festgelegt; Tasten wie die rechte `STRG`-Taste und die `EINGABE`-Taste auf dem Ziffernblock sind nicht mehr definierbar.
- Die Bildschirmaktualisierung ist langsamer, da statt Aktualisierungen der gesamte Bildschirm gesendet wird.
- Das Attribut `blink` wird nicht unterstützt.
- Die Tasten `PF1` bis `PF12` werden durch die Tastenkombinationen `F1` bis `F12` simuliert.
- Die Tasten `PF13` bis `PF24` werden durch die Tastenkombinationen `SHIFT+F1` bis `SHIFT+F12` simuliert.
- Die Tasten `PF25` bis `PF36` werden durch die Tastenkombinationen `STRG+F1` bis `STRG+F12` simuliert.
- Die Tasten `PF37` bis `PF48` werden durch die Tastenkombinationen `ALT+F1` bis `ALT+F12` simuliert.
- Die Löschtaste wird durch `STRG+UMSCHALT+F4` simuliert.

Nur bei IBM z/OS

- Die Bildschirmgröße des Terminals wird durch den Natural-Profilparameter `TMODEL` gesteuert. Die Standardeinstellung ist `TMODEL=0` (24 Zeilen und 80 Spalten).
- Es gibt keine `ATTN`-Taste (Aufmerksamkeitsunterbrechung), keine `RESET`-Taste und keine `EEOF`-Taste (Löschen des Dateiendes).

Nur bei VT

Die E/A erfolgt im Blockmodus. Daher reagiert das Natural-Programm nur, wenn eine Funktionstaste gedrückt wird.

II Vorstellung des Natural Web I/O Interface Server CICS

Adapter

3 Vorstellung des Natural Web I/O Interface Server CICS

Adapter

■ Zweck des Natural Web I/O Interface Server CICS Adapter	18
■ CICS-Unterstützung	18
■ Zusammenwirken der Produkte	19

Dieses Dokument beschreibt den Zweck und die Funktionen des Natural Web I/O Interface Server CICS Adapter.

Zweck des Natural Web I/O Interface Server CICS Adapter

Der Natural Web I/O Interface Server CICS Adapter ist für einen Natural-Mainframe-Kontext konzipiert, wo er den Einsatz eines Natural Web I/O Interface Servers ermöglicht, der unter z/OS im Batch-Modus innerhalb einer CICS TP-Monitor-Umgebung läuft.

CICS-Unterstützung

Die CICS-Unterstützung ist im Frontend-Stub `NATRNWO` nicht implementiert. Für das Dispatching der Natural-Sitzungen in CICS läuft der Web I/O Interface Server weiterhin im Batch-Modus oder unter `SMARTS`. Er verwendet jedoch das Remote-Frontend `NATCSRFE`, das mit dem Natural Web I/O Interface Server geliefert wird, um die Natural-Sitzungen in CICS zu disponieren. Das heißt, je nach installiertem Frontend verteilt ein Server die Sitzungen lokal (`NCFNUC` für `SMARTS`, `NATMVS` für den Batch-Modus) oder remote (`NATCSRFE` für CICS).

`NATCSRFE` wiederum nimmt die Natural-Anfrage von `NATRNWO` entgegen und leitet sie über die CICS-Socket-Schnittstelle an eine konfigurierte CICS-Umgebung weiter. Innerhalb der CICS-Umgebung wird eine CICS-Natural-Transaktion gestartet, die die Natural-Anfrage verarbeitet und das Ergebnis zurückgibt. Es ist also nicht notwendig, den gesamten Web I/O Interface Server unter CICS auszuführen. Nur wenn Natural aufgefordert wird, die Natural-Anwendung auszuführen, wird die Kontrolle zur Ausführung an CICS übergeben.

Der Natural Web I/O Interface Server CICS Adapter besteht aus den folgenden Komponenten:

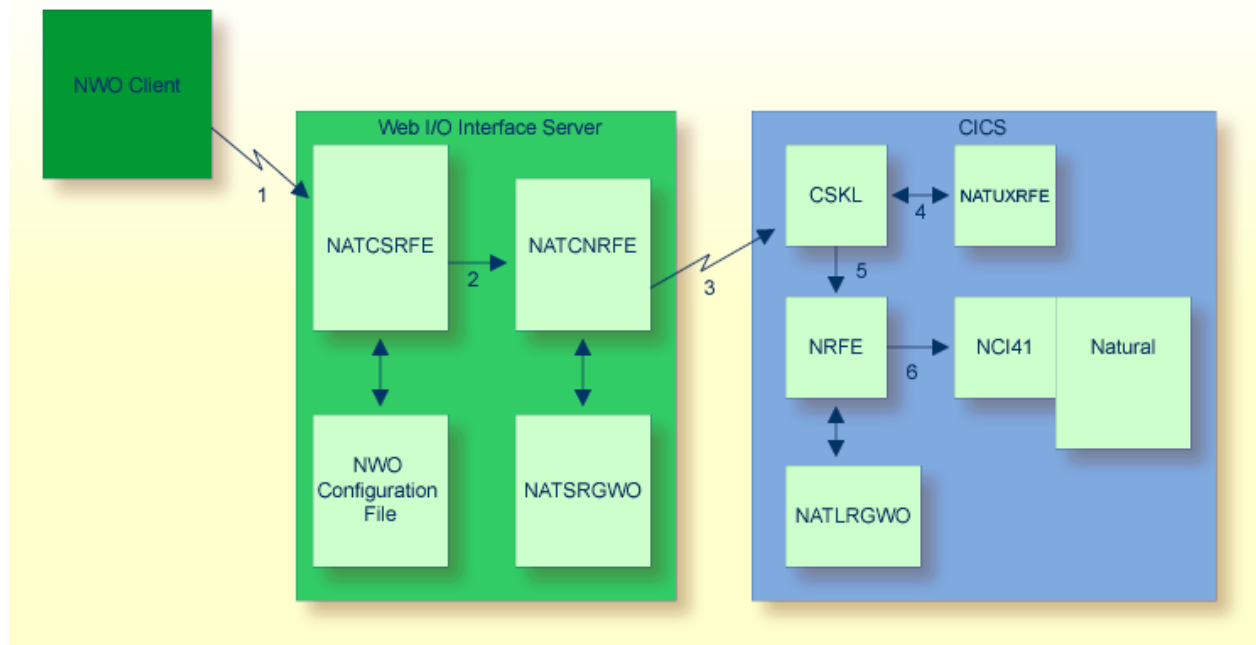
- `NATCSRFE`
Das Remote-Frontend, das vom Natural Web I/O Interface Server aufgerufen wird, um eine Natural-Anfrage zu disponieren. Es wird in den Adressraum des Web I/O Interface Server geladen.
- `NATCNRFE`
Das Gegenstück zu `NATCSRFE`. `NATCNRFE` läuft im CICS-Adressraum. Es wird von dem von IBM bereitgestellten Standard-Listener der CICS-Socket-Schnittstelle gestartet (siehe auch *TCP/IP V3R1 for MVS: CICS TCP/IP Socket Interface Guide*).
- `NATSRGW0/NATLRGW0`
Überträgt die für den Natural Web I/O Interface Server relevanten Daten zwischen dem Natural Web I/O Interface Server und der in CICS laufenden Natural-Sitzung. `NATSRGW0` muss in den Adressraum des Natural Web I/O Interface Server und `NATLRGW0` in den CICS-Adressraum geladen werden.

■ NATUXRFE

Dieser User-Exit holt sich die Client-Anmeldedaten vom Natural Web I/O Interface Server und authentifiziert sich dann mit einer CICS `VERIFY PASSWORD`-Anfrage. Wenn die Anfrage erfolgreich ist, startet der CICS-Listener die Natural Web I/O Interface-Transaktion unter dem Client-Konto (Impersonierung).

Zusammenwirken der Produkte

Die folgende Abbildung veranschaulicht die Interaktion zwischen dem Natural Web I/O Interface Server und der beteiligten CICS-Umgebung.



1. Der Web I/O Interface Client sendet eine Anfrage an den Natural Web I/O Interface Server unter Verwendung der Port-Nummer, die in der Natural Web I/O Interface Server-Konfigurationsvariablen `PORT_NUMBER` angegeben ist.
2. Der Natural Web I/O Interface Server dispatched die Natural-Sitzung über das Natural-Frontend, das Sie mit der Natural Web I/O Interface Server-Konfigurationsvariable `FRONTEND_NAME` angegeben haben. Geben Sie `NATCSRFE` an, um den Natural Web I/O Interface Server CICS Adapter zu verwenden.
3. `NATCSRFE` sendet die Anfrage an den Host/Port, der mit der Natural Web I/O Interface Server-Konfigurationsvariable `RFE_CICS_TA_HOST / RFE_CICS_TA_PORT` angegeben wurde. Sie müssen den von CICS bereitgestellten Standard-Listener `CSKL` so konfigurieren, dass er an diesem Port empfängt.

4. Wenn der Natural Web I/O Interface Server so konfiguriert ist, dass er Remote-Impersonierung durchführt (`SECURITY_MODE=IMPERSONATE/IMPERSONATE_REMOTE`), wird NATUXRFE zur Authentifizierung des Clients aufgerufen. Wenn die Authentifizierung erfolgreich ist, startet CSKL die CICS-Transaktion NRFE unter dem Konto des Clients (impersoniert).
5. CSKL startet die CICS-Transaktion, die Sie mit dem Server-Konfigurationsparameter `RFE_CICS_TA_NAME` (in diesem Beispiel NRFE) des Natural Web I/O Interface angegeben haben. Diese Transaktion muss so definiert sein, dass sie das Programm NATCNRFE verwendet.
6. NATCNRFE versendet schließlich die Natural-Sitzung über das Natural-CICS-Frontend, das Sie mit dem Natural Web I/O Interface Server-Konfigurationsparameter `RFE_CICS_FE_NAME` angegeben haben.

III

Vorstellung des Natural Web I/O Interface Server IMS

Adapter

4 Vorstellung des Natural Web I/O Interface Server IMS Adapter

■ Zweck des Natural Web I/O Interface Server IMS Adapter	24
■ IMS TM-Unterstützung	24
■ Zusammenwirken der Produkte	25

Dieses Dokument beschreibt den Zweck und die Funktionen des Natural Web I/O Interface Server IMS Adapter.

Zweck des Natural Web I/O Interface Server IMS Adapter

Der Natural Web I/O Interface Server IMS Adapter ist für einen Natural-Mainframe-Kontext konzipiert, in dem er den Einsatz eines Natural Web I/O Interface Server ermöglicht, der unter z/OS im Batch-Modus innerhalb einer IMS TM-Umgebung läuft.

IMS TM-Unterstützung

Die IMS TM-Unterstützung ist im Frontend-Stub `NATRNWO` nicht implementiert. Für die Abwicklung der Natural-Sitzungen in IMS TM läuft der Natural Web I/O Interface Server weiterhin im Batch-Modus. Er verwendet jedoch das Remote-Frontend `NATISRFE`, das mit dem Natural Web I/O Interface Server ausgeliefert wird, um die Natural-Sitzungen in IMS TM zu disponieren. Das heißt, je nach installiertem Frontend verteilt ein Server die Sitzungen lokal (`NATMVS` für den Batch-Modus) oder remote (`NATISRFE` für IMS TM).

`NATISRFE` wiederum nimmt die Natural-Anfrage von `NATRNWO` entgegen und überträgt sie über den von der IMS-Installation bereitgestellten BPM-Listener an eine konfigurierte IMS TM-Umgebung. Der IMS-Listener startet in einer dedizierten MPP-Region eine nicht konversationelle Natural-Transaktion, die die Natural-Anfrage verarbeitet und das Ergebnis zurückgibt. Es ist also nicht notwendig, den gesamten Natural Web I/O Interface Server unter IMS TM auszuführen. Nur kleine Arbeitseinheiten (Natural-Anfragen wie `save source` oder `get library list`) werden zur Ausführung an IMS TM übertragen.

Der Natural Web I/O Interface Server IMS Adapter besteht aus den folgenden Komponenten:

- `NATISRFE`
Das Remote-Frontend, das vom Natural Web I/O Interface Server aufgerufen wird, um eine Natural-Anfrage zu disponieren. Es wird in den Adressraum des Web I/O Interface Server geladen.
- `NATINRFE`
Das Gegenstück zu `NATISRFE`. `NATINRFE` läuft in einer MPP-Region. Es wird durch den von IBM bereitgestellten IMS-Listener gestartet (siehe auch *z/OS Communications Server IP IMS Socket Guide*).
- `NATSRGWO/NATLRGWO`
Überträgt die für den Natural Web I/O Interface Server-relevanten Daten zwischen dem Natural Web I/O Interface Server und der Natural-Sitzung, die in IMS TM läuft. `NATSRGWO` muss in den Adressraum des Natural Web I/O Interface Servers geladen werden, `NATLRGWO` in die MPP-Region.

Zusammenwirken der Produkte

Die folgende Beschreibung beschreibt die Interaktion zwischen dem Natural Web I/O Interface Server und der beteiligten IMS TM-Umgebung.

1. Der Natural Web I/O Interface Client sendet eine Anfrage an den Natural Web I/O Interface Server unter Verwendung der Port-Nummer, die in der Konfigurationsvariablen `PORT_NUMBER` des Natural Web I/O Interface Servers angegeben ist.
2. Der Natural Web I/O Interface Server dispatched die Natural-Sitzung über das Natural-Frontend, das Sie mit der Natural Web I/O Interface Server-Konfigurationsvariable `FRONTEND_NAME` angegeben haben. Geben Sie `NATISRFE` an, um den Natural Web I/O Interface Server IMS Adapter zu verwenden.
3. `NATISRFE` sendet die Anfrage an den Host/Port, der mit der Natural Web I/O Interface Server-Konfigurationsvariablen `RFE_IMS_TA_HOST` / `RFE_IMS_TA_PORT` angegeben wurde. Sie müssen den von IBM bereitgestellten BMP-Listener so konfigurieren, dass er an diesem Port empfängt.
4. Der BMP-Listener startet die IMS TM-Transaktion, die Sie mit dem Konfigurationsparameter `RFE_IMS_TA_NAME` des Natural Web I/O Interface Server angegeben haben. Diese Transaktion muss in der Konfiguration des IMS-Listeners angegeben werden.
5. Die Server-Transaktion ruft zunächst die Transaktionsinitialisierungsnachricht ab. Diese enthält die notwendigen Informationen für den `TAKESOCKET` und übergibt sie an `NATINRFE`. `NATINRFE` kümmert sich um den `TAKESOCKET` und baut die TCP/IP-Konversation mit `NATISRFE` auf.
6. `NATINRFE` disponiert schließlich das Natural-IMS-Frontend, das eine Natural-Server-Session einrichtet.

IV

Installation und Konfiguration des Natural Web I/O Interface Server

Konzept und Struktur des Natural Web I/O Interface Server
Voraussetzungen
Installation des Natural Web I/O Interface Server unter z/OS
Konfiguration des Natural Web I/O Interface Server
Die folgenden Themen gelten zusätzlich, wenn Sie den Natural Web I/O Interface Server mittels entsprechendem Adapter in einer CICS-Umgebung oder einer IMS TM-Umgebung einsetzen wollen:
Installation des Natural Web I/O Interface Server CICS Adapter
Konfiguration des Natural Web I/O Interface Server CICS Adapter
Installation des Natural Web I/O Interface Server IMS Adapter
Konfiguration des Natural Web I/O Interface Server IMS Adapter

5

Konzept und Struktur des Natural Web I/O Interface Server

■ Konzept des Web I/O Interface Server	30
■ Frontend Stub NATRNWO	30
■ Frontend	32
■ Server-Monitoring	32

Dieses Dokument beschreibt das Konzept und die Struktur des Servers für das Natural Web I/O Interface, der für den Einsatz auf z/OS konzipiert ist.

Konzept des Web I/O Interface Server

Ein Natural Web I/O Interface Server ist eine Mehrbenutzer- und Multitasking-Anwendung. Er kann als Host für Natural-Sitzungen für mehrere Benutzer dienen und deren Anwendungen gleichzeitig ausführen.

Das Konzept basiert auf dem Serverized Natural Runtime System. Seine Architektur besteht aus einem Server Frontend Stub (Web I/O Interface Server Stub NATRNWO), der das Natural-Frontend nutzt, um Natural-Sitzungen zu disponieren und Anwendungen innerhalb dieser Sitzung auszuführen.

Die Architektur des Natural Web I/O Interface Server besteht im Wesentlichen aus:

- **Frontend Stub**

Der Stub NATRNWO wird gestartet, um einen Natural Web I/O Interface Server zu initialisieren. Er wartet als Listener auf ankommende Verbindungsanforderungen und startet eine Natural-Session zur Ausführung der Anwendung.

- **Frontend**

Das Frontend wird (zusammen mit dem Natural-Laufzeitsystem) vom Frontend Stub für die Sitzungsinitialisierung/-beendigung, die Ausführung der Anwendung und das Ein- und Auspeichern (Roll-in/Roll-out) der Sitzung aufgerufen.

- **Server Monitor**

Eine Monitor-Task ermöglicht es dem Administrator, die Server-Aktivitäten zu kontrollieren, bestimmte Benutzersitzungen abubrechen oder den gesamten Server zu beenden usw.

Frontend Stub NATRNWO

Der Mehrbenutzer- und Multitasking-Frontend-Stub NATRNWO wird zur Initialisierung eines Natural Web I/O Interface Server gestartet.

Die folgenden Themen werden behandelt:

- [Stub-Beschreibung](#)

■ Verwendete Natural-Systemvariablen

Stub-Beschreibung

Die Task (T_{Main}), die die Initialisierung des Servers ausführt, ist im Wesentlichen der Haupt-Listener, der auf eingehende Anfragen vom Web I/O Interface Client wartet. Er verfügt über ein Sitzungsverzeichnis, um mehrere Clients (Benutzer) und ihre entsprechenden Remote Natural Sessions zu verwalten. T_{Main} hat die Aufgabe, alle eingehenden Anfragen anzunehmen und zu disponieren, d.h. sie auf andere Subtasks (T_{Work}) zu verteilen. Dieser Vorgang läuft folgendermaßen ab:

- Zunächst stellt eine vom Benutzer auf der Client-Seite (über die Schaltfläche **Login** des Web I/O Interface Client) erstellte Server-Verbindung eine Verbindung zu T_{Main} her, um eine Verbindung aufzubauen.
- Anschließend trägt T_{Main} den Client in sein Sitzungsverzeichnis ein, fügt eine neue T_{Work} -Subtask hinzu und übergibt die Verbindung an T_{Work} .
- T_{Work} initialisiert eine neue Natural-Sitzung und startet das angegebene Natural-Anwendungsprogramm.
- Während die Anwendung E/A-Anforderungen ausführt, fängt T_{Work} die Ein-/Ausgabe-Daten ab und leitet sie an den Web I/O Interface Client zur Verarbeitung der E/A weiter. Die E/A-Antwort wird an den Server zurückgesendet und der Server setzt die Anwendung fort.
- Wenn die Anwendung beendet wird (den **NEXT**-Modus erreicht), beendet T_{Work} die Natural-Sitzung und trennt die Verbindung zum Web I/O Interface Client.

Das heißt, jeder Client besitzt eine Subtask T_{Work} auf dem Natural Web I/O Interface Server. Diese Subtask führt eine Natural-Sitzung (und innerhalb der Natural-Sitzung eine Natural-Anwendung) aus und bleibt so lange aktiv, wie die Anwendung läuft.

Verwendete Natural-Systemvariablen

Innerhalb einer Natural Web I/O Interface Server-Sitzung werden die folgenden Natural-Systemvariablen verwendet:

- `*TPSYS` enthält `SERVSTUB`,
- `*DEVICE` enthält `BROWSER`,
- `*SERVER-TYPE` enthält `WEBIO`.

Frontend

Unter z/OS ist das für einen Natural Web I/O Interface Server erforderliche Natural-Frontend ein Natural-Batch-Frontend-Treiber, der LE-fähig sein sollte (Option LE370=YES). Einzelheiten finden Sie im Beispiel-Installationsjob.

Unter z/OS wird das Frontend (zusammen mit dem Natural-Laufzeitsystem) vom Frontend-Stub für die Sitzungsinitialisierung/-beendigung, die Ausführung von Anfragen und das Ein-/Ausspeichern der Sitzung (Session Roll-in/Roll-out) aufgerufen.

Bei dem Natural-Frontend, das für die Ausführung von Natural-Sitzungen unter der Kontrolle von CICS erforderlich ist, handelt es sich um das Natural-Remote-Frontend NATCSRFE, das mit dem Natural Web I/O Interface Server ausgeliefert wird. Weitere Informationen finden Sie unter *Natural Web I/O Interface Server CICS Adapter*.

Server-Monitoring

Damit der Administrator den Status des Natural Web I/O Interface Server überwachen kann, wird eine Monitor-Task bereitgestellt, die beim Start des Servers automatisch initialisiert wird. Mit den Monitor-Kommandos kann der Administrator die Serveraktivitäten kontrollieren, bestimmte Benutzersitzungen abbrechen, den gesamten Server beenden usw. Siehe *Betrieb des Web I/O Interface Server*.

6 Voraussetzungen

- Allgemeine Voraussetzungen für die Installation des Web I/O Interface Server 34
- Voraussetzungen für den Web I/O Interface Server für z/OS 34

Im Folgenden werden die Voraussetzungen beschrieben, die für die Installation eines Natural Web I/O Interface Server auf einem z/OS-Mainframe gelten.

Allgemeine Voraussetzungen für die Installation des Web I/O Interface Server

Es gelten die folgenden allgemeinen Voraussetzungen:

- Es muss die aktuell relevante Version von Natural for z/OS installiert sein.
- Die ICU for Adabas & Natural (ICS) muss mit dem Web I/O Interface Server installiert werden. Siehe *ICU for Adabas & Natural (ICS)* in der *Unicode und Codepage-Unterstützung-Dokumentation*.



Anmerkung: Weitere Informationen finden Sie unter *Von Natural unterstützte Produkt-Versionen* in der aktuellen *Natural-Freigabemitteilung (Release Notes)*.

Voraussetzungen für den Web I/O Interface Server für z/OS

Zusätzlich zu den oben beschriebenen allgemeinen Voraussetzungen gelten die folgenden betriebssystemspezifischen Voraussetzungen:

- z/OS muss installiert sein.
- Um die Bildung von Endlosschleifen in Benutzerprogrammen zu verhindern, die unter dem Web I/O Interface laufen, müssen Sie einen angemessenen Wert für den Natural-Profilparameter MT (Maximale CPU-Zeit) angeben.

7 Installation des Natural Web I/O Interface Server unter z/OS

■ Voraussetzungen für z/OS	36
■ Inhalt des Distributionsmediums des Web I/O Interface Server	36
■ Installationsschritte	36

Dieses Dokument beschreibt, wie Sie einen Server für das Natural Web Interface (Produktcode NWO) unter dem Betriebssystem z/OS installieren.

Die Installation des Web I/O Interface Server erfolgt über Installationsjobs. Beispieljobs sind im Dataset `NW0vrs.JOBS` enthalten (wobei *vrs* für die jeweilige Produktversion steht) und tragen das Präfix `NW0` oder werden von System Maintenance Aid (SMA) erzeugt.

Voraussetzungen für z/OS

Einzelheiten finden Sie im Abschnitt [Voraussetzungen](#).

Inhalt des Distributionsmediums des Web I/O Interface Server

Der Installationsdatenträger enthält die in der folgenden Tabelle aufgeführten Datasets. Die Reihenfolge der Datasets und die Anzahl der benötigten Bibliotheksblöcke entnehmen Sie bitte dem *Product Delivery Report*, der dem Installationsmedium beiliegt.

Dataset-Name	Inhalt
<code>NW0vrs.OBJS</code>	Enthält die Objektmodule des Servers.
<code>NW0vrs.JOBS</code>	Beispiel-Installationsjobs.

In diesem Dokument steht in den Dataset-Namen die Notation *vrs* oder *vr* für die jeweilige Produktversion (siehe auch Version im *Glossar*).

Installationsschritte

Step 1: Zuweisung der Web I/O Interface Server LOAD Library

(Job I008, Step 8410)

Step 2: Erstellen einer Web I/O Interface Server-Konfigurationsdatei und einer Beispiel-Clist

(Job I009 / Step 9410, 9420, 9430)

Step 9410 erstellt ein NWOCONFIG-Beispiel für den Batch-Server.

Step 9420 erstellt eine Beispiel-Clist zum Anpingen und Beenden eines Web I/O Interface Server.

Step 9430 erzeugt einen Beispiel-Batch-Job zum Anpingen und Beenden eines Web I/O Interface Server.

Die folgenden Parameter der Konfigurationsdatei müssen definiert werden. Siehe [Konfigurieren des Natural Web I/O Interface Server](#). Bei den anderen Parametern können die Standardwerte verwendet werden:

FRONTEND_NAME	Geben Sie den Namen des Web I/O Interface Server-Frontend-Moduls an, das Sie in einem der folgenden Schritte erzeugen werden.
PORT_NUMBER	Geben Sie die TCP/IP-Port-Nummer an, unter der der Server verbunden werden kann.

Step 3: Verlinken der Objektmodule in die NWO Load Library

(Job I054, Step 9410)

Die NWO-Objektmodule müssen mit den notwendigen Laufzeit-Erweiterungen Ihrer Batch-Installationen zu ausführbaren Lademodulen verlinkt werden.



Anmerkung: Das Modul NATCNRFE gilt für zwei verschiedene Produkte, den Natural Web I/O Interface Server (NWO) und den Natural Development Server (NDV). Wenn Sie also den NDV bereits installiert haben, könnte das Modul NATCNRFE schon vorhanden sein. Es spielt jedoch keine Rolle, ob Sie NATCNRFE mit NWO neu installieren, da das resultierende Modul bei beiden Installationen dasselbe ist.

Siehe Beispieljob NW0I054 im Dataset NW0vrs.JOBS.

Step 4: Erstellen des Natural-Parametermoduls und des NWO-Server-Frontend-Moduls

(Job I060, Steps 9410, 9420, 9430)

- Job I060, Step 9410 startet das Batch-Programm IEBUPDATE, um das Parametermodul NWOPARM zu speichern.
- Job I060, Step 9420 assembliert und verlinkt das Parametermodul NWOPARM.
- Job I060, Step 9430 verlinkt das NWO-Server-Frontend-Modul. Das reentrante ADALINK-Modul ADALNKR muss verwendet werden.



Anmerkung: Wenn Sie Natural-Subprodukte für den IBM-Datenbankzugriff installiert haben, z. B. Natural for Db2, Natural for VSAM, müssen Sie den Link-Job so anpassen, dass er `ATRCSS` aus `SYS1.CSSLIB` einschließt, um die Nutzung der `SYNCPPOINT-/ROLLBACK-Funktionen` zu ermöglichen.

Step 5: Erstellen der Server-Start-JCL

(Job I200, Step 9415)

Beschreibung siehe [Konfigurieren des Natural Web I/O Interface Server](#). Siehe Beispiel-Member `NWOSTART` im Dataset `NW0vrs.JOBS`.

Step 9415 erzeugt eine Startprozedur für den Batch-Server.

Beispiel:

```
// PROC SRV=SAGNWO //NWO EXEC PGM=NATRNWO, //  
REGION=4000K,TIME=1440,PARM='POSIX(ON),TRAP(ON,NOSPIE)&SRV' //STEPLIB DD  
DISP=SHR,DSN=NW0vrs.LOAD // DD DISP=SHR,DSN=SAGLIB.SMALOAD //SYSUDUMP DD  
SYSOUT=X //CEEDUMP DD SYSOUT=X //CMPRINT DD SYSOUT=X //STGCONFIG DD DISP=SHR, //  
DSN=NW0.CONFIG(&SRV) //STGTRACE DD SYSOUT=X //STGSTDO DD SYSOUT=X //STGSTDE  
DD SYSOUT=X //SYSOUT DD SYSOUT=X
```



Anmerkung: Das Konto des Web I/O Interface Server muss in den z/OS Linux System Services (OE Segment) definiert sein. Wenn das Serverkonto nicht definiert ist, wird der Server mit U4093 und der Systemmeldung CEE5101C in der Trace-Datei beendet.

Step 6: Web I/O Interface Clients müssen in Natural Security definiert sein

Wenn Natural Security (NSC) installiert ist:

- Die initiale Web I/O Interface-Benutzerkennung (Standardkennung ist `STARGATE`) muss in Natural Security mit einer gültigen Default Library definiert sein. Siehe auch den Web I/O Interface Server-Konfigurationsparameter `INITIAL_USERID`. Alternativ können Sie den Natural-Profilparameter `AUTO=OFF` (Automatische Anmeldung) für das Web I/O Interface definieren.
- Jede Client-Benutzerkennung muss in Natural Security definiert werden.

Wenn die initiale Web I/O Interface-Benutzerkennung nicht definiert ist, bricht die Web I/O Interface Server-Initialisierung mit dem Fehler `NAT0856` ab.

Wenn ein Web I/O Interface Client nicht definiert ist, gibt die Serververbindung einen NSC-Fehler zurück.

Wenn Sie von einem Web I/O Interface Client aus eine Verbindung zum Server herstellen, müssen Sie sicherstellen, dass für den in Natural Security definierten Benutzer eine Default Library oder eine Private Library definiert ist. Andernfalls tritt die Fehlermeldung `NAT0815` auf.

Step 7: Web I/O Interface Clients müssen im Server-Host definiert werden

Wenn Sie den Web I/O Interface Server so konfigurieren, dass er ein externes Security-System verwendet (siehe Web I/O Interface Server-Konfigurationsparameter [SECURITY_MODE](#)), müssen die Web I/O Interface Clients in dem externen Security-System definiert werden.

8 Konfigurieren des Natural Web I/O Interface Server

■ Konfigurationserfordernisse für z/OS	42
■ Web I/O Interface Server-Konfigurationsdatei für z/OS	50
■ Konfigurationsparameter für den Web I/O Interface Server	51
■ Beispiel für eine Web I/O Interface Server-Konfigurationsdatei	62
■ Web I/O Interface Server Datasets für z/OS	62
■ Web I/O Interface Server User Exits	63

Dieses Dokument beschreibt, wie Sie einen Natural Web I/O Interface Server konfigurieren.

Konfigurationserfordernisse für z/OS

Die folgenden Themen werden behandelt:

- [Parametereinstellungen für Language Environment \(LE\)](#)
- [Externe Security-Konfiguration](#)
- [SSL TLS-Unterstützung](#)
- [Konfigurieren der Port-Sharing-Komponente](#)

Parametereinstellungen für Language Environment (LE)

Ein Natural Web I/O Interface Server benötigt die folgende z/OS-LE-Parameterkonfiguration:

Parameter	Definition
POSIX(ON)	Ermöglicht einem Natural Web I/O Interface Server den Zugriff auf die POSIX-Funktionalität von z/OS. Wenn Sie einen Natural Web I/O Interface Server mit POSIX(OFF) starten, wird dieser sofort mit einem User Abend U4093 und der Systemmeldung EDC5167 beendet. IBM liefert die Standardeinstellung OFF.
TRAP(ON,NOSPIE)	Definiert, wie Abends in der LE/370-Umgebung behandelt werden:
	ON Aktiviert den Language Environment Condition Handler.
	NOSPIE Legt fest, dass die Language Environment Programmunterbrechungen und -Abends über einen ESTAE behandelt, d. h., der Natural Abend Handler erhält die Kontrolle zur Behandlung von Programmunterbrechungen und Abends.
	Wenn Sie TRAP(ON,NOSPIE) nicht angeben, funktioniert die Natural Abend-Behandlung nicht ordnungsgemäß. IBM liefert die Standardeinstellung (ON,SPIE).
TERMTHDACT(UADUMP)	Legt den Umfang der Informationen fest, die im Falle eines Abends ausgegeben werden. Die Option UADUMP generiert einen Sprachumgebungs-CEEDUMP und einen System-Dump des Benutzeradressraums. Der CEEDUMP enthält keine für Natural relevanten Speicherbereiche. IBM liefert die Standardeinstellung (TRACE).
ENVAR(TZ=...)	Mit der Option ENVAR können Sie Linux-Umgebungsvariablen festlegen. Die einzige für den Natural Web I/O Interface Server relevante Umgebungsvariable ist TZ (Zeitzone). Mit dieser Variablen können Sie den Zeitstempel in der Trace-Datei des Natural Web I/O Interface Server an Ihre Ortszeit anpassen. Beispiel:

Parameter	Definition
	ENVAR(TZ=CET-1DST) CET
	- 1 Stunde Sommerzeit

Sie können die z/OS-Sprachumgebungsparameter festlegen:

- Mit dem Parameter `PARM`, der in der `EXEC`-Karte des Startjobs des Natural Web I/O Interface Server angegeben ist. Die Länge der Optionen ist durch die maximale Länge des Parameters `PARM` begrenzt.
- Assemblieren Sie ein LE/370-Laufzeitoptionsmodul `CEEUOPT` und verlinken Sie es mit dem Lademodul des Natural Web I/O Interface Server.

Externe Security-Konfiguration

Wenn Sie den Web I/O Interface Server so konfigurieren, dass er die Web I/O Interface Clients auf dem Server impersoniert (Web I/O Interface Server-Konfigurationsparameter `SECURITY_MODE=IMPERSONATE` oder `IMPERSONATE_LOCAL`), muss der Web I/O Interface Server „programmgesteuert“ ausgeführt werden. Unter RACF sind die folgenden Definitionen für den Web I/O Interface Server erforderlich:

- Die Ressource `BPX.SERVER` muss definiert sein, und das Web I/O Interface Server-Konto muss `READ`-Zugriff auf diese Ressource haben.
- Die in der Startjobdefinition des Web I/O Interface Server definierten `LOAD`-Datasets müssen für die Programmklasse `**` definiert sein.

```
ralt program ** addmem('natural load library') uacc(read) ralt program **
addmem('NWO load library'//NOPADCHK) uacc(read) ralt program ** addmem('user
load library'//NOPADCHK) uacc(read) ←
```

- `SETR WHEN(PROGRAM) REFRESH`

Darüber hinaus muss jeder Client, der eine Verbindung zum Server herstellt, in RACF definiert sein und die Berechtigung zur Nutzung der z/OS Linux System Services erhalten haben.

SSL TLS-Unterstützung

- [SSL/TLS über AT-TLS](#)
- [Verwaltung von Zertifikaten unter z/OS](#)
- [Verwendung von RACF Key Rings](#)
- [Verwendung von Key Databases](#)
- [Wie konfiguriert man TCP/IP für AT-TLS?](#)
- [Wie verifiziert man die AT-TLS-Konfiguration?](#)
- [Häufig gestellte Fragen](#)
- [Generierung eines Natural Web I/O Interface Server-Zertifikats](#)

SSL/TLS über AT-TLS

Die SSL/TLS-Unterstützung für den Natural Web I/O Interface Server basiert auf der z/OS Communication Server-Komponente AT-TLS (Application Transparent-Transport Layer Security).

AT-TLS bietet SSL/TLS-Verschlüsselung als konfigurierbaren Dienst für Socket-Anwendungen. Es wird als zusätzliche Schicht über dem TCP/IP-Protokoll-Stack realisiert, die die SSL/TLS-Funktionalität in nahezu oder sogar vollständig transparentem Modus für Socket-Anwendungen nutzt. AT-TLS bietet drei Betriebsmodi. Siehe *z/OS Communications Server, IP Programmer's Guide and Reference*. Version 1, Release 9, Chapter 15, IBM manual SC31-8787-09.

Diese Modi sind:

■ Basic

Die Sockets-Anwendung läuft ohne Änderungen im transparenten Modus, ohne zu erkennen, dass eine verschlüsselte Kommunikation über AT-TLS stattfindet. So können ältere Anwendungen ohne Änderung des Quellcodes im gesicherten Modus ausgeführt werden.

■ Aware

Die Anwendung erkennt, dass sie im gesicherten Modus läuft, und kann TLS-Statusinformationen abfragen.

■ Controlling

Die Sockets-Anwendung erkennt AT-TLS und steuert die Verwendung der AT-TLS-Verschlüsselungsdienste selbst. Das bedeutet, dass die Anwendung zwischen gesicherter und ungesicherter Kommunikation wechseln kann.

Beim Natural Web I/O Interface Server wird für seine SSL/TLS-Implementierung der Basic-Modus verwendet. Das heißt, ein als SSL/TLS-Server konfigurierter Server lehnt Anfragen von ungesicherten Clients ab.

Verwaltung von Zertifikaten unter z/OS

Zertifikate, die mit AT-TLS verwendet werden sollen, können unter z/OS auf zwei Arten verwaltet werden. Sie werden entweder in RACF Key Rings oder in Key Databases gespeichert, die sich im z/OS Linux-Dateisystem befinden. Welches dieser Verfahren tatsächlich angewendet wird, ist im AT-TLS Policy Agent Configuration File für den z/OS TCP/IP-Stack definiert, der vom Natural HTTPS Client verwendet wird.

IBM liefert mit jedem z/OS-System eine Reihe gängiger CA-Root-Zertifikate mit. Wenn Key Rings zur Speicherung von Server-Zertifikaten verwendet werden sollen, müssen diese Root-Zertifikate vom Systemadministrator manuell in die Key Rings importiert werden. Wenn IBM neuere Ersatzzertifikate für abgelaufene Root-Zertifikate liefert, müssen alle betroffenen Key Rings entsprechend aktualisiert werden.

Im Gegensatz zu Key Rings enthalten Key Databases den aktuellen Satz von Root-Zertifikaten automatisch, nachdem diese neu erstellt wurden. Die Notwendigkeit, immer den neuesten Satz von Root-Zertifikaten zu pflegen, gilt jedoch auch für die Alternative der Key Database.

Verwendung von RACF Key Rings

In RACF werden digitale Zertifikate in sogenannten Key Rings gespeichert. Das RACF-Kommando `RACDCERT` wird zum Erstellen und Verwalten von Key Rings und Zertifikaten verwendet, die in diesen Key Rings enthalten sind.

Siehe *z/OS Security Server RACF Security Administrator's Guide*, IBM manual SA22-7683-11, and *z/OS Security Server RACF Command Language Reference*, IBM manual SA22-7687-11.

Verwendung von Key Databases

Alternativ zu RACF können Zertifikate in Key Databases gespeichert werden, die sich im Dateisystem der z/OS Linux Services befinden. Für die Erstellung und Pflege von Key Databases muss das Dienstprogramm `GSKKYM` verwendet werden.

Siehe *z/OS Cryptographic Services PKI Services Guide and Reference*, IBM manual SA22-7693-10.

Wie konfiguriert man TCP/IP für AT-TLS?

Gehen Sie folgendermaßen vor:

1. Setzen Sie in der TCP/IP-Konfigurationsdatei die Option `TTLS` in der `TCPCONFIG`-Anweisung.
2. Konfigurieren und starten Sie den AT-TLS Policy Agent. Dieser Agent wird von TCP/IP bei jeder neuen TCP-Verbindung aufgerufen, um zu überprüfen, ob es sich um eine SSL/TLS-Verbindung handelt.
3. Erstellen Sie die Policy Agent-Datei mit den AT-TLS-Regeln. Die Policy Agent-Datei enthält die Regeln, die festlegen, welche Verbindung SSL/TLS ist.

Siehe auch *z/OS Communications Server: IP Configuration Guide, Chapter 18 Application Transparent Transport Layer Security (AT-TLS) data protection*.

Die Beispiel-Policy-Agent-Datei definiert den Server mit dem Jobnamen, der mit `NWODEV` beginnt und als Listener an Port 4843 empfängt, um SSL/TLS zu verwenden.

In dem Beispiel wird die Zertifikatsdatenbank in der HFS-Datei `/u/admin/CERT.kdb` erwartet.

```
TTLSRule ConnRule01~1 { LocalAddrSetRef addr1 RemoteAddrSetRef
addr1 LocalPortRangeRef portR1 RemotePortRangeRef portR2 Jobname NWODEV*
Direction Inbound Priority 255 TTLSGroupActionRef gAct1~NWO_Server
TTLSEnvironmentActionRef eAct1~NWO_Server TTLSConnectionActionRef
cAct1~NWO_Server } TTLSGroupAction gAct1~NWO_Server { TTLSEnabled On }
TTLSEnvironmentAction eAct1~NWO_Server { HandshakeRole Server
EnvironmentUserInstance 0 TTLSKeyringParmsRef keyR1 } TTLSConnectionAction
cAct1~NWO_Server { HandshakeRole Server TTLSCipherParmsRef
cipher1~AT-TLS__Silver TTLSConnectionAdvancedParmsRef cAdv1~NWO_Server }
TTLSConnectionAdvancedParms cAdv1~NWO_Server { CertificateLabel NDV_TEST_CERT }
TTLSKeyringParms keyR1 { Keyring /u/admin/CERT.kdb KeyringStashFile
/u/admin/CERT.sth } TTLSCipherParms cipher1~AT-TLS__Silver { V3CipherSuites
TLS_RSA_WITH_DES_CBC_SHA V3CipherSuites TLS_RSA_WITH_3DES_EDE_CBC_SHA
V3CipherSuites TLS_RSA_WITH_AES_128_CBC_SHA } IpAddrSet addr1 { Prefix
0.0.0.0/0 } PortRange portR1 { Port 4843 } PortRange portR2 { Port 1024-65535
}
```

Wie verifiziert man die AT-TLS-Konfiguration?

Überprüfen Sie die Ausgabe des Policy-Agent-Jobs JESMSG LG auf:

```
EZZ8771I PAGENT CONFIG POLICY PROCESSING COMPLETE FOR
<your TCP/IP address space>: TTLS
```

Diese Meldung zeigt eine erfolgreiche Initialisierung an.

Überprüfen Sie die Ausgabe des Policy-Agent-Jobs JESMSG LG auf:

```
EZZ8438I PAGENT
POLICY DEFINITIONS CONTAIN ERRORS FOR <your TCP/IP address space>:
TTLS
```

Diese Meldung weist auf Fehler in der Konfigurationsdatei hin. Weitere Informationen finden Sie in der Datei `syslog.log`.

Gilt die Konfigurationsregel für den Server?

Versuchen Sie, eine Verbindung zum Server herzustellen, und überprüfen Sie `syslog.log` auf Folgendes:

```
EZD1281I TTLS Map CONNID: 00018BED LOCAL:
10.20.91.61..4843 REMOTE: 10.20.160.47..4889 JOBNAME: NWDEV42 USERID: NWOSRV
TYPE: InBound STATUS: Enabled RULE: ConnRule01~1 ACTIONS: gAct1
eAct1~NWO_Server cAct1~NWO_Server
```

Der obige Eintrag zeigt an, dass die Verbindung zu Port 4843 SSL/TLS-fähig ist.

Häufig gestellte Fragen

Gibt es weitere Informationen zur Fehlerbehebung?

Siehe auch *z/OS V1R8.0 Comm Svr: IP Diagnosis Guide: 3.23, Chapter 29 Diagnosing Application Transparent Transport Layer Security (AT-TLS)*

Wie schaltet man die P-Agent-Ablaufverfolgung (Trace) ein?

Siehe *Comm Svr: IP Configuration Reference, Chapter 20 Syslog daemon and Comm Svr: IP Configuration Guide, Chapter 1.5.1 Configuring the syslog daemon (syslogd)*

Fehler beim Verbindungsaufbau

Suchen Sie den Rückgabecode RC und den entsprechenden GSK_-Funktionsnamen im P-Agent-Trace.

Siehe *System SSL Programming* und suchen Sie den RC im Chapter 12.1 *SSL Function Return Codes*.

Beispiel-Trace mit trace=255:

```
EZD1281I TTLS
Map CONNID: 00002909 LOCAL: 10.20.91.61..1751 REMOTE: 10.20.91.117..443
JOBNAME: KSP USERID: KSP TYPE: OutBound STATUS: A EZD1283I TTLS Event GRPID:
00000003 ENVID: 00000000 CONNID: 00002909 RC: 0 Connection Init EZD1282I TTLS
Start GRPID: 00000003 ENVID: 00000002 CONNID: 00002909 Initial Handshake
ACTIONS: gAct1 eAct1 AllUsersAsClient HS-Client EZD1284I TTLS Flow GRPID:
00000003 ENVID: 00000002 CONNID: 00002909 RC: 0 Call GSK_SECURE_SOCKET_OPEN -
7EE4F718 EZD1284I TTLS Flow GRPID: 00000003 ENVID: 00000002 CONNID: 00002909
RC: 0 Set GSK_SESSION_TYPE - CLIENT EZD1284I TTLS Flow GRPID: 00000003 ENVID:
00000002 CONNID: 00002909 RC: 0 Set GSK_V3_CIPHER_SPECS - 090A2F EZD1284I TTLS
Flow GRPID: 00000003 ENVID: 00000002 CONNID: 00002909 RC: 0 Set GSK_FD -
00002909 EZD1284I TTLS Flow GRPID: 00000003 ENVID: 00000002 CONNID: 00002909
RC: 0 Set GSK_USER_DATA - 7EEE9B50 EZD1284I TTLS Flow GRPID: 00000003 ENVID:
00000002 CONNID: 00002909 RC: 435 Call GSK_SECURE_SOCKET_INIT - 7EE4F718
EZD1283I TTLS Event GRPID: 00000003 ENVID: 00000002 CONNID: 00002909 RC: 435
Initial Handshake 00000000 7EEE8118 EZD1286I TTLS Error GRPID: 00000003 ENVID:
00000002 CONNID: 00002909 JOBNAME: KSP USERID: KSP RULE: ConnRule01 RC: 435
Initial Handshake EZD1283I TTLS Event GRPID: 00000003 ENVID: 00000002 CONNID:
00002909 RC: 0 Connection Close 00000000 7EEE8118 ↵
```

Generierung eines Natural Web I/O Interface Server-Zertifikats

Unter z/OS können SSL/TLS-Zertifikate mit dem Linux-System Services-Dienstprogramm `GSKKYM`AN erstellt werden. Die folgenden Schritte müssen ausgeführt werden, um ein neues Zertifikat zu erstellen, das für die SSL/TLS-gesicherte Kommunikation zwischen dem Natural Web I/O Interface Client und dem Server verwendet werden soll:

1. Starten Sie eine Shell Session aus TSO oder verbinden Sie sich über Telnet mit der z/OS Linux Shell.
2. Starten Sie `GSKKYM`AN.
3. Erstellen Sie eine neue Key Database oder öffnen Sie eine vorhandene.
4. Erstellen Sie ein selbstsigniertes Zertifikat, beispielsweise vom Typ „User or server certificate with 2048-bit RSA key“ (Benutzer- oder Serverzertifikat mit 2048-Bit-RSA Key).
5. Exportieren Sie das Zertifikat in eine (HFS-)Datei. Wählen Sie Base64 ASN.1 DER als Exportformat.

Diese generierte Datei kann nun mit FTP im Übertragungsformat ASCII auf den/die Natural Web I/O Interface Client(s) kopiert werden. Auf der Client-Seite sollte die empfangene Datei mit der Dateiendung `.CER` gespeichert werden. Das Zertifikat kann nun vom Natural Web I/O Interface Client verwendet werden.

Wenn Zertifikate in einem RACF Key Ring gespeichert sind, muss das generierte Zertifikat mit dem Kommando `RADCERT` in den entsprechenden Key Ring importiert werden.

Zertifikate, die auf einer anderen Plattform, z.B. auf einem Windows-PC, erstellt wurden, können ebenfalls in einen RACF Key Ring oder in eine Key Database importiert werden.

Ausführliche Informationen zur Verwendung des Dienstprogramms `GSKKYM`AN finden Sie in der Dokumentation zum IBM Communications Server, z. B. in den folgenden Handbüchern:

z/OS Communications Server IP Configuration Guide Version 1 Release 2 (IBM manual SC31-8775-01)
oder

z/OS Communications Server Cryptographic Services System Secure Sockets Layer Programming (IBM manual SC24-5901-04).

Für die Generierung von Zertifikaten unter Windows steht auf mehreren Websites ein kostenloses Dienstprogramm namens Ikeyman zum Download bereit. Ikeyman ist ebenfalls ein IBM-Produkt und bildet die Funktionalität von `GSKKYM`AN auf der Windows-Plattform ab.

Konfigurieren der Port-Sharing-Komponente

Die TCP/IP-Port-Sharing-Komponente des z/OS TCP/IP-Kommunikationssubsystems ermöglicht es mehreren Listnern, auf derselben Kombination aus Port und IP-Adresse empfangsbereit zu sein. Um dies im TCP/IP-Stack zu aktivieren, müssen Sie das Schlüsselwort `SHAREPORT` oder `SHAREPORTWLM` zur PORT-Profilanweisung hinzufügen, die den TCP-Port für die Serverinstanzen reserviert. Ausführliche Informationen finden Sie unter *PORT Statement* im Kapitel *TCP/IP profile (PROFILE.TCPIP) and Configuration Statements* im IBM Manual *z/OS Communications Server: IP Configuration Reference*.

Sie können den NWO-Server wie im Abschnitt [Starten des Natural Web I/O Interface Server](#) beschrieben starten. Sie müssen den verwendeten Port mit dem Parameter `PORT_NUMBER` in der Konfigurationsdatei des NWO-Servers definieren. Wenn die Port-Sharing-Komponente aktiviert ist, kann die Started Task mehrfach ausgeführt werden, um mehrere Instanzen eines NWO-Servers zu starten, die am selben Port empfangsbereit sind.



Anmerkung: Alle Server müssen mit derselben Konfigurationsdatei gestartet werden.

- [Serverkennung \(Server-ID\)](#)
- [Zuverlässigkeit](#)
- [Planung](#)
- [TRACE, STDOUT und STDERR](#)

Serverkennung (Server-ID)

Da die Serverkennung zur Identifizierung eines Servers im Serververzeichnis verwendet wird, müssen Sie eine eindeutige Serverkennung festlegen. Die Serverkennung wird als Parameter an den NWO-Server übergeben. Sie müssen die Parameter in der `PARM`-Karte der Started Task festlegen. Beispiel:

```
//NWOSRV EXEC PGM=NATRNWO,REGION=0M,TIME=20,
// PARM=('POSIX(ON) STACK(64K,32K,ANY,FREE) TRAP(ON,NOSPIE) ',
// 'TERMTHDACT(UADUMP),ENVAR(TZ=CET+0DST) /SERVERID')
```

Zuverlässigkeit

Wenn mehrere NWO-Server über einen Port empfangsbereit sind und ein Server ausfällt, kann die Arbeitslast weiterhin von den verbleibenden Servern bewältigt werden. Aus Sicht des Benutzers bleibt der vom NWO-Server angebotene Dienst verfügbar.

Planung

Wenn eine Anfrage für eine neue Verbindung eingeht, ist es Aufgabe des z/OS-Betriebssystems, die Anfrage an einen der empfangsbereiten Server weiterzuleiten. Die Verbindungen werden anhand einer gewichteten Round-Robin-Verteilung auf der Grundlage der Akzeptanzeffizienzan-teile (Servers' accept Efficiency Fractions/SEFs) der Server auf die verfügbaren Server verteilt. Sobald eine Sitzung gestartet wurde, wird sie auf diesem Server ausgeführt, bis sie beendet ist.

Sie können eine Sitzung in der Trace-Datei mit dem folgenden Code identifizieren:

```
Start of Session:
26 10:16:45 0000000E Worker task active, SID=DED9D087D842B606, client: USER

End of session:
26 10:21:53 0000000E Worker task disconnect, client: USER
26 10:21:53 0000000E Free LTCB...and finish
```

TRACE, STDOUT und STDERR

Jeder Server verfügt über eigene STGTRACE-, STGSTDO- und STDSTGE-Dateien, die Teil des Job-Pro-tokolls jedes Servers sind. Es ist möglich, die Ausgabe der Dateien in einen sequenziellen Dataset umzuleiten. Dazu muss jedoch sichergestellt werden, dass jedem Server ein eigener sequenzieller Dataset zugewiesen ist. Andernfalls kann die Ausgabe der Dateien keinem bestimmten Server zugewiesen werden.

Web I/O Interface Server-Konfigurationsdatei für z/OS

Einer Konfigurationsdatei wird der Name `<serverid>C` (z. B. NWOS1C) oder alternativ STGCONFIG zugewiesen.

Die Konfigurationsdatei ist eine Textdatei, die sich in einem Dataset oder in einer HFS-Datei unter z/OS befindet.

Die Konfigurationsdatei enthält die Server-Konfigurationsparameter in Form einer Schlüssel-wort=Wert-Syntax: `keyword=value`. Darüber hinaus kann sie Kommentare enthalten, deren Anfang mit einem Hash-Symbol (#) gekennzeichnet ist.

Siehe auch das unten gezeigte [Beispiel für eine Web I/O Interface Server-Konfigurationsdatei](#).

Konfigurationsparameter für den Web I/O Interface Server

Für den Web I/O Interface Server stehen folgende Konfigurationsparameter zur Verfügung:

- COMPATIBILITY_MODE
- FRONTEND_NAME
- FRONTEND_OPTIONS
- FRONTEND_PARAMETER
- HANDLE_ABEND
- HOST_NAME
- HTPMON_ADMIN_PSW
- HTPMON_PORT
- IGNORE_PRESENT_SERVER
- INITIAL_USERID
- KEEP_TCB
- O4I
- PASSWORD_MIXEDCASE
- PORT_NUMBER
- SECURITY_MODE
- SECURITY_SYSLOG
- SESSION_PARAMETER
- SESSION_TIMEOUT
- THREAD_NUMBER
- THREAD_SIZE
- TRACE_FILTER
- TRACE_LEVEL
- UPPERCASE_SYSTEMMESSAGES

COMPATIBILITY_MODE

Die aktuelle Version des Natural Web I/O Interface (NWO) setzt voraus, dass sie mit der neuesten Version von Natural ausgeführt wird. Bei der Ausführung mit älteren Natural-Versionen wird der Fehler NAT7729 NWO and Natural version do not agree (NWO- und Natural-Version stimmen nicht überein) ausgegeben. Dies liegt daran, dass NWO eine Teilmenge der Funktionen mit dem Client aushandeln muss, und zwar zu einem Zeitpunkt, zu dem die verwendete Natural-Version noch nicht bekannt ist.

Wenn Sie NWO mit einer früheren Version von Natural ausführen möchten, können Sie diesen Parameter auf YES setzen. Es wird empfohlen, diesen Parameter auf seinem Standardwert zu belassen, wenn Sie NWO mit der neuesten Version von Natural ausführen möchten, da in diesem Fall COMPATIBILITY_MODE=YES die Funktionalität unnötig einschränken würde.

Wert	Erklärung
YES	Auch ältere Versionen von Natural akzeptieren. Führt zu einer Einschränkung der mit der neuesten Version dokumentierten Funktionalität.
NO	Ausführung mit der neuesten Version von Natural voraussetzen. Dies ist der Standardwert.

Beispiel:

```
COMPATIBILITY_MODE=YES
```

FRONTEND_NAME

Dieser Konfigurationsparameter gibt den Namen des Natural-Frontends an, das zum Starten einer Natural-Sitzung verwendet werden soll. Das Frontend befindet sich in einem PDS-Member.

Wert	Erklärung
<i>frontend-name</i>	Zu verwendendes Natural-Frontend. Maximale Länge: 8 Zeichen.

Es gibt keinen Standardwert.

Beispiel:

```
FRONTEND_NAME=NATvrssv
```

FRONTEND_OPTIONS

Die Werte dieses Konfigurationsparameters können verwendet werden, um zusätzliche Optionen für das Natural-Frontend festzulegen.

Wert	Erklärung
01	Roll Server nicht verwenden. Dies ist der Standardwert.
02	Roll File bei Server-Beendigung bereinigen.
04	GTF-Trace schreiben.
08	ETRACE schreiben.
10	Automatische Beendigung des Frontends.
20	Konsoleninformationen schreiben.

Sie können die oben genannten Optionen nach Belieben kombinieren, indem Sie ihre Werte addieren und das Ergebnis wie im folgenden Beispiel einstellen.

Beispiel:

```
FRONTEND_OPTIONS=07
```

Die Einstellung in diesem Beispiel aktiviert die Optionen 01, 02 und 04.

FRONTEND_PARAMETER

Dieser optionale Konfigurationsparameter enthält zusätzliche Natural-Frontend-Parameter, wie in der Startup Parameter Area angegeben.

Wert	Erklärung
<i>parameter-name</i>	Sie können mehrere Parameter definieren. Jede Parameterspezifikation besteht aus einem Paar von 8-stelligen Zeichenketten, wobei die erste das Parameter-Schlüsselwort und die zweite den Parameterwert enthält, zum Beispiel: <pre>FRONTEND_PARAMETER = 'MSGCLASSX ' ↵</pre>

Es gibt keinen Standardwert.

Weitere Informationen finden Sie im Abschnitt *Natural im Batch-Modus* in der *Natural Operations-Dokumentation*.

Beispiel:

```
FRONTEND_PARAMETER='MSGCLASSX '
```

Die Einstellung in diesem Beispiel legt fest, dass X die Standardausgabeklasse für CMPRINT ist.

HANDLE_ABEND

Wenn außerhalb der Natural-Verarbeitung ein Abbruch in der Server-Verarbeitung auftritt, wird dieser Abbruch nicht von der Natural-Abbruchbehandlung abgefangen. Aus diesem Grund verfügt der Natural Web I/O Interface Server bei einem Abbruch über eine eigene Wiederherstellung.

Es wird empfohlen, diesen Parameter auf seinem Standardwert zu belassen, um die Auswirkungen eines Abbruchs auf einen einzelnen Benutzer zu beschränken. Wenn Sie den Wert dieses Parameters auf NO setzen, beendet jeder Fehler in der Server-Verarbeitung die gesamte Server-Verarbeitung. Das heißt, er wirkt sich auf alle Benutzer aus, die auf diesem Server arbeiten.

Wert	Erklärung
YES	Fehler in der Server-Verarbeitung abfangen, einen Snap-Dump schreiben und die Sitzung des betroffenen Benutzers abbrechen. Dies ist der Standardwert.
NO	Die Server-Abbruchbehandlung außer Kraft setzen.

Beispiel:

```
HANDLE_ABEND=NO
```

HOST_NAME

Dieser optionale Konfigurationsparameter ist nur erforderlich, wenn der Server-Host mehrere TCP/IP-Stacks unterstützt.

Wert	Erklärung
<i>host-name</i>	Wenn HOST_NAME angegeben ist, überwacht der Server den durch HOST_NAME angegebenen Stack, andernfalls überwacht er alle Stacks.

Es gibt keinen Standardwert.

Beispiel:

```
HOST_NAME=node1
```

oder

```
HOST_NAME=157.189.160.55
```

HTPMON_ADMIN_PSW

Dieser Konfigurationsparameter definiert das Passwort, das für bestimmte Überwachungsaktivitäten (z. B. Terminate Server) des HTML Monitor Client erforderlich ist.

Wert	Erklärung
beliebige Zeichenkette	Das Passwort, das für bestimmte Überwachungsaktivitäten im HTML Monitor Client eingegeben werden muss.

Es gibt keinen Standardwert.

Beispiel:

```
HTPMON_ADMIN_PSW=GHAU129B
```

HTPMON_PORT

Ein Web I/O Interface Server kann so konfiguriert werden, dass er eine HTTP-Monitor-Task hostet, die den in einem Webbrowser ausgeführten HTML Monitor Client unterstützt. Es ist nicht erforderlich, diese Monitor-Task auf jedem Server auszuführen. Mit einer einzigen Task können Sie alle Server überwachen, die auf einem Knoten ausgeführt werden.

Dieser Konfigurationsparameter definiert die TCP/IP-Portnummer, unter der die Server-Monitor-Task von einem Webbrowser aus verbunden werden kann.

Wert	Erklärung
1 - 65535	TCP/IP-Portnummer.

Es gibt keinen Standardwert.

Beispiel:

```
HTPMON_PORT=3141
```

IGNORE_PRESENT_SERVER

Dieser Konfigurationsparameter gilt in Verbindung mit dem Web I/O Interface Server CICS Adapter.

Ein Web I/O Interface Server weist eine sogenannte Serverumgebung zu, die die serverabhängigen gemeinsamen Ressourcen enthält.

Diese Umgebung ist für jeden Web I/O Interface Server einzigartig und bezieht sich auf den Servernamen. Wenn ein Web I/O Interface Server mit Web I/O Interface Server CICS Adapter abnormal beendet wird, kann dies zu einer blockierten Web I/O Interface Serverumgebung innerhalb der CICS-Region führen. Dies hat zur Folge, dass ein Neustart des Servers mit der Fehlermeldung NAT9913 fehlschlägt.

Wenn Sie einen Web I/O Interface Server mit `IGNORE_PRESENT_SERVER=YES` starten, kann dies einen bereits laufenden Server beschädigen, der denselben Servernamen und dieselbe CICS-Region verwendet.

Wert	Erklärung
YES	Die vorhandene CICS-Serverumgebung beenden.
NO	Die Serverinitialisierung abbrechen, wenn bereits eine CICS-Serverumgebung vorhanden ist. Dies ist der Standardwert.

Beispiel:

```
IGNORE_PRESENT_SERVER=YES
```

INITIAL_USERID

Bei der Serverinitialisierung erstellt der Natural Web I/O Interface Server eine temporäre Natural-Sitzung, um die Eigenschaften der installierten Natural-Umgebung abzurufen.

Dieser Konfigurationsparameter gibt die Benutzerkennung an, die für diese Natural-Sitzung verwendet werden soll.

Wert	Erklärung
<code>userid</code>	Der angegebene Wert darf nicht länger als 8 Zeichen sein, andernfalls wird er gekürzt.
STARGATE	Dies ist der Standardwert.

Beispiel:

```
INITIAL_USERID=NWOINITU
```

Siehe auch [Web I/O Interface-Clients müssen in Natural Security definiert sein](#).

KEEP_TCB

Standardmäßig beendet die Remote-Natural-Sitzung einer zugeordneten Umgebung ihre TCB, sobald Sie den Fokus innerhalb von Natural Studio auf eine andere zugeordnete Umgebung umschalten. Wenn Sie den Fokus wieder zurückschalten, wird die Remote-Sitzung mit einer anderen TCB ausgeführt.

Die maximale Anzahl aktiver TCBs entspricht der Anzahl der verbundenen Clients.

Der Konfigurationsparameter `KEEP_TCB` legt fest, ob die Remote-Natural-Sitzung während ihrer gesamten Laufdauer denselben TCB verwenden soll. Dies ist erforderlich, wenn Sie Adabas verwenden und der Adabas-Parameter `ADANAME` auf `ADAUSER` gesetzt ist oder wenn Sie auf Db2 zugreifen möchten. Es kann auch erforderlich sein, wenn Sie auf 3GL-Programme zugreifen, die für aufeinanderfolgende Aufrufe unter demselben TCB ausgeführt werden müssen.

Wert	Erklärung
YES	Die Remote-Natural-Sitzung verwendet während ihrer gesamten Dauer dieselbe TCB.
NO	Dies ist der Standardwert.

Beispiel:

```
KEEP_TCB=YES
```

04I



Anmerkung: Dieser Parameter ist nur bei Natural Web I/O Interface Version 8.3 oder höher verfügbar.

Mit diesem Parameter können Sie Serverdaten für Optimize for Infrastructure erfassen.

Wert	Erklärung
YES	Serverdaten für Optimize for Infrastructure erfassen.
NO	Dies ist der Standardwert.

PASSWORD_MIXEDCASE

Mit diesem Parameter können Sie festlegen, ob die im Verbindungsdialog angegebenen Passwörter in Großbuchstaben umgesetzt werden sollen oder nicht.

Dieser Parameter gilt nur bei

SECURITY_MODE=IMPERSONATE, IMPERSONATE_LOCAL **oder** IMPERSONATE_REMOTE.

Wert	Erklärung
YES	Passwörter bleiben in Groß- und Kleinschreibung.
NO	Passwörter werden in Großbuchstaben umgesetzt. Dies ist die Standardeinstellung.

Beispiel:

```
PASSWORD_MIXEDCASE=YES
```

PORT_NUMBER

Dieser Konfigurationsparameter definiert die TCP/IP-Portnummer, unter der der Server verbunden werden kann.

Wert	Erklärung
1 - 65535	TCP/IP-Portnummer

Es gibt keinen Standardwert.

Beispiel:

PORT_NUMBER=3140

SECURITY_MODE

Der Natural Web I/O Interface Server bietet ein Sicherheitskonzept, das auch die Ressourcen des Betriebssystems abdeckt. Die Anmeldedaten des Clients werden vom betriebssystemabhängigen Security-System validiert, und die Client-Anfrage wird unter den Kontodaten des Clients ausgeführt.

Mit dem Parameter `SECURITY_MODE` können Sie festlegen, auf welcher Ebene (im Batch-Modus z/OS oder unter CICS) Sie die Aktivitäten eines Web I/O Interface Client impersonieren möchten.

Wert	Erklärung
IMPERSONATE_LOCAL	Die Impersonierung erfolgt innerhalb der Natural Web I/O Interface Server-Umgebung. Wenn die Sitzung in einer Remote-TP-Umgebung (z. B. in CICS unter Verwendung des NWO CICS Adapters) verteilt wird, wird sie weiterhin anonym ausgeführt. Der Client muss im Security-System des Web I/O Interface Server definiert sein. Es ist nicht erforderlich, den Client in einer Remote-TP-Umgebung zu definieren. Siehe auch Externe Security-Konfiguration .
IMPERSONATE_REMOTE	Innerhalb der Natural Web I/O Interface Server-Umgebung findet keine Impersonierung statt. Wenn die Sitzung in einer Remote-TP-Umgebung verteilt wird, wird der Client impersoniert. Der Client muss im Security-System der Remote-TP-Umgebung definiert sein. Siehe auch Web I/O Interface Server Security Exit <code>NATUXRFE</code> und den Abschnitt Zusammenwirken der Produkte im Kapitel Vorstellung des Natural Web I/O Interface Server CICS Adapter . Anmerkung: Überprüfen Sie die korrekte Installation von <code>NATUXRFE</code> . Ein Map Environment-Versuch mit einer gültigen Benutzerkennung und einem ungültigen Kennwort sollte mit dem Fehler NAT0873 fehlschlagen.
IMPERSONATE	Die Impersonierung erfolgt innerhalb der Natural Web I/O Interface Server-Umgebung und in einer Remote-TP-Umgebung. Der Client muss im Security-System des Natural Web I/O Interface Server und in der Remote-TP-Umgebung definiert sein.



Anmerkung: Für einen Batch-Server sind `SECURITY_MODE=IMPERSONATE` und `SECURITY_MODE=IMPERSONATE_LOCAL` identisch. `SECURITY_MODE` erfordert Natural Security (NSC).

Es gibt keinen Standardwert.

Beispiel:

```
SECURITY_MODE=IMPERSONATE
```

SECURITY_SYSLOG

Schreiben einer Konsolenmeldung bei Security-Problemen mit der externen Security. Es wird eine Zeile in die Konsole geschrieben, zum Beispiel 'Aug-23 16:22:46.77 S0305217 BPXM023I (INITUSER) SERVERID - Ungültige Benutzer-ID angegeben.' Weitere Informationen zum Grund finden Sie im entsprechenden Trace- oder Fehlerprotokoll des Servers.

Wert	Erklärung
YES	Security-Meldung an die Konsole schreiben.
NO	Keine Security-Meldung an die Konsole schreiben. Dies ist der Standardwert.

Beispiel:

```
SECURITY_SYSLOG=YES
```

SESSION_PARAMETER

Dieser optionale Konfigurationsparameter definiert Sitzungsparameter, die der im Verbindungsdialog des Natural Web I/O Interface Client angegebenen Parameterzeichenkette vorangestellt werden.

Wert	Erklärung
<i>parameter-string</i>	Diese Zeichenkette kann sich über mehrere Zeilen erstrecken. Ein +-Zeichen am Ende einer Zeichenkettenzeile bedeutet, dass eine weitere Zeile folgt.

Es gibt keinen Standardwert.

Beispiel 1:

```
SESSION_PARAMETER='NUCNAME=NATNUCvr' +
'PROFILE=(NWOPARM,18006,48),ADAMODE=0,' +
'BPI=(TYPE=NAT,SIZE=6044),BPI=(TYPE=EDIT,SIZE=2048)', +
'BPI=(TYPE=SORT,SIZE=1024)'
```

- dabei steht *vr* für die Versions- und die Release-Nummer.

Beispiel 2:

```
SESSION_PARAMETER=FNAT=(10,930)
```

Die Einstellung im zweiten Beispiel legt fest, dass jede Sitzung auf diesem Natural Web I/O Interface Server mit dem Session-Parameter `FNAT=(10,930)` gestartet wird, der an die vom Benutzer angegebenen Parameter oder die Definitionen im Konfigurationsparameter `DEFAULT_PROFILE` angehängt wird.

SESSION_TIMEOUT

Beenden inaktiver Sitzungen, wenn der Wert des Parameters `SESSION_TIMEOUT` erreicht ist. Überprüfung einmal täglich um `HH:MM` (24 Stunden) oder alle *n* Minuten, ob Sitzungen länger als *n* Minuten inaktiv sind.

Der Server startet nicht, wenn ein ungültiger Parameter `SESSION_TIMEOUT` angegeben wird.

Wert	Erklärung
hh:mm,n <numeric value greater than 0> oder m <numeric value greater than 0>,n <numeric value>0>	Wenn das Format hh:mm lautet, wird einmal täglich um hh:mm auf Sitzungen geprüft, die länger als <i>n</i> Minuten inaktiv sind. oder Wenn das Format ein numerischer Wert ist, wird alle <i>m</i> Minuten auf Sitzungen geprüft, die länger als <i>n</i> Minuten inaktiv sind.

Beispiele:

```
SESSION_TIMEOUT=19:30,480
```

Jeden Tag um 19:30 Uhr werden Sitzungen gelöscht, die länger als 480 Minuten inaktiv sind.

```
SESSION_TIMEOUT=360,480
```

Alle 360 Minuten werden Sitzungen gelöscht, die länger als 480 Minuten inaktiv sind.

THREAD_NUMBER

Dieser Konfigurationsparameter gibt die Anzahl der physischen Speicher-Threads an, die vom Natural-Frontend zugeordnet werden sollen, d. h. die Anzahl der Sitzungen, die parallel ausgeführt werden können.



Anmerkung: Dieser Parameter ist obsolet, wenn der Natural Web I/O Interface Server CICS Adapter oder der Natural Web I/O Interface Server IMS Adapter verwendet wird.

Wert	Erklärung
<i>thread-number</i>	Anzahl der zuzuordnenden physischen Speicher-Threads. Anmerkung: Diese Zahl begrenzt nicht die Anzahl der Sitzungen innerhalb des Servers, sondern die Anzahl der Sitzungen, die gleichzeitig im Ausführungsstatus sein können. Die Anzahl der Sitzungen wird durch die Größe des Natural-Swap-Mediums begrenzt.
3	Dies ist der Standardwert.

Beispiel:

```
THREAD_NUMBER=5
```

THREAD_SIZE

Dieser Konfigurationsparameter gibt die Größe jedes physischen Speicher-Threads an, der die Natural-Sitzungsdaten zum Zeitpunkt der Ausführung enthält.



Anmerkung: Dieser Parameter ist obsolet, wenn der Natural Web I/O Interface Server CICS Adapter oder der Natural Web I/O Interface Server IMS Adapter verwendet wird.

Wert	Erklärung
<i>thread-size</i>	Größe (in KB) jedes physischen Speicher-Threads.
500	Dies ist der Standardwert.

Beispiel:

```
THREAD_SIZE=800
```

TRACE_FILTER

Mit diesem optionalen Konfigurationsparameter können Sie das Tracing durch einen logischen Filter einschränken, um das Volumen der Server-Trace-Ausgabe zu reduzieren, zum Beispiel:

```
TRACE_FILTER="Client=(KSP P*)"
```

Jede Anfrage der Benutzerkennung KSP und jede Anfrage der Benutzerkennungen, die mit einem P beginnen, werden protokolliert.

Siehe [Trace-Filter](#) im Kapitel *Betrieb des Natural Web I/O Interface Server*.

TRACE_LEVEL

Wert	Erklärung
<i>trace-level</i>	Siehe <i>Trace Level</i> im Kapitel <i>Betrieb des Natural Web I/O Interface Server</i> .
0	Dies ist der Standardwert.

Beispiel:

```
TRACE_LEVEL=0x00000011
```

or alternativ

```
TRACE_LEVEL=31+27
```

Die Einstellung im Beispiel schaltet die **Bits 31 und 27** ein.

UPPERCASE_SYSTEMMESSAGES

Dieser Konfigurationsparameter dient zum Aktivieren oder Deaktivieren der Umsetzung aller NWO-Fehlermeldungen und Trace-Ausgaben in Großbuchstaben. Diese Funktion ist für Kunden gedacht, die Zeichensätze ohne definierte Kleinbuchstaben verwenden.

Wert	Erklärung
YES	Umsetzung in Großbuchstaben aktivieren.
NO	Umsetzung in Großbuchstaben deaktivieren. Dies ist der Standardwert.

Beispiel für eine Web I/O Interface Server-Konfigurationsdatei

```
# This is a comment
SESSION_PARAMETER=profile=(stgqa,10,930) fuser=(10,32) CFICU=ON THREAD_NUMBER=2
THREAD_SIZE=700 FRONTEND_NAME=NATOS42L # and another comment PORT_NUMBER=4811
```

Web I/O Interface Server Datasets für z/OS

Der Natural Web I/O Interface Server benötigt die folgenden Datasets:

STGCONFIG	Definiert die Server-Konfigurationsdatei.
STGTRACE	Die Server-Trace-Ausgabe.
STGSTDO	Der stdo-Dataset.
STGSTDE	Die stde-Fehlerausgabe.

Alternativ können Sie jeden Dataset-Namen durch die Server-ID qualifizieren.

NWOS1C	Definiert die Server-Konfigurationsdatei für den Server NWOS1.
NWOS1T	Die Server-Trace-Ausgabe für den Server NWOS1.
NWOS1O	Der stdo-Datensatz für den Server NWOS1.
NWOS1E	Die stde-Fehlerausgabe für den Server NWOS1.

Web I/O Interface Server User Exits

User Exit NSECUX01

Dieser User Exit ist nur anwendbar, wenn der Parameter `SECURITY_MODE` auf `IMPERSONATE` oder `IMPERSONATE` gesetzt ist.

Mit diesem User Exit können Sie die für die RACF-Anmeldung verwendete Benutzerkennung anpassen. Dies ist nützlich, wenn sich die RACF-Benutzerkennungen und die in Natural verwendeten Benutzerkennungen gemäß einer standardisierten Regel unterscheiden. Beispielsweise entspricht jede RACF-Benutzerkennung der entsprechenden Natural-Benutzerkennung, der zwei Dollarzeichen (\$\$) vorangestellt sind.

Wenn der Exit (das Lademodul `NSECUX01`) in der Lademodul-Verkettung des Web I/O Interface gefunden wird, wird er unter Verwendung der Standard-Linkage-Konventionen (direkter Zweig unter Verwendung einer BASR-Anweisung) aufgerufen, bevor der Benutzer gegenüber RACF validiert wird.

Die folgenden Parameter werden an den Exit übergeben:

Name	Format	Ein/Aus	Beschreibung
sUId	CL64	I/O	Für die RACF-Anmeldung zu ändernde Benutzerkennung.

Der Exit wird unter Verwendung von Standard-Linkage-Konventionen aufgerufen.

Beispiel für einen in C implementierten User Exit:

```
#include  
<string.h> #include <stdio.h> # pragma linkage (NSECUX01,  
FETCHABLE) void NSECUX01(char sUId[64]) { char sUIdTemp[64]; printf("Uex got  
usid:%s\n", sUId); strcpy(sUIdTemp, sUId); sprintf(sUId, "$$%s", sUIdTemp);  
printf("Uex ret usid:%s\n", sUId); return; }
```

Der oben genannte Exit erweitert jede Benutzerkennung um zwei vorangestellte Dollarzeichen (\$\$), wenn sie für die RACF-Anmeldung verwendet wird.

9 Installation des Natural Web I/O Interface Server CICS

Adapter unter z/OS

- Voraussetzungen für den CICS Adapter 66
- Installationsvorgang 66

Dieses Kapitel beschreibt, wie Sie die CICS-Verbindung für einen Natural Web I/O Interface Server installieren, der unter z/OS im Batch-Modus ausgeführt wird.

Voraussetzungen für den CICS Adapter

Weitere Informationen finden Sie im Abschnitt [Voraussetzungen](#).

Installationsvorgang

Um den Natural Web I/O Interface Server CICS Adapter zu installieren, müssen Sie die folgenden Schritte ausführen:

Schritt 1: CICS anpassen

(Job I005, Steps 9405, 9406, 9410, 9411)

Die Natural Web I/O Interface Server-Ladebibliothek muss in der CICS DFHRPL-Verkettung definiert werden.

Die CICS TCP/IP-Umgebung kann mit dem Konfigurationsmakro EZACICD im Konfigurations-Dataset EZACONFG oder der Konfigurationstransaktion EZAC angepasst werden. In dieser Anleitung wird die Verwendung der Transaktion EZAC beschrieben.

Passen Sie den Standard-Listener CSKL der CICS-Socket-Schnittstelle mit der CICS-Transaktion EZAC, ALTER, LISTENER an und definieren Sie auf dem zweiten Bildschirm NATUXRFE im SECEXIT.

Für die Impersonierung mit Passphrasen ist ein erweiterter Listener erforderlich. Passen Sie einen vorhandenen erweiterten Listener an oder definieren Sie einen neuen mit der CICS-Transaktion EZAC, DEFINE, LISTENER mit TRANID CSEL (oder einem anderen Namen Ihrer Wahl) und FORMAT ENHANCED. Passen Sie diesen erweiterten Listener an, indem Sie die Einstellungen auf dem zweiten Bildschirm wie folgt ändern:

```
CSTRANid ===> NRFE CSSTYPe ===> KC CSDELAY  
===> 000000 MSGLENgth ===> 000 PEEKDATA ===> NO MSGFORMat ===>  
EBCDIC USEREXIT ===> NATUXRFE GETTID ===> NO USERID ===>
```

Die Definition von SECEXIT=NATUXRFE ist obligatorisch, wenn ein erweiterter Listener verwendet wird oder wenn der Natural Web I/O Interface Server mit Impersonierung (Parameter [SECURITY_MODE](#)) gestartet wird.

Starten Sie den Listener mit der CICS-Transaktion EZA0.

Geben Sie AF=INET6 an, wenn IPv6 (Internet Protocol Version 6) verwendet werden soll.



Anmerkung: Die Verwendung von IPv6 ist nur bei Natural Web I/O Interface Server-Version 8.3.2 oder höher möglich.

Die folgenden CICS-Ressourcendefinitionen sind erforderlich:

1. Definieren Sie die CICS-Transaktion für das Remote-Frontend. Dieser Transaktionsname ist ein beliebiger Name, der im Natural Web I/O Interface-Konfigurationsparameter `RFE_CICS_TA_NAME` definiert werden muss. In diesem Dokument wird der Transaktionsname `NRFE` verwendet.

```
DEFINE TRANSACTION(NRFE)
GROUP(NW0group) PROGRAM(NATCNRFE) TWASIZE(128)
RESART(NO) TASKDATAKEY(USER) TASKDATALOC(ANY)
```

2. Definieren Sie die Programme `NATCNRFE`, `NATLRGW0` und `NATUXRFE`.

Define the program `NATUXRFE`.

```
DEFINE PROGRAM(NATCNRFE)
GROUP(NW0group) LANGUAGE(C) DATALOCATION(ANY)
EXECKEY(USER)
```

```
DEFINE PROGRAM(NATLRGW0)
GROUP(NW0group) LANGUAGE(C) DATALOCATION(ANY)
EXECKEY(USER)
```

```
DEFINE PROGRAM(NATUXRFE)
GROUP(NW0group) LANGUAGE(LE370) DATALOCATION(ANY)
EXECKEY(CICS)
```

`NATUXRFE` muss mit `EXECKEY(CICS)` definiert werden, da die Transaktion `CSKL` mit `TASKDATAKEY(CICS)` definiert ist und das Programm `NATUXRFE` Teil der von `CSKL` initiierten Aufrufkette ist. Dies gilt auch, wenn eine andere mit `TASKDATAKEY(CICS)` definierte Transaktion zum Aufruf von `NATUXRFE` verwendet wird.

Wenn Sie die CICS Open Transaction Environment (OTE) verwenden, müssen Sie außerdem die folgenden Parameter für `NATCNRFE` und `NATUXRFE` setzen:

```
API(OPENAPI) CONCURRENCY(THREADSAFE)
```

Dies ist beispielsweise erforderlich, wenn der Parameter `OTE=YES` für das Konfigurationsmakro `EZACICD` mit `TYPE=CICS` für die CICS-Region gesetzt ist oder wenn der Parameter mit der Transaktion `EZAC,ALTER,CICS` gesetzt ist.

Die Werte von `API`, `CONCURRENCY` und `EXECKEY` für `NATCNRFE` müssen mit denen für den umgebungsabhängigen Nukleus von Natural übereinstimmen, da Natural von `NATCNRFE` unter Ver-

wendung von Standard-Linkage-Konventionen (direkte Verzweigung mit einer BASR-Anweisung) anstelle eines EXEC CICS LINK-Kommandos aufgerufen wird.

Die Programmdefinition von NATLRGW0 muss für CICS OTE nicht angepasst werden.

3. Für den Db2-Zugriff muss ein Db2-Plan-Name definiert werden. Wenn Sie in der Ressourcendefinition DB2CONN keinen Db2-Plan-Namen für Pool-Threads angegeben haben, müssen die in RFE_CICS_TA_NAME angegebene Transaktion und der zugehörige Db2-Plan-Name mit einer DB2TRAN- und/oder DB2ENTRY-Ressourcendefinition für CICS definiert werden.



Anmerkung: Die dynamische Planauswahl, die von der Natural for Db2-Schnittstelle bereitgestellt wird, darf nicht verwendet werden.

4. Für den Db2-Zugriff muss die Autorisierungskennung (IDD), unter der die NWO-CICS-Transaktion auf Db2 zugreift, über die erforderlichen Berechtigungen für den Db2-Zugriff verfügen. Die zu verwendende Autorisierungskennung wird in der Ressourcendefinition DB2ENTRY angegeben. Wenn Sie die Option USERID wählen, wird die Benutzerkennung des CICS-Systems verwendet, da die NWO-CICS-Transaktionen unter der Benutzerkennung des CICS-Systems ausgeführt werden.

Die Beispiel-JCL mit den folgenden Members definiert alle erforderlichen CICS-Einträge:

NWOCONF

Schritt 2: Member für CICS-Server erstellen

Job I009, Step 9411, Member NWOCONF1 für CICS-Server erstellen

Schritt 3: Objektmodule in die NWO-Ladebibliothek einbinden

(Job I054, Step 8420)

Die NWO-Objektmodule müssen mit den erforderlichen Laufzeiterweiterungen Ihrer CICS-Installationen zu ausführbaren Lademodulen verlinkt werden.

Siehe Beispieljob NW0I054 im Dataset NW0vrs.JOBS.

Schritt 4: Startprozedur erstellen

Job I200, Step 9416, Startprozedur für CICS-Server erstellen

Schritt 5: Natural Web I/O Interface Server anpassen

Um die Natural-Sitzungen des Natural Web I/O Interface in CICS zu disponieren, müssen Sie die Konfigurationsdatei Ihres Natural Web I/O Interface Server anpassen, der unter z/OS im Batch-Modus läuft. Zu diesem Zweck stehen zwei Beispiel-JCL-Members (NW0I009C und NW0CONF0) zur Verfügung.

Siehe *Konfigurieren des Natural Web I/O Interface Server CICS Adapter* und *Konfigurieren des Natural Web I/O Interface Server*.

10 Konfiguration des Natural Web I/O Interface Server CICS

Adapter

- Konfigurationsdatei für den CICS Adapter 72
- Konfigurationsparameter für den CICS Adapter 72

Dieses Dokument beschreibt, wie Sie die CICS-Verbindung für einen Natural Web I/O Interface Server (Produktcode NWO) unter z/OS konfigurieren.

Konfigurationsdatei für den CICS Adapter

Nach Abschluss der Installation des Natural Web I/O Interface Server CICS Adapter muss die Konfiguration des Natural Web I/O Interface Server CICS Adapter in der Konfigurationsdatei des entsprechenden Natural Web I/O Interface Server vorgenommen werden.

Um den CICS Adapter zu aktivieren, müssen Sie das Remote-Frontend-Modul im NWO-Konfigurationsparameter `FRONTEND_NAME` (`FRONTEND_NAME=NATCSRFE`) angeben.

Konfigurationsparameter für den CICS Adapter

Es gibt die folgenden CICS-relevanten Konfigurationsparameter:

- `RFE_CICS_TA_NAME`
- `RFE_CICS_FE_NAME`
- `RFE_CICS_TA_HOST`
- `RFE_CICS_TA_PORT`
- `RFE_CICS_TA_INIT_TOUT`
- `RFE_CICS_KEEP_TA`
- `RFE_CICS_TRACE`

RFE_CICS_TA_NAME

Dieser Konfigurationsparameter gibt die CICS-Transaktion an, die zum Starten des Remote-Frontend in CICS verwendet werden soll. Diese Transaktion muss in CICS definiert sein und auf das Programm `NATCNRFE` verweisen. Siehe auch [Installation des Natural Web I/O Interface Server CICS-Adapters unter z/OS](#).

Standardwert	Erklärung
Beispiel	<code>RFE_CICS_TA_NAME=NRFE</code>

RFE_CICS_FE_NAME

Dieser Konfigurationsparameter gibt den Natural CICS-Nukleus an, den Sie mit der entsprechenden Natural for z/OS-Installation unter CICS installiert haben. Dieses Programm muss in CICS definiert sein.

Standardwert	keiner
Beispiel	RFE_CICS_FE_NAME=NCI <i>vr</i> NUC

wobei *vr* für die relevante Versions- und Release-Nummer steht.

Siehe auch *Natural CICS-Schnittstelle auf z/OS installieren, CICS anpassen* in der *Installation für z/OS-Dokumentation*.

RFE_CICS_TA_HOST

Dieser Konfigurationsparameter gibt die TCP/IP-Adresse des Hosts an, auf dem das gewünschte CICS ausgeführt wird. Dieser Parameter kann weggelassen werden, wenn der Web I/O Interface Server und CICS auf demselben TCP/IP-Knoten ausgeführt werden.

Standardwert	Die Hostadresse des Servers.
Beispiel	RFE_CICS_TA_HOST=node1 oder RFE_CICS_TA_HOST=157.189.160.55

RFE_CICS_TA_PORT

Dieser Konfigurationsparameter gibt den TCP/IP-Port des von CICS bereitgestellten Listener an.

Sie können diese Portnummer mit der von CICS bereitgestellten Transaktion EZAC abrufen. Das CICS-Kommando `EZAC DISPLAY LISTENER` zeigt die Definitionen des CICS-Standard-Listener an.



Anmerkung: Diese Portnummer wird im Web I/O Interface Client nicht für die Verbindung mit einem Web I/O Interface Server verwendet. Diese Portnummer (und die Definition `RFE_CICS_TA_HOST`) wird intern vom Web I/O Interface Server für die Kommunikation mit der CICS-Region verwendet.

Mögliche Werte	1 - 65535
Standardwert	keiner
Beispiel	RFE_CICS_TA_PORT=3010

RFE_CICS_TA_INIT_TOUT

Wenn der Web I/O Interface Client eine Anfrage an einen Natural Web I/O Interface Server sendet, der für die Verwendung des CICS-Remote-Frontend konfiguriert ist, startet das Remote-Frontend eine CICS-Transaktion (NRFE) zur Verarbeitung der Anfrage. Die CICS-Transaktion überwacht wiederum das TCP/IP, um die für die Verarbeitung der Anfrage erforderlichen Daten vom Web I/O Interface Server zu empfangen.

Dieser Konfigurationsparameter gibt den Timeout-Wert (in Sekunden) an, den eine gestartete Transaktion wartet, bis die erwarteten Anfragedaten vom Server eintreffen. Wenn dieser Timeout abläuft, wird die Anfrage mit einem NAT9940-Fehler abgebrochen.

Standardwert	5
Beispiel	RFE_CICS_TA_INIT_TOUT=20



Anmerkung: Definieren Sie keinen Wert unter 5.

RFE_CICS_KEEP_TA

Für jede von Natural gesendete Anfrage öffnet der Natural Web I/O Interface Server eine TCP/IP-Verbindung zur CICS-Region und startet eine CICS-Transaktion (NRFE) zur Verarbeitung der Anfrage. Mit `RFE_CICS_KEEP_TA=YES` bleibt die CICS-Transaktion für die Verarbeitung weiterer Anfragen desselben Client aktiv. Dies spart den Aufwand für die Herstellung der TCP/IP-Verbindung und die Transaktionsinitialisierung für aufeinanderfolgende Anfragen, verbraucht jedoch aufgrund der wartenden Transaktionen innerhalb der CICS-Region mehr Ressourcen.

Die Transaktionswartezeit (für aufeinanderfolgende Anfragen) wird durch [RFE_CICS_TA_INIT_TOUT](#) begrenzt. Das heißt, wenn der Zeitabschnitt zwischen zwei aufeinanderfolgenden Anfragen die durch `RFE_CICS_TA_INIT_TOUT` festgelegte Zeit überschreitet, werden die CICS-Transaktion und die TCP/IP-Verbindung unabhängig von der Definition von `RFE_CICS_KEEP_TA` beendet.

`RFE_CICS_TA_INIT_TOUT=5` ist ein angemessener Wert, um Transaktionen für mehrere Anfragen wiederzuverwenden, die durch eine einzelne Aktion in Natural Studio initiiert wurden, und um CICS-Ressourcen zu sparen, wenn Natural Studio auf die nächste Aktion des Benutzers wartet.

Standardwert	NO
Beispiel	RFE_CICS_KEEP_TA=YES

RFE_CICS_TRACE

Dieser Konfigurationsparameter legt die Trace-Stufe für das Remote-Frontend fest.

Die Trace-Stufe ähnelt der für den Web I/O Interface Server implementierten Trace-Stufe. Es handelt sich um eine Bitfolge, wobei jedes Bit für eine bestimmte Trace-Information zuständig ist:

Bit 31	Trace-Hauptereignisse (Transaktionsinitialisierung/-beendigung, Anfrageverarbeitung).
Bit 30	Detaillierte Funktionen.
Bit 29	Interne Speicherbereiche ausgeben (Dump).
Bit 27	Zwischen dem Web I/O Interface Server und CICS ausgetauschten Puffer-Header ausgeben (Dump).
Bit 26	Gesamten zwischen dem Web I/O Interface Server und CICS ausgetauschten Puffer ausgeben (Dump).
Bit 25	Nur den für den Natural Web I/O Interface Server relevanten Puffer (Remote-Gateway-Puffer) ausgeben (Dump).
Bit 23	Trace nur für Fehlersituationen erstellen.
Bit 07	Trace in der Web I/O Interface Server-Region aktivieren.
Bit 06	Trace in der CICS-Region aktivieren.
Bit 00	Für Trace-Stufenerweiterung reserviert.

Die Trace-Destination ist der für STDOUT definierte Dataset.

Standardwert	0
Beispiel	RFE_CICS_TRACE=0x02000011 Dump der Hauptereignisse und des Pufferheaders in der CICS-Region (Bits 31 + 27 + 07).

Das folgende Beispiel zeigt eine Serverkonfigurationsdatei, die den Natural Web I/O Interface Server CICS Adapter verwendet:

```
# the Web I/O Interface Server parameter
SESSION_PARAMETER=PROFILE=(NWO,10,930) FRONTEND_NAME=NATCSRFE # use the CICS
Adapter front-end PORT_NUMBER=4711 # the port number used by the Web I/O
Interface Client # the CICS Adapter parameter RFE_CICS_TA_NAME=NRFE # the CICS
transaction for remote front-end RFE_CICS_TA_PORT=3010 # the port of the CICS
listener # no RFE_CICS_TA_HOST is defined. This requires # that CICS runs on
the same node as the # server.
RFE_CICS_FE_NAME=NCIvrNUC # the name of the
installed Natural CICS nucleus RFE_CICS_TA_INIT_TOUT=20 # transaction timeout
is 20 seconds ↵
```

wobei *vr* für die relevante Versions- und Release-Nummer steht.



Anmerkung: Die Serverparameter `THREAD_NUMBER` und `THREAD_SIZE` sind bei Verwendung des Natural Web I/O Interface Server CICS Adapter nicht mehr erforderlich.

11 Installation des Natural Web I/O Interface Server IMS

Adapter

■ Voraussetzungen für den IMS Adapter	78
■ Beispieljobs	78
■ Installationsverfahren	78

Dieses Kapitel beschreibt, wie Sie die IMS TM-Verbindung für einen Natural Web I/O Interface Server (abgekürzt: „NWO-Server“) installieren, der unter z/OS im Batch-Modus ausgeführt wird.

In diesem Dokument steht die Notation *vrs* oder *vr* für die jeweilige Produktversion (siehe auch Version im *Glossar*).

Voraussetzungen für den IMS Adapter

Weitere Informationen finden Sie im Abschnitt *Voraussetzungen*.

Beispieljobs

Die Installation erfolgt über Installationsjobs. Diese sind im Dataset `NWOvrs.JOBS` auf dem Installationsdatenträger für das Natural Web I/O Interface enthalten. Dabei ist der Produktcode `NWO` das Präfix und *vrs* der Platzhalter für die jeweilige Produktversion.

Installationsverfahren

Führen Sie die folgenden Schritte aus:

Schritt 1: Passen Sie den IMS-Listener an

Eine Beschreibung der folgenden Punkte finden Sie im *z/OS Communications Server IP IMS Socket Guide*:

- Einrichten und Betreiben des IMS-Listeners und
- Verwenden der IMS-TCP/IP-Steueranweisungen.



Anmerkung: Die Portnummer des IMS-Listener wird später im NWO-Konfigurationsparameter `RFE_IMS_TA_PORT` verwendet.

Schritt 2: Installieren Sie die Server-Transaktion

Es wird davon ausgegangen, dass die Natural IMS TM-Schnittstelle bereits installiert ist. Daher werden in diesem Schritt nur die zusätzlichen Schritte beschrieben, die zum Erstellen der Natural Web I/O Interface Serverumgebung erforderlich sind.

Schritt 2.1: Passen Sie die Transaktionscodetabelle an

(Job I055, Steps 2555, 2556)

- Fügen Sie im Makro NTIMSPT des Natural-Parametermoduls folgenden Eintrag hinzu:

NTIMSPT TRAN=NAT <i>vrs</i> AD,TYPE=SFE,	X
ENVPID=ENVNW000,	X
PSB=NII <i>vrs</i> AD	

- Assemblieren und verlinken Sie das Parametermodul.

Step 2.2: Passen Sie die Umgebungsparameter an

(Job I055, Steps 2580, 2581)

- Fügen Sie im Makro NTIMSPE des Natural-Parametermoduls folgenden Eintrag hinzu:

NTIMSPE ENTRYNM=ENVNW000,	X
THBELOW=OFF,	X
THSIZE=1536000 (1500K)	

- Assemblieren und verlinken Sie das Parametermodul.

Schritt 2.3: Verlinken Sie das Natural IMS TM-Schnittstellenmodul neu

(Job I055, Step 2582)

Verlinken Sie das Natural IMS TM-Schnittstellenmodul NII*vrs*IF neu.

Schritt 2.4: Natural IMS TM-Parametermodul

(Job I080, Steps 2500, 2510)

- Erstellen Sie das Natural IMS TM-Parametermodul.
- Assemblieren und verlinken Sie das Natural IMS TM-Parametermodul.

Schritt 2.5: Verlinken Sie das Server-Frontend

(Job I080, Step 2586)

Verlinken Sie das Server-Frontend.

Stellen Sie sicher, dass der Name des Server-Frontends mit dem Wert übereinstimmt, der mit dem Schlüsselwort-Subparameter PSB des Makros NTIMSPT in Installationsschritt 2.1 oben angegeben wurde.

Schritt 3: Passen Sie IMS TM an

- Definieren Sie den PSB für die Server-Transaktion.

Als Mindestanforderung benötigen Sie einen TP PCB.

Definieren Sie die Anwendung der Server-Transaktion in der Stage1-Eingabe. Die Transaktion ist eine nichtkonversationale Single-Mode-, Single-Segment-Transaktion.

Beispiel:

```
APPLCTN PSB=NIIvrsAD,PGMTYPE=TP,SCHDTYP=PARALLEL
TRANSACT CODE=NATvrsAD,MODE=SNGL,X
MSGTYPE=(SNGLSEG,NONRESPONSE,4)
```

- Passen Sie die MPP-Region an.

Fügen Sie in der JCL der MPP-Region die SYSTPCD DD-Anweisung hinzu.

Eine Beschreibung zur Konfiguration von TCPIP.DATA finden Sie im *z/OS Communications Server IP IMS Socket Guide*.

- Verketteten Sie den Dataset, der die NWO-Ladebibliothek enthält, mit der STEPLIB DD-Anweisung.

Bevor Sie mit der Installation des Natural Web I/O Interface Server IMS Adapter beginnen, müssen Sie den SMA-Parameter NWO-SRV-IMS auf Y (Ja) setzen.

Um den Natural Web I/O Interface Server IMS Adapter zu installieren, müssen Sie folgende Schritte ausführen:

Schritt 4: Member für NWO-Server erstellen

(Job I009, Step 9412)

Erstellen Sie das Member `NWOCONF1` für den NWO-Server.

Schritt 5: Objektmodule in die NWO-Ladebibliothek verlinken

(Job I054, Step 9440)



Anmerkungen:

1. Das Modul `NATINRFE` findet bei zwei verschiedenen Produkten Anwendung: dem Natural Development Server (NDV) und der Natural Web I/O Interface (NWO-Server). Wenn Sie NDV bereits installiert haben, ist das Modul `NATINRFE` möglicherweise schon vorhanden. Es ist jedoch unerheblich, wenn Sie `NATINRFE` mit NWO neu installieren, da das resultierende Modul bei beiden Installationen identisch ist.
2. Das Modul `NATISRFE` (in der Konfiguration über `FRONTEND_NAME=NATISRFE` referenziert) wurde bereits in den vorherigen Schritten der Batch-Installation verlinkt.

Schritt 6: Startprozedur erstellen

(Job I200, Step 9417)

Erstellen Sie die Startprozedur für den NWO-Server.

Schritt 7: Natural Web I/O Interface Server anpassen

Um die NWO Natural-Sitzungen in IMS zu dispatchen, müssen Sie die Konfigurationsdatei Ihres Development Server, der unter z/OS im Batch-Modus läuft, anpassen. Zu diesem Zweck stehen zwei Beispiel-JCL-Members (`NWOI009I` und `NWOCONF1`) zur Verfügung.

Weitere Informationen siehe [Konfiguration des Natural Development Server IMS Adapter](#) und [Konfiguration des Natural Web I/O Interface Server](#).

12 Konfiguration des Natural Web I/O Interface Server IMS

Adapter

- Konfigurationsdatei für den IMS Adapter 84
- Konfigurationsparameter für den IMS Adapter 84

Dieses Dokument beschreibt, wie Sie die IMS TM-Verbindung für einen Natural Web I/O Interface Server (Produktcode NWO) unter z/OS konfigurieren.

Konfigurationsdatei für den IMS Adapter

Nach Abschluss der Installation des Natural Web I/O Interface Server IMS Adapter muss die Konfiguration des Natural Web I/O Interface Server IMS Adapter in der Konfigurationsdatei des entsprechenden Natural Web I/O Interface Server vorgenommen werden.

Um den Natural Web I/O Interface Server IMS Adapter zu aktivieren, müssen Sie das Remote-Frontend-Modul im NWO-Konfigurationsparameter `FRONTEND_NAME` angeben:

```
FRONTEND_NAME=NATISRFE.
```

Konfigurationsparameter für den IMS Adapter

Es gibt die folgenden IMS-relevanten Konfigurationsparameter:

- `RFE_IMS_TA_NAME`
- `RFE_IMS_TA_HOST`
- `RFE_IMS_TA_PORT`
- `RFE_IMS_TRACE`

RFE_IMS_TA_NAME

Dieser Konfigurationsparameter gibt die IMS-Transaktion an, die zum Starten des Remote-Frontend in IMS TM verwendet werden soll. Diese Transaktion muss in IMS TM definiert sein und sich auf das Programm `NATINRFE` beziehen. Siehe auch [Installation des Natural Web I/O Interface Server IMS Adapter](#).

Standardwert	keiner
Beispiel	<code>RFE_IMS_TA_NAME=NATvr\$AD</code>

RFE_IMS_TA_HOST

Dieser Konfigurationsparameter gibt die TCP/IP-Adresse des Hosts an, auf dem das gewünschte IMS TM ausgeführt wird. Dieser Parameter kann weggelassen werden, wenn der Web I/O Interface Server und IMS TM auf demselben TCP/IP-Knoten ausgeführt werden.

Standardwert	Die Hostadresse des Servers.
Beispiel	RFE_IMS_TA_HOST=node1 or RFE_IMS_TA_HOST=157.189.160.55

RFE_IMS_TA_PORT

Dieser Konfigurationsparameter gibt den TCP/IP-Port des von IMS bereitgestellten Listener an.



Anmerkung: Diese Portnummer wird im Web I/O Interface Client nicht für die Verbindung mit einem Web I/O Interface Server verwendet. Diese Portnummer (und die RFE_IMS_TA_HOST-Definition) wird intern vom Web I/O Interface Server für die Kommunikation mit der IMS-Region verwendet.

Possible Values	1 - 65535
Standardwert	keiner
Beispiel	RFE_IMS_TA_PORT=3010

RFE_IMS_TRACE

Dieser Konfigurationsparameter legt die Trace-Stufe für das Remote-Frontend fest.

Die Trace-Stufe ähnelt der für den Web I/O Interface Server implementierten Trace-Stufe. Es handelt sich um eine Bitfolge, wobei jedes Bit für bestimmte Trace-Informationen zuständig ist:

Bit 31	Trace-Hauptereignisse (Transaktionsinitialisierung/-beendigung, Anfrageverarbeitung)
Bit 30	Detaillierte Funktionen.
Bit 29	Interne Speicherbereiche ausgeben (Dump).
Bit 27	Zwischen Web I/O Interface Server und IMS TM ausgetauschten Puffer-Header ausgeben (Dump).
Bit 26	Zwischen Web I/O Interface Server und IMS TM ausgetauschten gesamten Puffer ausgeben (Dump).
Bit 25	Nur den für den Natural Web I/O Interface Server relevanten Puffer (Remote-Gateway-Puffer) ausgeben (Dump).
Bit 23	Trace nur für Fehlersituationen erstellen.
Bit 07	Trace in der Web I/O Interface Server-Region aktivieren.
Bit 06	Trace in der IMS-MPP-Region aktivieren.

Bit 00	Für Trace-Stufenerweiterung reserviert.
--------	---

Die Trace-Destination ist der für STDOUT definierte Dataset.

Standardwert	0
Beispiel	RFE_IMS_TRACE=31+27+7 Dump der Hauptereignisse und des Puffer-Header in der IMS MPP-Region (Bits 31 + 27 + 7).

Das folgende Beispiel zeigt eine Serverkonfigurationsdatei, die den Natural Web I/O Interface Server IMS Adapter verwendet:

```
# The Web I/O Interface Server parameter.
SESSION_PARAMETER=PROFILE=(NWO,10,930)
FRONTEND_NAME=NATISRFE           # Use the IMS Adapter front-end.
PORT_NUMBER=4711                 # The port number used by the Web I/O Interface Client.

# The IMS Adapter parameter.
RFE_IMS_TA_NAME=NATvrsAD         # The IMS transaction for the remote front-end.
RFE_IMS_TA_PORT=3020             # The port of the IMS listener.
                                  # No RFE_IMS_TA_HOST is defined. This requires
                                  # that IMS runs on the same node as the server.
```

wobei *vr* für die relevante Versions- und Release-Nummer steht.



Anmerkung: Die Serverparameter THREAD_NUMBER und THREAD_SIZE sind bei Verwendung des Natural Web I/O Interface Server IMS Adapter nicht mehr erforderlich.

V

Installation des Natural Web I/O Interface Client

In diesem Kapitel wird erläutert, wie Sie den Natural Web I/O Interface Client auf Tomcat installieren, damit er mit dem Serverteil des Natural Web I/O Interface verwendet werden kann, der in einer Natural for z/OS-, Natural for Linux- oder Natural for Windows-Laufzeitumgebung ausgeführt wird.

Die folgenden Themen werden behandelt:

[Voraussetzungen für den Natural Web I/O Interface Client](#)

[Installation des Natural Web I/O Interface Client auf Apache Tomcat](#)

[Migration des Natural Web I/O Interface Client von IIS nach Apache Tomcat](#)

13 Voraussetzungen für den Natural Web I/O Interface Client

■ Servlet Container	90
■ Natural for z/OS	90
■ Natural for Linux	90
■ Natural for Windows	91
■ Browser-Voraussetzungen	91

Servlet Container

Der folgende Servlet-Container wird unterstützt. Der Servlet-Container wird nicht mit dem Natural Web I/O Interface ausgeliefert. Er kann gemäß den Lizenzbedingungen unter der unten angegebenen Adresse bezogen werden.

Apache Tomcat 9 und 10 (siehe <https://tomcat.apache.org/>).

Natural for z/OS

Wenn Sie den Natural Web I/O Interface Client mit Natural for z/OS verwenden möchten, muss Folgendes installiert sein:

- Natural for z/OS Version 8.2.5 oder höher und
- der Natural Web I/O Interface Server.

Ausführliche Informationen siehe:

- *Installation-Dokumentation*, die für Natural for z/OS bereitgestellt wird.
- Kapitel *Installation und Konfiguration des Natural Web I/O Interface Server* in der Version dieser *Natural Web I/O Interface-Dokumentation*, die für Natural for z/OS bereitgestellt wird.

Natural for Linux

Wenn Sie den Natural Web I/O Interface Client mit Natural for Linux and Cloud verwenden möchten, muss Folgendes installiert sein:

- Natural on Linux and Cloud Version 9.2.1 oder höher und
- Natural Web I/O Interface Server und Daemon.

Ausführliche Informationen siehe:

- *Installation-Dokumentation*, die für Natural for Linux bereitgestellt wird.
- Kapitel *Installation und Konfiguration des Natural Web I/O Interface Server* in der Version dieser *Natural Web I/O Interface-Dokumentation*, die für Natural for Linux bereitgestellt wird.

Natural for Windows

Wenn Sie den Natural Web I/O Interface Client mit Natural for Windows verwenden möchten, muss Folgendes installiert sein:

- Natural für Windows Version 9.2.1 oder höher und
- der Natural Web I/O Interface Server und Service.

Ausführliche Informationen siehe:

- *Installation-Dokumentation*, die für Natural for Windows bereitgestellt wird.
- Kapitel *Installation und Konfiguration des Natural Web I/O Interface Server* in der Version dieser *Natural Web I/O Interface-Dokumentation*, die für Natural for Windows bereitgestellt wird.

Browser-Voraussetzungen

In dieser Version werden folgende Browser unterstützt:

- Microsoft Edge (Chromium) – aktuelle Versionen
- Mozilla Firefox Extended Support Release – aktuelle Versionen, siehe Anmerkung (1)
- Google Chrome – aktuelle Versionen, siehe Anmerkung (2)

Anmerkungen:

(1) Es werden ausdrücklich nur die Extended Support Releases von Mozilla Firefox unterstützt. Aufgrund häufiger Upgrades der Mozilla Firefox-Consumer-Version kann die Kompatibilität des Natural Web I/O Interface Client mit zukünftigen Versionen von Mozilla Firefox nicht vollständig garantiert werden. Mögliche Inkompatibilitäten werden im Rahmen der regelmäßigen Wartungsarbeiten am Natural Web I/O Interface Client behoben.

(2) Aufgrund häufiger Versions-Upgrades von Google Chrome kann die Kompatibilität des Natural Web I/O Interface Client mit zukünftigen Versionen von Google Chrome nicht vollständig garantiert werden. Mögliche Inkompatibilitäten werden im Rahmen der regelmäßigen Wartungsarbeiten am Natural Web I/O Interface Client beseitigt.



Wichtig: Cookies und JavaScript müssen im Browser aktiviert sein.

14 Installation des Natural Web I/O Interface Client auf Apache

Tomcat

- Installationsschritte 94
- Überprüfung der Installation 95

Wenn Sie den Natural Web I/O Interface Client mit Apache Tomcat verwenden möchten, müssen Sie wie in diesem Kapitel beschrieben vorgehen.

Installationsschritte

Der Natural Web I/O Interface Client wird unter Verwendung des Tomcat Managers installiert.

Folgendes wird vorausgesetzt:

- `<install-dir>` ist der Pfad zum Hauptinstallationsverzeichnis. Standardmäßig ist dies `C:\SoftwareAG` unter Windows und `/opt/softwareag` unter Linux.
- `<host>` ist der Name des Rechners, auf dem Apache Tomcat installiert ist.
- `<port>` ist der Name des Ports, auf dem Apache Tomcat installiert ist. Bei einer Standardinstallation ist dies Port 8080.
- `<tomcat>` ist der Pfad zum Verzeichnis, in dem Apache Tomcat installiert ist.

Die folgenden Themen werden behandelt:

- [Erstinstallation](#)
- [Update-Installation](#)

Erstinstallation

➤ Um den Natural Web I/O Interface Client zu installieren:

- 1 Natural for Windows und Linux: Kopieren Sie den gesamten Inhalt des Verzeichnisses `<install-dir>/natural/INSTALL/nwoclient/tomcat` in ein Verzeichnis Ihrer Wahl auf Ihrer Festplatte.
- 2 Vergewissern Sie sich, dass Apache Tomcat läuft.
- 3 Öffnen Sie Ihren Webbrowser und geben Sie die folgende URL ein:

```
http://<host>:<port>/manager/html
```

Dadurch wird der Tomcat Manager geöffnet.

- 4 Machen Sie die Webanwendungsdatei `natuniweb.war` einsatzbereit:
 - Wählen Sie unter **Select WAR file to upload** den Pfad zur Datei `natuniweb.war` aus.
 - Wählen Sie **Deploy**.
- 5 Suchen Sie im Tomcat Manager nach der Anwendung **Natural Web I/O Interface Client** und wählen Sie **Reload**.

Update-Installation

➤ Um den Natural Web I/O Interface Client zu aktualisieren:

- 1 Natural for Windows und Linux: Kopieren Sie den gesamten Inhalt des Verzeichnisses `<install-dir>/natural/INSTALL/nwoclient/tomcat` in ein Verzeichnis Ihrer Wahl auf Ihrer Festplatte.

Oder:

Laden Sie den Natural Web I/O Interface Client für Apache Tomcat von <https://empower.softwareag.com/> herunter und entpacken Sie den Inhalt in ein Verzeichnis Ihrer Wahl auf Ihrer Festplatte.

- 2 Beenden Sie Apache Tomcat.
- 3 Erstellen Sie eine Sicherungskopie Ihrer Datei `sessions.xml`, die sich in `<tomcat>/webapps/natuniweb/WEB-INF` befindet.
- 4 Starten Sie Apache Tomcat.
- 5 Öffnen Sie Ihren Webbrowser und geben Sie die folgende URL ein:

```
http://<host>:<port>/manager/html
```

Dadurch wird der Tomcat Manager geöffnet.

- 6 Markieren Sie `natuniweb.war` in der Liste der installierten Anwendungen.
- 7 Wählen Sie **Undeploy**.
- 8 Stellen Sie die neue Version des Natural Web I/O Interface Clients wie bei einer Erstinstallation bereit.
- 9 Stellen Sie die zuvor gesicherte Datei `sessions.xml` wieder her.

Überprüfung der Installation

Es wird davon ausgegangen, dass `http://<host>:<port>` die URL Ihres Anwendungsservers ist.

Um die Installation zu überprüfen, müssen Sie die folgende URL in Ihren Webbrowser eingeben:

```
http://<host>:<port>/natuniweb/natural.jsp
```

Beispiel:

```
http://myhost:8080/natuniweb/natural.jsp
```

Der Natural Web I/O Interface Client wird nun in Ihrem Browser gestartet. Die Einträge, die auf der daraufhin angezeigten Anmeldeseite erscheinen, hängen von den Einstellungen in Ihrer Konfigurationsdatei ab. Weitere Informationen finden Sie unter [Konfiguration des Client](#).

15

Migration des Natural Web I/O Interface Client von IIS nach Apache Tomcat

- Bevor Sie den Natural Web I/O Interface Client installieren 98
- Installieren des Natural Web I/O Interface Client auf Apache Tomcat 99
- Konfigurieren des Natural Web I/O Interface Client auf Apache Tomcat 99

Microsoft Internet Information Services (IIS) wird nicht mehr unterstützt. Wenn Sie derzeit den Natural Web I/O Interface Client auf IIS verwenden, müssen Sie auf Apache Tomcat umsteigen.



Anmerkung: JBoss Application Server und Oracle GlassFish Server werden ebenfalls nicht mehr unterstützt. Wenn Sie derzeit einen dieser Anwendungsserver verwenden, müssen Sie ebenfalls auf Apache Tomcat umsteigen. In diesem Fall können Sie jedoch Ihre bisherigen Einstellungen (d. h. die URL für die Anmeldeseite und die Konfigurationsdatei *sessions.xml*) wiederverwenden.

Die einfachste Lösung besteht darin, den Natural Web I/O Interface Client von IIS auf Apache Tomcat zu migrieren. Daher bietet dieses Kapitel IIS-Administratoren eine kurze Einführung in die Installation von Tomcat und beschreibt die Migrationsschritte.

Bevor Sie den Natural Web I/O Interface Client installieren

Wenn Apache Tomcat noch nicht installiert ist, müssen Sie die folgenden Schritte ausführen:

- [Tomcat installieren](#)
- [Java installieren](#)
- [Tomcat-Server starten](#)

Tomcat installieren

Gehen Sie zu <http://tomcat.apache.org/> und laden Sie Tomcat als ZIP-Datei herunter.

Für Benutzer von Microsoft Windows: Laden Sie entweder die 32-Bit- oder die 64-Bit-Windows-ZIP-Datei herunter.

Entpacken Sie die heruntergeladene ZIP-Datei in ein Verzeichnis Ihrer Wahl.

Java installieren

Tomcat basiert auf Java. Daher müssen Sie sicherstellen, dass eine Java-Laufzeitumgebung (JRE) oder ein Java Development Kit (JDK) installiert ist. Die Version der Java-Laufzeitumgebung sollte mindestens Java 6 Update 24 sein. Dies ist die Mindestversion, die für den Natural Web I/O Interface Client auf Tomcat erforderlich ist.

Sie können die Java JRE oder das JDK von der Oracle-Website unter <http://www.oracle.com/technetwork/java/javase/downloads/index.html>.

Wenn Java auf Ihrem System installiert ist, stellen Sie sicher, dass die Umgebungsvariable `JAVA_HOME` auf das Java-Home-Verzeichnis gesetzt ist.

Tomcat-Server starten

Wenn Tomcat und die entsprechende Java-Version installiert sind, können Sie Tomcat starten.

Um Tomcat zu starten, führen Sie die Datei *startup.bat* aus dem Verzeichnis *bin* Ihrer Tomcat-Installation aus. Um zu überprüfen, ob Tomcat ausgeführt wird, geben Sie die folgende URL ein:

```
http://localhost:8080
```

Daraufhin sollte die Standard-Startseite von Tomcat angezeigt werden.

Installieren des Natural Web I/O Interface Client auf Apache Tomcat

Wenn Apache Tomcat installiert wurde, installieren Sie den Natural Web I/O Interface Client wie unter [Installieren des Natural Web I/O Interface Clients auf Apache Tomcat](#) beschrieben.

Konfigurieren des Natural Web I/O Interface Client auf Apache Tomcat

Wenn der Natural Web I/O Interface Client installiert wurde, fahren Sie wie in den folgenden Abschnitten beschrieben fort:

- [Aufrufen der Anmeldeseite](#)
- [Ändern des Tomcat-HTTP-Ports](#)
- [Using the Settings from Your IIS Configuration File](#)
- [Verwendung des Konfigurationstools](#)
- [Schutz des Konfigurationstools vor unbefugtem Zugriff](#)
- [Anmeldeseite standardmäßig anzeigen](#)

Aufrufen der Anmeldeseite

Geben Sie die folgende URL ein, um die Anmeldeseite aufzurufen (diese unterscheidet sich von der URL, die mit IIS verwendet wurde):

```
http://localhost:8080/natuniweb/natural.jsp
```

Ändern des Tomcat-HTTP-Ports

IIS läuft normalerweise auf dem Standardport 80. Wenn Sie möchten, dass Tomcat mit demselben Port arbeitet, bearbeiten Sie die Datei *server.xml*, die sich im Unterverzeichnis *conf* von Tomcat befindet, und suchen Sie dann nach dem folgenden Text:

```
<Connector port="8080" protocol="HTTP/1.1"
```

Ändern Sie die Portnummer so, dass sie wie folgt aussieht:

```
<Connector port="80" protocol="HTTP/1.1"
```

Using the Settings from Your IIS Configuration File

Mit Tomcat können Sie die Konfigurationsdatei *settings.xml* von IIS wiederverwenden, müssen die Datei jedoch in *sessions.xml* umbenennen. Gehen Sie wie folgt vor:

1. Kopieren Sie die Datei *settings.xml* aus Ihrer IIS-Installation in das folgende Verzeichnis Ihrer Tomcat-Installation:

webapps/natuniweb/WEB-INF

2. Benennen Sie entweder die Datei *sessions.xml*, die mit der Installation des Natural Web I/O Interface Clients auf Tomcat geliefert wird, um (z. B. in *sessions-original.xml*) oder löschen Sie sie.
3. Benennen Sie die Datei *settings.xml* in *sessions.xml* um.

Verwendung des Konfigurationstools

Wenn der Natural Web I/O Interface Client auf Tomcat ausgeführt wird, ist es nicht mehr erforderlich, die Konfigurationsdatei manuell zu bearbeiten. Stattdessen können Sie das Konfigurationstool verwenden. Die Verwendung dieses Tools hat den Vorteil, dass Sie keinen ungültigen XML-Code erstellen und somit die XML-Datei beschädigen können. Weitere Informationen finden Sie unter [Verwendung des Konfigurationstools](#).

Die IIS-spezifischen Einträge in der umbenannten Konfigurationsdatei werden ignoriert. Dies sind:

```
natural_parameter visible  
theme  
screen top  
screen left  
screen size  
screen pfkeypos
```

Sie können die Konfigurationsdatei weiterhin manuell bearbeiten. Dies wird jedoch nicht mehr empfohlen.

Schutz des Konfigurationstools vor unbefugtem Zugriff

Es ist möglich, das Konfigurationstool vor unbefugtem Zugriff zu schützen. Ausführliche Informationen finden Sie unter [Konfigurieren der containerverwalteten Security](#).

Ausführliche Informationen zur erforderlichen Realm-Konfiguration für Tomcat finden Sie unter <http://tomcat.apache.org/tomcat-6.0-doc/realm-howto.html>.

Anmeldeseite standardmäßig anzeigen

Wenn Sie die URL `http://localhost:8080/natuniweb` eingeben, zeigt Tomcat die Standardseite des Natural Web I/O Interface Client an, über die Sie entweder auf die [Anmeldeseite](#) oder das [Konfigurationstool](#) des Natural Web I/O Interface Client zugreifen können.



Anmerkung: Wenn Sie einen anderen Port definiert haben (z. B. 80), stellen Sie sicher, dass Sie diese Portnummer in der URL verwenden.

Dieses Verhalten unterscheidet sich von IIS, das standardmäßig die Anmeldeseite anzeigt. Wenn Sie möchten, dass Tomcat ebenfalls standardmäßig die Anmeldeseite anzeigt, bearbeiten Sie die Datei `web.xml`, die sich im Verzeichnis `webapps\natuniweb\WEB-INF` von Tomcat befindet, und suchen Sie nach dem folgenden Eintrag:

```
<welcome-file-list>
  <welcome-file>
    index.html
  </welcome-file>
</welcome-file-list>
```

Ändern Sie den Namen der Welcome-Datei wie im folgenden Beispiel gezeigt in `natural.jsp`:

```
<welcome-file-list>
  <welcome-file>
    natural.jsp
  </welcome-file>
</welcome-file-list>
```


VI

Konfiguration des Client

In diesem Kapitel wird erläutert, wie Sie den Natural Web I/O Interface Client so konfigurieren, dass er in einer Natural-Laufzeitumgebung verwendet werden kann. Folgende Themen werden behandelt:

- Informationen zur Anmeldeseite (Logon Page)
- Natural Client-Konfigurationstool
- Natural Web I/O-Stylesheets
- Starten einer Natural-Anwendung mit einer URL
- Konfigurieren der containerverwalteten Security
- Konfigurieren von SSL
- Protokollierung (Logging)

16

Informationen zur Anmeldeseite (Logon Page)

■ Starten einer Natural-Anwendung über die Anmeldeseite (Logon Page)	106
■ Beispiele für Anmeldeseiten	106
■ Dynamisches Ändern des CICS-Transaktionsnamens, des Hostnamens und der Portnummer beim Starten einer Sitzung	107
■ Angeben eines Passworts auf der Anmeldeseite (Logon Page)	108
■ Ändern des Passworts auf der Anmeldeseite	108
■ Browser-Einschränkungen	109

Starten einer Natural-Anwendung über die Anmeldeseite (Logon Page)

Wenn Sie den Natural Web I/O Interface Client im Browser starten, wird eine Anmeldeseite (Logon Page) angezeigt. Die Einträge auf dieser Anmeldeseite hängen von den Einstellungen in Ihrer [Sitzungskonfiguration](#) ab.

Um eine Natural-Anwendung über die Anmeldeseite zu starten, geben Sie die folgende URL in Ihren Browser ein:

```
http://<host>:<port>/natuniweb/natural.jsp
```

wobei *<host>* und *<port>* der Hostname und die Portnummer Ihres Anwendungsservers sind.

Beispiele für Anmeldeseiten

Für jede Sitzungskonfiguration, die in der Konfigurationsdatei konfiguriert wurde, erscheint ein Eintrag auf der Anmeldeseite. Wenn der Benutzer den entsprechenden Eintrag auswählt, müssen nur diejenigen Parameter, die nicht in der Konfigurationsdatei vorkonfiguriert wurden, auf der Anmeldeseite angegeben werden, um die Anwendung zu starten. In der Regel werden Sie alle Verbindungsparameter außer Benutzername und Passwort vorkonfigurieren.

Das folgende Beispiel zeigt einen Ausschnitt aus einer Anmeldeseite, die aus einer Konfigurationsdatei resultiert, in der keine speziellen Einträge für eine Sitzung definiert sind:

The screenshot shows a web form titled "Enter connection details:". It contains several input fields and two buttons. The fields are: "Session ID:" with a dropdown menu showing "Connect to Natural"; "Natural Server:" with a text input field; "Natural Port:" with a text input field; "User Name:" with a text input field; "Password:" with a text input field; "Natural Application:" with a text input field; and "Natural Parameter:" with a text input field. To the right of the "Session ID" dropdown are two buttons: "Logon" and "Change password".

Das folgende Beispiel zeigt einen Ausschnitt aus einer Anmeldeseite, die aus einer Konfigurationsdatei resultiert, in der viele Einstellungen bereits vordefiniert sind (einschließlich Benutzerkennung und Passwort):

Enter connection details:

Session ID: Demo Application (localhost:2900) Logon

Um sich bei einer Sitzung anzumelden, müssen Sie alle erforderlichen Informationen auf der Anmeldeseite angeben (z. B. wählen Sie eine Sitzung aus der entsprechenden Aufklapp-Liste aus). Wenn Sie auf die Schaltfläche **Connect** (Verbinden) klicken, wird der Bildschirm für die ausgewählte Sitzung angezeigt.

Dynamisches Ändern des CICS-Transaktionsnamens, des Hostnamens und der Portnummer beim Starten einer Sitzung

Die folgende Beschreibung gilt, wenn Sie zu einer anderen CICS-Transaktion unter z/OS wechseln möchten. Es ist auch möglich, einen anderen Host und eine andere Portnummer auszuwählen.

Sie geben die dynamischen Serverparameter in demselben Textfeld an, in dem Sie auch die dynamischen Parameter für die Natural-Umgebung angeben. Damit die dynamischen Serverparameter ausgewertet werden können, ist es wichtig, dass Sie sie vor allen Natural-Parametern mit der folgenden Syntax angeben:

```
<TA_NAME=name TA_HOST=hostname TA_PORT=number>
```

name

Kann 1 bis 4 Zeichen lang sein. Es muss sich um den Namen einer vorhandenen CICS-Transaktion handeln, die für einen CICS-Adapter gilt. Er überschreibt den Transaktionsnamen, der derzeit in der Konfigurationsdatei für den CICS-Adapter auf dem Natural Web I/O Interface Server (NWO) definiert ist. Weitere Informationen erhalten Sie von Ihrem Administrator.

hostname

Kann 1 bis 32 Zeichen lang sein. Es muss sich um den Namen oder die TCP/IP-Adresse des Hosts handeln, auf dem das gewünschte CICS ausgeführt wird. Er überschreibt den Hostnamen (RFE_CICS_TA_HOST), der derzeit in der Konfigurationsdatei für den CICS-Adapter auf dem Natural Web I/O Interface Server (NWO) definiert ist. Weitere Informationen erhalten Sie von Ihrem Administrator.

number

Kann ein numerischer Wert im Bereich von 1 bis 65535 sein. Es handelt sich um den TCP/IP-Port des von CICS bereitgestellten Listeners. Er überschreibt den Hostnamen (RFE_CICS_TA_PORT), der derzeit in der Konfigurationsdatei für den CICS-Adapter auf dem Natural Web I/O Interface Server (NWO) definiert ist. Weitere Informationen erhalten Sie von Ihrem Administrator.

Setzen Sie die gesamte Definition in spitze Klammern. Dynamische Serverparameter können durch ein Komma oder ein Leerzeichen getrennt werden. Es müssen nur die zu überschreibenden Parameter angegeben werden. Wenn auf diese Definition ein Natural-Parameter folgt, müssen Sie vor dem Natural-Parameter ein Leerzeichen einfügen.

Beispiel:

```
<TA_NAME=NA82 TA_HOST=HOST.LOCAL.NET TA_PORT=1898> STACK=(LOGON SYSCP)
```

Wenn einer der dynamischen Serverparameter falsch ist, wird eine Fehlermeldung angezeigt und die Sitzung kann nicht gestartet werden.

Wenn eine falsche CICS-Transaktionskennung angegeben wird, erscheint die Fehlermeldung
EZY1315E 09/03/24 15:17:46 INVALID TRANID=G PARTNER INET ADDR=ip-address.

Wenn ein falscher Hostname oder eine falsche Portnummer angegeben wird, wird die Fehlermeldung NAT9939 Open socket connection failed angezeigt.



Anmerkung: Die oben genannte Definition der dynamischen Serverparameter kann auch im Konfigurationstool angegeben werden, und zwar an derselben Stelle, an der Sie auch die Natural-Parameter angeben, und zusammen mit dem **URL-Parameter** natparam.

Angeben eines Passworts auf der Anmeldeseite (Logon Page)

Die folgenden Informationen gelten, wenn das Feld zur Eingabe eines Passworts auf der Anmeldeseite angezeigt wird. Dieses Feld wird nicht angezeigt, wenn bereits ein Passwort in der Konfigurationsdatei definiert wurde.

Unter Windows und Linux müssen Sie immer das Betriebssystempasswort eingeben, auch wenn Natural Security aktiv ist.

Unter z/OS verhält es sich anders: Wenn Natural Security nicht aktiv ist, müssen Sie das Betriebssystempasswort eingeben. Wenn Natural Security aktiv ist, müssen Sie das Natural Security-Passwort eingeben.

Ändern des Passworts auf der Anmeldeseite

Derzeit ist diese Funktion nur für Natural for Linux und Natural for Windows verfügbar.

Die folgenden Informationen gelten, wenn die Felder zur Eingabe einer Benutzerkennung und eines Passworts auf der Anmeldeseite angezeigt werden. Diese Felder werden nicht angezeigt, wenn die Benutzerkennung und das Passwort bereits in der Konfigurationsdatei definiert sind. In diesem Fall ist es nicht möglich, das Passwort auf der Anmeldeseite zu ändern.

Wenn Sie versuchen, sich mit einem abgelaufenen Passwort anzumelden, wird eine Fehlermeldung angezeigt.

➤ **Um das Passwort zu ändern:**

- 1 Wählen Sie auf der Anmeldeseite die Schaltfläche **Change password** (Passwort ändern).

Der Name dieser Schaltfläche ändert sich zu **Don't change password** (Passwort nicht ändern) und die folgenden zwei Eingabefelder werden auf der Anmeldeseite angezeigt:

- **New password** (Neues Passwort)
- **Repeat new password** (Neues Passwort wiederholen)

- 2 Geben Sie wie gewohnt Ihre Benutzerkennung und Ihr aktuelles Passwort ein.
- 3 Geben Sie das neue Passwort in die beiden Eingabefelder ein.
- 4 Wählen Sie die Schaltfläche **Connect** (Verbinden), um das Passwort zu ändern.

Oder:

Wenn Sie Ihr Passwort nicht ändern möchten, wählen Sie die Schaltfläche **Don't change password**. Die beiden Eingabefelder werden dann ausgeblendet.

Browser-Einschränkungen

Die Schaltflächen **Back** und **Forward** des Browsers funktionieren nicht mit dem Natural Web I/O Interface Client und sollten daher nicht verwendet werden.

Wenn Sie zwei Natural-Sitzungen parallel ausführen möchten, müssen Sie eine neue Instanz des Browsers starten (z. B. durch Auswahl des entsprechenden Symbols in der Schnellstartleiste von Windows). Sie dürfen nicht die Funktion **New Window** (Neues Fenster) des Browsers verwenden. Dies würde dazu führen, dass eine Sitzung in zwei Browsern ausgeführt wird, was nicht zulässig ist.

17

Natural Client-Konfigurationstool

■ Aufrufen des Konfigurationstools	112
■ Session Configuration - Sitzungskonfiguration	114
■ Protokollierungskonfiguration (Logging Configuration-Seite)	122
■ Anmeldeseite (Logon Page)	122
■ Abmelden (Logout)	123

Aufrufen des Konfigurationstools

Der Natural Web I/O Interface Client verfügt über ein Konfigurationstool. Das Konfigurationstool dient zum Erstellen der Sitzungskonfigurationen, die dann auf der Anmeldeseite verfügbar sind. Es kann auch zu Protokollierungszwecken bei Problemen verwendet werden; dies sollte jedoch nur auf Anforderung des Supports erfolgen.

Das Konfigurationstool wird automatisch installiert, wenn Sie den Natural Web I/O Interface Client installieren.

> Um das Konfigurationstool aufzurufen:

- Geben Sie die folgende URL in Ihren Browser ein:

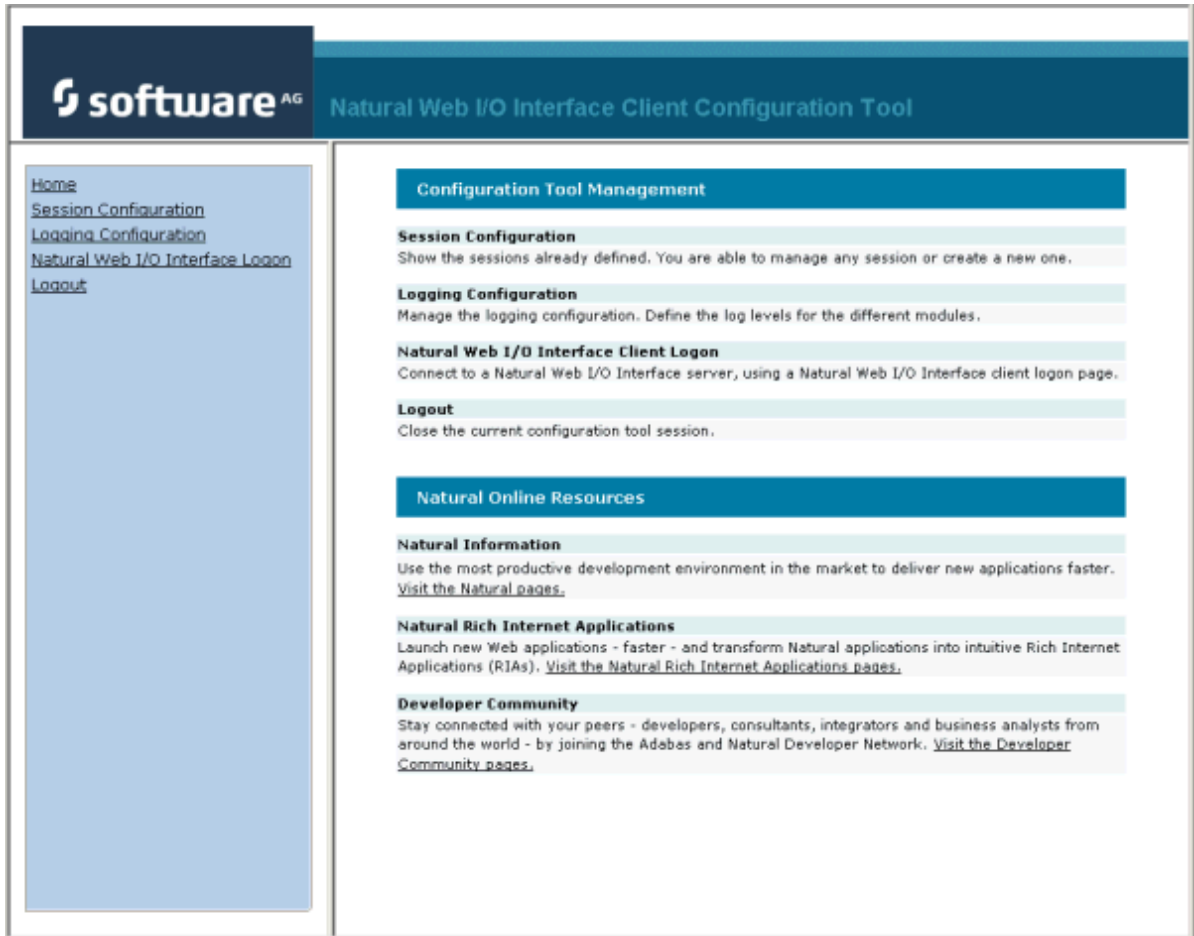
```
http://<host>:<port>/natuniweb/conf_index.jsp
```

wobei *<host>* und *<port>* der Hostname und die Portnummer Ihres Anwendungsservers sind.



Anmerkung: Falls Sie das Konfigurationstool vor unbefugtem Zugriff schützen möchten: Informationen zum Einschränken des Zugriffs auf sensible Bereiche der Anwendungsserverumgebung finden Sie unter *Konfigurieren der containerverwalteten Sicherheit*. Wenn Sie den Zugriff auf das Konfigurationstool eingeschränkt haben, wird ein Authentifizierungsdialogfeld angezeigt. Das Erscheinungsbild dieses Dialogfelds hängt vom ausgewählten Authentifizierungsmodell ab.

Das Konfigurationstool wird angezeigt.



Das Konfigurationstool besteht aus zwei Frames.

Die Startseite des Konfigurationstools wird zunächst im rechten Frame angezeigt. Sie enthält kurze Beschreibungen der Links im linken Frame.

Wenn Sie eine Funktion aufgerufen haben (z. B. wenn Sie gerade die Sitzungskonfiguration anzeigen), können Sie jederzeit den Link **Home** im linken Frame auswählen, um zur Startseite des Konfigurationstools zurückzukehren.

Die Funktionen, die über die anderen Links im linken Frame aufgerufen werden, werden im Folgenden beschrieben.

Session Configuration - Sitzungskonfiguration

In diesem Abschnitt wird beschrieben, wie Sie den Inhalt der Konfigurationsdatei für die Sitzung verwalten. Folgende Themen werden behandelt:

- [Aufrufen der Seite Session Configuration](#)
- [Global Settings - Globale Einstellungen](#)
- [Hinzufügen einer neuen Sitzung](#)
- [Bearbeiten einer Sitzung](#)
- [Übersicht über die Session-Optionen](#)
- [Duplizieren einer Sitzung](#)
- [Löschen einer Sitzung](#)
- [Hinzufügen eines neuen Benutzers](#)
- [Speichern der Konfiguration](#)

Aufrufen der Seite Session Configuration

Der Inhalt der Konfigurationsdatei für die Sitzungen wird auf der Seite **Session Configuration** verwaltet.

Um die Seite **Session Configuration** aufzurufen, müssen Sie den Link **Session Configuration** im linken Frame wählen.

Die Seite **Session Configuration** wird im rechten Frame angezeigt. Sie zeigt die globalen Einstellungen und listet alle derzeit definierten Sitzungen und Benutzer auf. Zu einer Sitzung werden einige der Informationen aus der Konfigurationsdatei angezeigt. Beispiel:

Session Configuration

Save Configuration

Global Settings

Last activity timeout (n seconds):

Trace directory:

SSL trust file path:

SSL trust file password:

Sessions

Session ID	Host Name	Port Number	Application	Natural Parameters	Edit	Duplicate	Delete
Connect to Natural					Edit	Duplicate	Delete
localtestserver	localhost	6640			Edit	Duplicate	Delete

Add New Session

Users

User ID	Edit	Duplicate	Delete
user1	Edit	Duplicate	Delete

Add New User

Global Settings - Globale Einstellungen

Die globalen Einstellungen gelten für alle definierten Sitzungen. Sie können die folgenden globalen Einstellungen in der Konfigurationsdatei definieren:

Option	Beschreibung
Last activity timeout (n seconds)	Zeitlimit für letzte Aktivität (n Sekunden). Die Anzahl der Sekunden, die der Client auf die nächste Benutzeraktivität wartet. Wenn die definierte Anzahl von Sekunden ohne Benutzeraktivität erreicht ist, wird die Sitzung geschlossen. Der Standardwert beträgt 3600 Sekunden.
Trace directory	Trace-Verzeichnis. Optional. Speicherort eines anderen Trace-Verzeichnisses. Optional. Wenn kein anderes Trace-Verzeichnis definiert ist, werden die Trace-Dateien in das Standard-Trace-Verzeichnis geschrieben. Standardmäßig werden die Trace-Dateien in das Verzeichnis geschrieben, das durch die Java-Property <code>java.io.tmpdir</code> festgelegt wurde. Unter Windows ist dies normalerweise die Umgebungsvariable TMP für den Benutzer, der den Anwendungsserver gestartet hat. Unter Linux ist dies normalerweise <code>/tmp</code> oder <code>/var/tmp</code>. Sie können diese Property auch im Startskript für den Anwendungsserver festlegen.

Option	Beschreibung
	Das Tracing kann für jede Session einzeln aktiviert werden (siehe Übersicht über die Session-Optionen weiter unten). Es sollte jedoch nur auf Anforderung des Supports aktiviert werden.
SSL trust file path	Optional. Der Pfad zu Ihrer vertrauenswürdigen Datei (Trust File). Weitere Informationen finden Sie unter Konfigurieren von SSL .
SSL trust file password	Wenn Ihr Trust File passwortgeschützt ist, müssen Sie das entsprechende Passwort angeben. Wenn Sie kein Passwort für eine passwortgeschützte vertrauenswürdige Datei angeben, kann die vertrauenswürdige Datei nicht geöffnet werden und es ist daher nicht möglich, eine SSL-Sitzung zu öffnen. Wenn Ihr Trust File nicht passwortgeschützt ist, sollten Sie kein Passwort angeben.

Hinzufügen einer neuen Sitzung

Sie können der Konfigurationsdatei eine neue Sitzung hinzufügen.

➤ Um eine neue Sitzung hinzuzufügen:

- 1 Wählen Sie die Schaltfläche **Add New Session**.

Die Seite **Edit Session** erscheint.
- 2 Geben Sie alle erforderlichen Informationen ein, wie unten im Abschnitt [Übersicht über die Session-Optionen](#) beschrieben
- 3 Wählen Sie die Schaltfläche **OK**, um zur Seite **Session Configuration** zurückzukehren.

Die neue Sitzung ist noch nicht in der Konfigurationsdatei verfügbar.
- 4 Wählen Sie die Schaltfläche **Save Configuration**, um die neue Sitzung in die Konfigurationsdatei zu schreiben.

Bearbeiten einer Sitzung

Sie können jede vorhandene Sitzung in der Konfigurationsdatei bearbeiten.

➤ Um eine Session zu bearbeiten:

- 1 Wählen Sie den Link **Edit** neben der Sitzung, die Sie bearbeiten möchten.

Die Seite **Edit Session** wird angezeigt.
- 2 Geben Sie alle erforderlichen Informationen ein, wie unten im Abschnitt [Übersicht über die Session-Optionen](#) beschrieben.
- 3 Wählen Sie die Schaltfläche **OK**, um zur Seite **Session Configuration** zurückzukehren.

Die Änderungen sind noch nicht in der Konfigurationsdatei verfügbar.

- 4 Wählen Sie die Schaltfläche **Save Configuration**, um die Änderungen in die Konfigurationsdatei zu schreiben.

Übersicht über die Session-Optionen

Die Seite **Edit Session** wird angezeigt, wenn Sie

- eine neue Sitzung **hinzufügen** (Add) oder
- eine vorhandene Sitzung **bearbeiten** (Edit).

Beispiel:

Edit Session

Session ID:	
Type:	Undefined
Host name:	
Port number:	
Use SSL:	☐ Yes ☒ No
User name:	
User name in upper case:	☐ Yes ☒ No
Password:	
Application:	
Natural parameters:	
Screen rows:	24
Screen columns:	80
Show function key numbers:	☐ Yes ☒ No
Trace:	☐ Yes ☒ No
Timeout (n seconds):	60

OK Cancel

Auf der Seite **Edit Session** stehen folgende Optionen zur Verfügung:

Option	Description
Session ID	Sitzungskennung. Pflichtangabe. Ein Sitzungsname Ihrer Wahl. Auf der Anmeldeseite wird der Sitzungsname in einem Dropdown-Listenfeld angezeigt.
Type	Die Plattform, auf der die Benutzererkennung (User-ID) und das Passwort authentifiziert werden. Sie können die gewünschte Einstellung aus dem Dropdown-Listenfeld auswählen. ■ Undefined Nicht definiert. Standardeinstellung. Benutzererkennung und Passwort können maximal 32 Zeichen lang sein. Siehe auch die Beschreibung für Natural for Windows or Linux weiter unten. ■ Natural for z/OS Die Angaben für Benutzererkennung und Passwort dürfen maximal 8 Zeichen lang sein. ■ Natural for z/OS with Natural Security Die Angaben für Benutzername und Passwort dürfen maximal 8 Zeichen lang sein. Der Benutzername muss den Natural-Namenskonventionen für Bibliotheksnamen entsprechen. Siehe <i>Namenskonventionen für Libraries</i>. ■ Natural for Windows or Linux Die Angaben für Benutzername und Passwort dürfen maximal 32 Zeichen lang sein. Wenn eine Domäne erforderlich ist, müssen Sie diese zusammen mit dem Benutzernamen angeben (in der Form " <i>domain \ user-ID</i> ").
Host name	Der Name oder die TCP/IP-Adresse des Servers, auf dem Natural und der Natural Web I/O Interface Server ausgeführt werden. Wird dieser hier angegeben, wird das entsprechende Feld auf der Anmeldeseite nicht angezeigt.
Port number	Die Nummer des TCP/IP-Ports, auf dem der Natural Web I/O Interface Server empfangsbereit ist. Wird diese Nummer hier angegeben, wird das entsprechende Feld auf der Anmeldeseite nicht angezeigt.
Use SSL	Wenn diese Option auf Yes gesetzt ist, wird eine sichere Verbindung zwischen dem Natural Web I/O Interface Client auf dem Anwendungsserver und dem Natural Web I/O Interface Server hergestellt. Wichtig: Wenn Sie SSL mit Natural for z/OS verwenden möchten, muss einer der entsprechenden Typen ausgewählt werden. Der Typ darf nicht Undefined oder Natural for Windows or Linux sein. Umgekehrt gilt: Wenn Sie SSL mit Natural for Windows oder Linux verwenden möchten, dürfen Sie keinen der plattformspezifischen Typen auswählen. Der Typ kann in diesem Fall auch Undefined sein.
User name	Optional. Eine gültige Benutzererkennung für den aktuellen Rechner. Wird diese hier angegeben, erscheint das entsprechende Feld nicht auf der Anmeldeseite.
User name in upper case	Benutzername in Großbuchstaben. Wird diese Option markiert, befindet sich das Eingabefeld für die Benutzererkennung im Großbuchstabenmodus.

Option	Description
Password	Optional. Ein gültiges Passwort für die oben erwähnte Benutzerkennung. Unter Windows und Linux ist dies immer das Betriebssystem-Passwort des Benutzers, auch wenn Natural Security aktiv ist. Unter z/OS ist dies anders: Wenn Natural Security nicht aktiv ist, ist dies das Betriebssystem-Passwort des Benutzers. Wenn Natural Security aktiv ist, ist dies das Natural Security-Passwort. Wenn hier ein Passwort angegeben wird, erscheint das entsprechende Feld nicht auf der Anmeldeseite. Das Konfigurationstool speichert das Passwort in verschlüsselter Form.
Application	Anwendung: ■ Natural for z/OS Der Name des Natural-Programms oder einer Kommandosequenz, die Ihre Anwendung startet, wie Sie sie an der NEXT-Eingabeaufforderung eingeben würden. Beispiel: <code>TEST01 data1,data2</code> ■ Natural for Linux Der Name des Linux-Shell-Skripts zum Starten der Natural-Anwendung (eine Datei ähnlich wie <i>nwo.sh</i>). ■ Natural for Windows Der Name der Windows-Kommandodatei (<i>.bat</i>) zum Starten der Natural-Anwendung. Wird der Name hier angegeben, erscheint das entsprechende Feld nicht auf der Anmeldeseite.
Natural parameters	Optional. Parameter zum Starten der Natural-Anwendung. Dies können STACK-Parameter, eine Parameterdatei bzw. ein Parametermodul oder andere Natural-spezifische Informationen sein. ■ Natural for z/OS Wird verwendet, um dynamische Natural-Profilparameter an die Sitzung zu übergeben, zum Beispiel: <code>SYSPARM=(MYPARMS) STACK=(LOGON MYAPPL)</code> Anmerkung: Es wird empfohlen, den Namen des Natural-Programms, das die Anwendung startet, über die Option Application anzugeben, anstatt ihn mit dem Profilparameter STACK zu übergeben. ■ Natural for Linux and Natural for Windows Wird verwendet, wenn das oben erwähnte Shell-Skript (Linux) oder die Kommandodatei (Windows) den Parameter \$5 nach "natural" verwendet, zum Beispiel: <code>PARM=MYPARM STACK=(LOGON MYLIB;MENU)</code>

Option	Description
Double-click behavior	Doppelklick-Verhalten. Die Taste, die beim Doppelklicken auf ein Ausgabefeld emuliert werden soll. Standardmäßig ist dies die ENTER-Taste. Sie können das Doppelklick-Verhalten deaktivieren oder eine Funktionstaste (PF1 bis PF12) definieren. You can select the required setting from the drop-down list box. Sie können die gewünschte Einstellung aus der Dropdown-Liste auswählen. Tipp: Wenn für die Ausgabefelder eine kontextsensitive Hilfe definiert wurde, kann es sinnvoll sein, PF1 zu definieren. Die Hilfefunktion wird dann aufgerufen, wenn der Benutzer auf ein Ausgabefeld doppelklickt.
Screen rows	Bildschirmzeilen. Die Anzahl der Zeilen im Ausgabefenster. Mögliche Werte: mindestens 24, keine Obergrenze. Standardeinstellung: 24. Wird von Natural for z/OS nicht verwendet, da dort stattdessen der Profilparameter TMODEL verwendet wird.
Screen columns	Bildschirmspalten. Die Anzahl der Spalten im Ausgabefenster. Mögliche Werte: mindestens 80, keine Obergrenze. Standardeinstellung: 80. Wird von Natural for z/OS nicht verwendet, da dort stattdessen der Profilparameter TMODEL verwendet wird.
Show function key numbers	Funktionstastennummern anzeigen. Wenn auf Yes gesetzt, werden die PF-Tastennummern neben den PF-Tasten angezeigt.
Trace	Nachverfolgung. Sollte nur auf Yes gesetzt werden, wenn dies vom Support angefordert wird.
Check for numeric input	Auf numerische Eingabe prüfen Wenn auf Yes (Standardeinstellung) gesetzt, werden numerische Eingabefelder validiert. In diesem Fall sind in numerischen Eingabefeldern nur die folgenden Zeichen zulässig (zusätzlich zu den Ziffern 0 bis 9): <i>leer</i> + (Plus) - (Minus) _ (underscore) , (Komma) . (Punkt) ? (Fragezeichen) Wenn auf No gesetzt, werden numerische Eingabefelder nicht validiert.
Timeout (in seconds)	Zeitlimit (in Sekunden). Die Anzahl der Sekunden, die der Client auf eine Antwort wartet, nachdem eine aktualisierte Seite an die Natural-Sitzung gesendet wurde. Wenn die festgelegte Anzahl von Sekunden ohne Antwort erreicht ist, wird die Sitzung geschlossen. Der Standardwert beträgt 60 Sekunden. Normalerweise brauchen Sie diesen Wert nicht zu ändern.
Filler character	Füllzeichen. Optional. Das Füllzeichen, das aus den Eingabefeldern entfernt werden soll. Eine Anwendung kann beispielsweise einen Unterstrich (_) als Füllzeichen definieren.

Option	Description
	Nachgestellte Füllzeichen werden aus den Eingabefeldern entfernt, und vorangestellte Füllzeichen werden durch Leerzeichen ersetzt.

Duplizieren einer Sitzung

Sie können eine Kopie einer beliebigen vorhandenen Sitzung zur Konfigurationsdatei hinzufügen.

» Um eine Sitzung zu duplizieren:

- 1 Wählen Sie den Link **Duplicate** neben der Sitzung, die Sie duplizieren möchten.

Am Ende der Liste der Sitzungen wird ein neuer Eintrag angezeigt. Sein Name lautet "Copy of *session-ID*". Die duplizierte Session ist noch nicht in der Konfigurationsdatei verfügbar.

- 2 Bearbeiten (**Edit**) und speichern Sie die duplizierte Sitzung wie oben beschrieben.

Löschen einer Sitzung

Sie können jede vorhandene Sitzung aus der Konfigurationsdatei löschen.

» Um eine Sitzung zu löschen:

- 1 Wählen Sie den Link **Delete** neben der Sitzung, die Sie löschen möchten.

Die Sitzung wird aus der Liste der Sitzungen gelöscht. Sie ist noch nicht aus der Konfigurationsdatei gelöscht.

- 2 Wählen Sie die Schaltfläche **Save Configuration**, um die Sitzung aus der Konfigurationsdatei zu löschen.

Hinzufügen eines neuen Benutzers

Sie können Natural-Benutzer und deren Passwörter in der Konfigurationsdatei vordefinieren.

Wenn eine Natural-Seite mit einer URL geöffnet wird, die einen Benutzer im URL-Parameter `natuser` angibt, wird der angegebene Benutzer mit der Liste der Benutzer in der Konfigurationsdatei abgeglichen. Wenn der angegebene Benutzer in der Konfigurationsdatei definiert ist, wird das entsprechende Passwort zur Authentifizierung des Benutzers beim Start der Natural-Sitzung verwendet. Siehe auch [Starten einer Natural-Anwendung mit einer URL](#).

Beispiel: Wenn die folgende URL verwendet wird, wird das für `user1` definierte Passwort verwendet:

`http://myhost:8080/natuniweb/natural.jsp?natuser=user1...`

➤ **Um einen neuen Benutzer hinzuzufügen:**

- 1 Wählen Sie die Schaltfläche **Add New User**.

Die Seite **Edit User** wird angezeigt.

- 2 Geben Sie einen Benutzernamen und ein Passwort ein.
- 3 Wählen Sie die die Schaltfläche **OK**, um zur Seite **Session Configuration** zurückzukehren.

Der neue Benutzer ist noch nicht in der Konfigurationsdatei verfügbar.

- 4 Wählen Sie die die Schaltfläche **Save Configuration**, um den neuen Benutzer in die Konfigurationsdatei zu schreiben.



Anmerkung: Sie können einen Benutzer auf die gleiche Weise wie eine Sitzung bearbeiten, duplizieren und löschen (siehe die entsprechenden Beschreibungen oben).

Speichern der Konfiguration

Wenn Sie die Schaltfläche **Save Configuration** wählen, werden alle Ihre Änderungen in die Konfigurationsdatei geschrieben. Der Server übernimmt die neuen Einstellungen automatisch, wenn er das nächste Mal Daten aus der Konfigurationsdatei liest.



Vorsicht: Wenn Sie nicht die Schaltfläche **Save Configuration** wählen, sondern sich stattdessen abmelden oder das Konfigurationstool durch Eingabe einer anderen URL verlassen, werden die neuen Einstellungen nicht in die Konfigurationsdatei geschrieben.

Protokollierungskonfiguration (Logging Configuration-Seite)

Der Inhalt der Konfigurationsdatei für die Protokollierung wird über die Seite **Logging Configuration** verwaltet. Ausführliche Informationen finden Sie im Abschnitt [Protokollierung \(Logging\)](#).

Anmeldeseite (Logon Page)

Das Konfigurationstool zeigt im linken Frame den folgenden Link an:

Natural Web I/O Interface Logon

Dieser Link öffnet die Anmeldeseite im rechten Frame.

Die Anmeldeseite verwendet die aktuellen Einstellungen in der Konfigurationsdatei. Wenn Sie eine Sitzung aus der Dropdown-Liste auswählen, können Sie überprüfen, ob die Verbindungsdetails

wie gewünscht angezeigt werden. Ist dies nicht der Fall, können Sie zur Session Configuration zurückkehren und die Einstellungen der entsprechenden Sitzung ändern.

Siehe auch [Informationen zur Anmeldeseite \(Logon Page\)](#) .

Abmelden (Logout)

Wenn das Konfigurationstool vor unbefugtem Zugriff geschützt ist und Sie sich vom Konfigurationstool abmelden, können Sie sicherstellen, dass kein anderer Benutzer die Clientkonfiguration ändern kann, wenn Sie Ihren PC für eine Weile unbeaufsichtigt lassen.

➤ Um sich abzumelden:

- Wählen Sie im linken Frame den Link **Logout**.

Wenn das Konfigurationstool vor unbefugtem Zugriff geschützt ist, wird wieder der Authentifizierungsdialog angezeigt.

Wenn es nicht geschützt ist, wird die Homepage erneut angezeigt.

18

Natural Web I/O-Stylesheets

■ Name und Speicherort der Stylesheets	126
■ Stylesheets bearbeiten	126
■ Position der Hauptausgabe (Main Output) und der PF-Tasten ändern	126
■ Schriftgröße ändern (Font Size)	127
■ Schriftart ändern (Font Type)	128
■ Unterstrichenen und blinkenden Text definieren	129
■ Kursiven Text definieren (Italic Text)	130
■ Fettgedruckten Text definieren (Bold Text)	130
■ Verschiedene Stile für Ausgabefelder definieren	131
■ Natural-Fenster ändern	131
■ Meldungszeile ändern (Message Line)	132
■ Hintergrundfarbe ändern (Background Color)	132
■ Farbattribute ändern (Color Attributes)	133
■ Stil der PF-Tasten ändern	134
■ JavaScript und XSLT-Dateien	135

Name und Speicherort der Stylesheets

Verschiedene Aspekte einer Seite (z. B. Schriftart, Schriftstil oder -farbe) werden durch eine als Stylesheet (CSS-Datei) bezeichnete Stilvorlage gesteuert.

Der Natural Web I/O Interface Client wird mit dem Stylesheet *3270.css* ausgeliefert, das sich an folgendem Speicherort befindet:

`../natuniapp.ear/natuniweb.war/resources`



Anmerkung: Weitere Informationen zu Stylesheets finden Sie unter <http://www.w3.org/Style/CSS/>.

Stylesheets bearbeiten

Sie sollten über grundlegende Kenntnisse von CSS-Dateien verfügen.

Sie können die vordefinierten Stylesheets bearbeiten oder eigene Stylesheets erstellen.

Es wird empfohlen, mit Sicherungskopien zu arbeiten. Wenn ein Problem mit Ihrem Stylesheet auftritt, können Sie so jederzeit zum ursprünglichen Zustand zurückkehren.

Um Ihre Änderungen im Browser zu sehen, müssen Sie

1. den Cache des Browsers löschen und
2. die Sitzung neu starten.

Position der Hauptausgabe (Main Output) und der PF-Tasten ändern

Gilt nur, wenn nur mit Namen belegte PF-Tasten angezeigt werden. Diese Funktion kann nicht verwendet werden, wenn alle PF-Tasten angezeigt werden, da diese immer an derselben Position angezeigt werden. Siehe auch [Übersicht über die Session-Optionen](#).

Die folgenden Elemente sind verfügbar:

Elementname	Beschreibung
.mainlayer	Steuert die Position der Hauptausgabe im Ausgabefenster. Wird für Sprachen verwendet, die von links nach rechts geschrieben werden (LTR).
.mainlayer_rtl	Steuert die Position der Hauptausgabe im Ausgabefenster. Wird für Sprachen verwendet, die von rechts nach links geschrieben werden (RTL).
.pfkeydiv	Wird für Sprachen verwendet, die von links nach rechts geschrieben werden (LTR).
.pfkeydiv_rtl	Steuert die Position der PF-Tasten im Ausgabefenster. Wird für Sprachen verwendet, die von rechts nach links geschrieben werden (RTL).

Die *_rtl-Elemente werden nur verwendet, wenn Natural den Web-Ein-/Ausgabe-Bildschirm mit einem Rechts-nach-links-Flag (SET CONTROL 'VON') sendet. Im Browser werden die Bildschirm-elemente dann auf der rechten Seite (anstelle der linken Seite) angezeigt.

Für Web-Ein-/Ausgabe in Anwendungen, in denen nur die Ausrichtung von links nach rechts verwendet wird, sind die *_rtl-Elemente nicht erforderlich.

Wenn die PF-Tasten unten angezeigt werden sollen, müssen Sie die Elemente wie im folgenden Beispiel gezeigt definieren:

```
/* Defines the main screen position */ .mainlayer { top: 5px;
  left: 0px;
  height: 550px; } /* Defines the main screen position for right-to-left */
.mainlayer_rtl { top: 5px;
  right: 30px;
  height: 550px; } /* Defines the PF keys screen position */ .pfkeydiv { height: 70px;
  left: 0px;
  top: 580px; width: 100%; } /* Defines the PF keys screen position for
right-to-left */ .pfkeydiv_rtl { height: 70px;
  right: 30px;
  top: 580px; width: 100%; }
```

Schriftgröße ändern (Font Size)

Je nach Bildschirmauflösung wird zusätzlich zum Standard-Stylesheet eines der folgenden Stylesheets zur Definition der Schriftgröße verwendet:

- *model2.css*
- *model3.css*
- *model4.css*
- *model5.css*

Diese Stylesheets befinden sich im Unterverzeichnis *tmallows* des Verzeichnisses *resources*, in dem alle Stylesheets gespeichert sind.

Je nachdem, was dem Standard 3270 Screen Model am nächsten kommt, wird automatisch das entsprechende Stylesheet aus dem Unterverzeichnis *tmodels* verwendet. Die Auswahl erfolgt nach folgenden Kriterien:

Standard 3270 Screen Model	Kriterien	Stylesheet
Model 2 (80x24)	30 Zeilen oder weniger.	<i>model2.css</i>
Model 3 (80x32)	Zwischen 31 und 40 Zeilen.	<i>model3.css</i>
Model 4 (80x43)	41 Zeilen oder mehr.	<i>model4.css</i>
Model 5 (132x27)	30 Zeilen oder weniger und mehr als 100 Spalten.	<i>model5.css</i>

Die Schriftgrößen in den oben genannten Stylesheets können angepasst werden. Beispiel für *model4.css*:

```
body { font-size: 10px; }
```

Die Standardschriftgrößen für die oben genannten 3270 Screen Models sind:

Standard 3270 Screen Model	Standardschriftgröße
Model 2	16px
Model 3	14px
Model 4	10px
Model 5	12px

Schriftart ändern (Font Type)

Generell sollten Sie nur Monospace-Schriftarten wie Courier New oder Lucida Console verwenden. Bei diesen Schriftarten haben alle Zeichen die gleiche Breite. Bei Verwendung von Schriftarten mit variabler Breite erscheint die Ausgabe sonst verzerrt.

Wenn Sie eine anderen Schriftart definieren möchten, sollten Sie für den Textkörper, die Ausgabefelder und die Eingabefelder dieselbe Schriftart definieren, wie im folgenden Beispiel gezeigt:

```
body {
  background-color: #F3F5F0;
  font-family: Lucida Console;
}

.OutputField {
  white-space:pre;
  border-width:0;
  font-family: Lucida Console;
  font-size: 100%;
}

.InputField {
  background-color: white;
  font-family: Lucida Console;
  border-width: 1px;
  font-size: 100%;
  border-color: #A7A9AB;
}
```

Verwenden Sie die CSS-at-rule `@font-face`, um eine benutzerdefinierte Schriftart für die Anzeige von Text festzulegen.

Beispiel: Arabische Schriftarten, die mit `InputFields` für die Darstellung von arabisch-indischen Ziffern anstelle von europäischen Ziffern verwendet werden sollen:

```
@font-face {
  font-family: CustomFont;
  src: url( CustomFont.woff );
}
```

Unterstrichenen und blinkenden Text definieren

Die folgenden Elemente sind verfügbar:

Elementname	Beschreibung
<code>.natTextDecoUnderline</code>	Definiert unterstrichenen Text.
<code>.natTextDecoBlinking</code>	Definiert blinkenden Text.
<code>.natTextDecoNormal</code>	Definiert normalen Text (ohne Unterstreichung, ohne Blinken).

Beispiel:

```
/* Text decoration */
.natTextDecoUnderline { text-decoration:underline; }
.natTextDecoBlinking {text-decoration:blink; }
.natTextDecoNormal {text-decoration:normal;}
```

Blinkender Text wird vom Internet Explorer nicht unterstützt.

Kursiven Text definieren (Italic Text)

Die folgenden Elemente sind verfügbar:

Elementname	Beschreibung
.natFontStyleItalic	Definiert kursiven Text.
.natFontStyleNormal	Definiert normalen Text (nicht kursiv).

Beispiel:

```
/* font style */
.natFontStyleItalic {font-style:italic;}
.natFontStyleNormal {font-style:normal;}
```

Fettgedruckten Text definieren (Bold Text)

Die folgenden Elemente sind verfügbar:

Elementname	Beschreibung
.natFontWeightBold	Definiert fettgedruckten Text.
.natFontWeightNormal	Definiert normalen Text (nicht fettgedruckt).

```
/* Font weight */
.natFontWeightBold {font-weight:bolder;}
.natFontWeightNormal {font-weight:normal;}
```

Wenn Sie fettgedruckten Text ({font-weight:bolder;}) für die Standardschriftart Courier New definieren, hat Ihr Text immer die gleiche Breite wie normaler Text ({font-weight:normal;}).

Wenn Sie jedoch fettgedruckten Text für Courier oder Lucida Console definieren, ist der fettgedruckte Text breiter als der normale Text, sodass Ihre Ausgabe möglicherweise deformiert erscheint. Es wird daher empfohlen, fettgedruckten Text bei Courier und Lucida Console zu deaktivieren:

```
.natFontWeightBold {font-weight:normal;}
```

Verschiedene Stile für Ausgabefelder definieren

Die folgenden Elemente sind verfügbar:

Elementname	Beschreibung
.FieldVariableBased	Definiert den Stil für Ausgabefelder, die auf einer Variablen basieren.
.FieldLiteralBased	Definiert den Stil für Ausgabefelder, die auf einem Literal basieren.

Beispiel:

```
.FieldVariableBased {
    /* font-style:italic; */
}

.FieldLiteralBased {
    /* font-style:normal; */
}
```



Anmerkung: Im obigen Beispiel sowie in den vom Produkt gelieferten Standard-CSS-Dateien sind die variablenbasierten Ausgabefelder als kursiv definiert, jedoch auskommentiert.

Natural-Fenster ändern

Die folgenden Elemente sind verfügbar:

Elementname	Beschreibung
.naturalwindow	Steuert die Darstellung der Natural-Fenster.
.wintitle	Steuert die Darstellung der Titel der Natural-Fenster.

Beispiel:

```
.naturalwindow {
    border-style: solid;
    border-width: 1px;
    border-color: white;
    background-color: black;
}

.wintitle {
    left: 0px;
```

```
top: 1px;
height: 17px;
width: 100%;
color: black;
font-size: 100%;
font-weight: bold;
background-color: white;
text-align: center;
font-family: Verdana;
border-bottom-style: solid;
border-bottom-width: 2px;
}
```



Anmerkung: In einer z/OS-Umgebung müssen Sie den Natural-Profilparameter `WEBIO` entsprechend einstellen, um diese Funktion zu aktivieren. Siehe *WEBIO – Web I/O Interface Rendering* in der *Parameter-Referenz-Dokumentation* für Natural for z/OS.

Meldungszeile ändern (Message Line)

Die Darstellung der Meldungszeile wird durch das Element `.MessageLine` gesteuert.

Beispiel:

```
.MessageLine {
  color: blue;
}
```



Anmerkung: In einer z/OS-Umgebung müssen Sie den Natural-Profilparameter `WEBIO` entsprechend einstellen, um diese Funktion zu aktivieren. Siehe *WEBIO – Web I/O Interface Rendering* in der *Parameter-Referenz-Dokumentation* für Natural for z/OS.

Hintergrundfarbe ändern (Background Color)

Die Hintergrundfarbe wird im Element `body` definiert.

Beispiel:

```
body {
  background-color: #F3F5F0;
  font-family: Lucida Console;
}
```

Farbattribute ändern (Color Attributes)

Sie können für alle Natural-Farbattribute unterschiedliche Farben definieren. Diese sind:

Rot
 Grün
 Blau
 Gelb
 Weiß
 Schwarz
 Rosa
 Türkis
 Transparent

Sie können diese Farbattribute für Eingabe- und Ausgabefelder sowie für normale Ausgabe und inverse Anzeige definieren.

Die folgenden Beispiele zeigen, wie Sie das Farbattribut „Rot“ definieren.

Definiert die Farbe für ein normales Ausgabefeld:

```
.natOutputRed {color: darkred;}
```

Definiert die Vordergrund- und Hintergrundfarben für ein Ausgabefeld mit inverser Anzeige:

```
.reverseOutputRed {background-color: darkred; color:#F3F5F0;}
```

Definiert die Farbe für ein normales Eingabefeld:

```
.natInputRed {color: darkred;}
```

Definiert die Vordergrund- und Hintergrundfarben für ein Eingabefeld mit inverser Anzeige:

```
.reverseInputRed {background-color: darkred; color:#F3F5F0;}
```

Stil der PF-Tasten ändern

Die folgenden Elemente sind verfügbar:

Elementname	Beschreibung
.PFButton	Steuert den Stil für die normale Darstellung.
.PFButton:hover	Steuert den Stil, der verwendet wird, wenn der Mauszeiger über eine PF-Taste bewegt wird.

Beispiel:

```
.PFButton {
  text-align: center;
  width: 90px;
  border-style: ridge;
  border-width: 3px;
  padding: 2px;
  text-decoration: none;
  font-family: Verdana;
  font-size: 12px;
  height: 22px;
}

.PFButton:hover {
  color: #FFFF00;
  background-color: #222222;
}
```



Anmerkung: In einer z/OS-Umgebung müssen Sie den Natural-Profilparameter WEBIO entsprechend einstellen, um diese Funktion zu aktivieren. Siehe *WEBIO – Web I/O Interface Rendering* in der *Parameter-Referenz-Dokumentation* für Natural for z/OS.

JavaScript und XSLT-Dateien

Informationen zu XSLT-Dateien im Natural Web I/O Interface

Zusätzlich zu den oben beschriebenen CSS-Dateien verwendet der Natural Web I/O Interface Client XSLT-Dateien mit bestimmten Namen, um die Natural Web I/O Interface-Bildschirme vom internen XML-Format in HTML zu konvertieren. Das HTML enthält auch Aufrufe an JavaScript. Für Web I/O-Bildschirme wird folgende Konvertierung durchgeführt:

- Eingabetext wird in das HTML-Element `<input>` eingefügt.
- Ausgabertext wird in das HTML-Element `<input>` (mit dem Attribut `readonly="readonly"`) eingefügt.
- Eine Meldungszeile wird in das HTML-Element `<span>` eingefügt.
- PF-Tasten werden in eine XML-Insel eingebettet und dann mit JavaScript gerendert.
- Fensterelemente werden in eine XML-Insel eingebettet und dann mit JavaScript gerendert.

Die Konvertierung wird zur Laufzeit durch die folgenden Produktdateien Ihrer Webanwendung durchgeführt:

1. `<mywebapp>/WEB-INF/transuni.xml`
2. `<mywebapp>/scripts/natuniscript.js`

Die XSLT-Datei *transuni.xml* wird nur einmal beim Start des Servers gelesen.



Wichtig: Ändern Sie die oben genannten Dateien nicht. Die Funktionalität dieser Dateien kann sich in neuen Versionen oder Service Packs des Produkts ändern, wodurch Ihre Änderungen überschrieben würden.

JavaScript anpassen

Sie können Ihre eigene JavaScript-Datei mit erweiterter JavaScript-Funktionalität in das Verzeichnis `<mywebapp>/scripts` kopieren. Diese Datei muss den Namen *usernaticscript.js* haben. Definieren Sie Ihre neue Funktionalität in dieser JavaScript-Datei.

Wenn beim Start des Servers eine Datei *usernaticscript.js* gefunden wird, wird diese zusätzlich zur JavaScript-Datei *natuniscript.js* gelesen.

Damit die neuen JavaScript-Funktionen ausgeführt werden können, müssen Sie möglicherweise auch die [XSLT-Datei anpassen](#).

XSLT anpassen

Erstellen Sie eine Kopie von `<mywebapp>/WEB-INF/transuni.xml` und speichern Sie sie unter `<mywebapp>/WEB-INF/usertransuni.xml`. Passen Sie die XSL-Elemente an Ihre individuellen Anforderungen an.

Nachdem Sie Änderungen an der Benutzerdatei `usertransuni.xml` vorgenommen haben, müssen Sie den Server neu starten, damit Ihre Änderungen wirksam werden. Wenn beim Starten des Servers eine Datei `usertransuni.xml` gefunden wird, wird diese anstelle der standardmäßig vorhandenen XSLT-Datei gelesen.

Überschreib-/Einfügemodus (Overwrite / Insert) in Eingabefeldern anpassen

Bis NJX 9.1.1 verwendete Internet Explorer immer den Überschreibmodus, während alle anderen Browser den Einfügemodus verwendeten. Mit NJX 9.1.2 wurde diese Inkonsistenz behoben. Das Standardverhalten ist nun für alle Browser der Überschreibmodus. Dies entspricht dem Standardverhalten von Natural-Anwendungen.

Wenn Sie den Einfügemodus bevorzugen, gehen Sie wie folgt vor:

1. Kopieren Sie die Datei `WEB-INF/transuni.xml` nach `WEB-INF/usertransuni.xml` – siehe Abschnitt [XSLT anpassen](#) oben.
2. Öffnen Sie `usertransuni.xml` in einem Texteditor. Suchen Sie nach der folgenden Zeile:

```
<xsl:attribute name="onkeypress">handleKeyPress(this.name, this);</xsl:attribute>
```

3. Entfernen Sie diese Zeile und speichern Sie die Datei.

Benutzerspezifisch angepasste Dateien in Natural for Ajax WAR-Dateien packen

Wenn Sie den Deployment Wizard for Web Applications von NaturalONE zum Bereitstellen Ihrer Natural for Ajax `.war`-Datei verwenden, können Sie automatisch eine benutzerdefinierte `usertransuni.xml` in die `.war`-Datei zum Bereitstellen packen:

- Fügen Sie Ihrem NaturalONE-Projekt ein `webconfig`-Verzeichnis hinzu, siehe *NaturalONE > Natural for Ajax > Deploying the Application > Content of the Sample webconfig Directory*.
- Kopieren Sie Ihre `usertransuni.xml` in den Ordner `webconfig/web-inf` Ihres NaturalONE-Projekts.

19

Starten einer Natural-Anwendung mit einer URL

Die in der Konfigurationsdatei für die Sitzung und auf der Anmeldeseite (Logon Page) verfügbaren Verbindungsparameter können auch als URL-Parameter der Anmeldeseiten-URL angegeben werden. Auf diese Weise können Sie die Start-URL einer Natural-Anwendung mit einem Bookmark (Lesezeichen) versehen oder eine Anwendung durch Klicken auf einen Hyperlink in einem Dokument starten.

Die URL-Parameter haben Vorrang vor den Definitionen in der Konfigurationsdatei, mit Ausnahme der in der folgenden Tabelle beschriebenen Fälle.

Die folgenden URL-Parameter sind für die Anmeldeseite verfügbar:

URL-Parameter	Entsprechende Option in der Sitzungskonfiguration
<code>natsession</code>	Session ID
<code>natserver</code>	Host name
<code>natport</code>	Port number
<code>natuser</code>	User name
<code>natprog</code>	Application
<code>natparam</code>	Natural parameters
<code>natparamext</code>	Natural parameters Der URL-Parameter <code>natparamext</code> erweitert eine vorhandene Natural-Parameterdefinition in der Konfigurationsdatei. Die Erweiterung funktioniert wie folgt: Die in der Konfigurationsdatei definierten Natural-Parameter stehen an erster Stelle. Anschließend werden die im URL-Parameter <code>natparamext</code> definierten Natural-Parameter hinzugefügt, getrennt durch ein Leerzeichen. Wenn Sie die Definition in der Konfigurationsdatei aufheben möchten, verwenden Sie stattdessen den URL-Parameter <code>natparam</code> .
<code>nattimeout</code>	Timeout (<i>n</i> seconds)



Wichtig: Alle Parameterwerte müssen URL-codiert sein.

Beispiel: Um das Natural-Programm `dump` zu starten, während Ihr Anwendungsserver auf *myhost:8080* und Ihr Natural Web I/O Interface Server auf *myserver1:4811* läuft, können Sie die folgende URL verwenden:

`http://myhost:8080/natuniweb/natural.jsp?natserver=myserver1&natport=4811&natprog=dump&natuser=myusername`

20 Konfigurieren der containerverwalteten Security

■ Informationen zur containerverwalteten Security	140
■ Name und Speicherort der Konfigurationsdatei für die Security	140
■ Security aktivieren	140
■ Security-Beschränkungen definieren	141
■ Rollen definieren	141
■ Authentifizierungsmethode auswählen	142
■ UserDatabaseRealm konfigurieren	142

Informationen zur containerverwalteten Security

Der Natural Web I/O Interface Client ist eine Java EE-basierte Anwendung. Um die Installation zu vereinfachen, ist der Zugriff auf diese Anwendung standardmäßig nicht gesichert. Möglicherweise möchten Sie jedoch den Zugriff auf bestimmte Teile der Anwendung auf bestimmte Benutzer beschränken. Ein wichtiges Beispiel hierfür ist das Konfigurationstool, mit dem Sie die Natural-Sitzungsdefinitionen und die Protokollierungskonfiguration des Natural Web I/O Interface Client ändern können. Ein weiteres Beispiel ist die Natural-Anmeldeseite.

Dieses Kapitel behandelt nicht alle Aspekte der JAAS-basierten Security. Es enthält jedoch ausreichende Informationen, um die vorkonfigurierten Security-Einstellungen des Natural Web I/O Interface Client zu aktivieren und an Ihre Anforderungen anzupassen.

Name und Speicherort der Konfigurationsdatei für die Security

Die Security wird in der Datei *web.xml* konfiguriert. Diese Datei befindet sich im folgenden Verzeichnis:

```
<tomcat-install-dir>/webapps/natuniweb/WEB-INF
```

Security aktivieren

Bei der Bearbeitung und Änderung der Konfigurationsdatei *web.xml* ist größte Sorgfalt geboten. Nach einer Änderung muss der Anwendungsserver neu gestartet werden.

Bearbeiten Sie die Datei *web.xml* und suchen Sie den Abschnitt, der mit "Uncomment the next lines to add security constraints and roles." kommentiert ist. Entfernen Sie die Kommentare aus diesem Abschnitt, indem Sie die unten fett gedruckten Kommentarzeichen entfernen:

```
<!-- Uncomment the next lines to add security constraints and roles. -->
<!--
<security-constraint>
    <web-resource-collection>
        <web-resource-name>Administration Tools</web-resource-name>
        <url-pattern>/njxruntimeools/*</url-pattern>
    </web-resource-collection>
    ...
<security-role>
    <description>Administrator</description>
    <role-name>nwoadmin</role-name>
</security-role>
-->
```

Security-Beschränkungen definieren

Die standardmäßig definierten Security-Beschränkungen sind nur Beispiele. Ein `<security-constraint>`-Element enthält eine Reihe von `<web-resource-collection>`-Elementen in Kombination mit einem `<auth-constraint>`-Element. Das `<auth-constraint>`-Element enthält einen `<role-name>`. Das gesamte `<security-constraint>`-Element beschreibt, welche Rollen Zugriff auf die angegebenen Ressourcen haben.

Beispiel – Die folgende Definition legt fest, dass nur Benutzer mit der Rolle `nwoadmin` Zugriff auf das Konfigurationstool haben:

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>Administration Tools</web-resource-name>
    <url-pattern>/njxruntime/tools/*</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <role-name>nwoadmin</role-name>
  </auth-constraint>
</security-constraint>
```

Im folgenden Abschnitt erfahren Sie, wo und wie die Rollen definiert werden.

Rollen definieren

Einige Zeilen weiter unten in der Datei *web.xml* befindet sich ein Abschnitt `<security-role>`. Hier werden die Rollen definiert, die in `<security-constraint>`-Elementen verwendet werden können. Bei Bedarf können Sie zusätzliche Rollen definieren. Die Zuordnung von Benutzern zu Rollen erfolgt außerhalb dieser Datei und wird häufig in einer bereits auf Ihrer Site eingerichteten Benutzerverwaltung vorgenommen.

Beispiel:

```
<security-role>
  <description>Administrator</description>
  <role-name>nwoadmin</role-name>
</security-role>
```

Authentifizierungsmethode auswählen

In der Datei *web.xml* gibt es einen Abschnitt `<login-config>`. Das einzige Element, das hier möglicherweise angepasst werden muss, ist `<auth-method>`. Sie können zwischen den Authentifizierungsmethoden `FORM` und `BASIC` wählen. Bei der formularbasierten Authentifizierung wird eine bestimmte Seite angezeigt, auf der sich Benutzer, die auf eine zugangsbeschränkte Ressource zugreifen möchten, authentifizieren können. Bei der Basisauthentifizierung wird der Webbrowser angewiesen, die Anmeldedaten des Benutzers in einem eigenen Dialogfeld abzufragen.

Beispiel:

```
<login-config>
  <auth-method>FORM</auth-method>
  ...
</login-config>
```

UserDatabaseRealm konfigurieren

Geben Sie in der Datei *tomcat-users.xml* (die sich im Verzeichnis *conf* befindet) die Rolle `nwadmin` für einen gewünschten Benutzernamen und ein Passwort an. Beispiel:

```
<user username="pepe" password="pepe123" roles="nwadmin"/>
```

Ausführliche Informationen zur erforderlichen Realm-Konfiguration für Tomcat finden Sie unter <http://tomcat.apache.org/tomcat-6.0-doc/realm-howto.html#UserDatabaseRealm> .

21

Konfigurieren von SSL

■ Informationen zum Konfigurieren von SSL	144
■ Eigene Trust-Datei erstellen	144
■ SSL-Verwendung in der Konfigurationsdatei definieren	145

Informationen zum Konfigurieren von SSL

Trust-Dateien werden für eine sichere Verbindung zwischen dem Natural Web I/O Interface Server und dem Natural Web I/O Interface Client verwendet. Die Serverauthentifizierung kann nicht deaktiviert werden. Eine Trust-Datei ist immer erforderlich.

Eine Trust-Datei enthält die Zertifikate, denen Sie vertrauen. Dies können Zertifikate einer Zertifizierungsstelle (Certificate Authority, CA) wie VeriSign oder selbstsignierte Zertifikate sein.

Informationen zu den erforderlichen Schritten auf dem Natural Web I/O Interface Server und zum Erstellen eines selbstsignierten Zertifikats, das in den Client importiert werden muss, finden Sie unter [SSL-Unterstützung](#).

Um eine sichere Verbindung herzustellen, müssen Sie so wie in den folgenden Abschnitten beschrieben vorgehen.

Eigene Trust-Datei erstellen

Um eine eigene Trust-Datei zu erstellen, können Sie beispielsweise das Dienstprogramm `keytool` im Verzeichnis `bin` der Java-Laufzeitumgebung (JRE) verwenden. Hier sind einige hilfreiche Beispiele:

- Erstellen Sie einen passwortgeschützten Keystore, der einen privaten Schlüssel und ein selbstsigniertes Zertifikat enthält:

```
keytool -genkey -alias foo -keyalg RSA -keystore truststore.jks -keysize 2048
```

Sie werden zur Eingabe eines Passworts aufgefordert. Sie müssen ein Passwort festlegen.

```
keytool -genkey -alias foo -keyalg RSA -keystore truststore.jks -keysize 2048 -storepass your-password
```

Das Passwort wird inline angegeben.

Beide Kommandos

- erstellen ein Schlüsselpaar (privater Schlüssel und selbstsigniertes Zertifikat),
 - speichern es unter dem Alias `foo`,
 - speichern es in einer Datei namens `truststore.jks` und
 - schützen die Datei mit einem Passwort.
- Importieren Sie ein Zertifikat:

```
keytool -import -alias "name-for-ca" -keystore truststore.jks -storepass ←
"your-password" -file server.cert.crt
```



Anmerkung: Sie *müssen* einen Alias angeben und sollten einen aussagekräftigen Namen für den Alias verwenden. Sie können in Ihrer Trust-Datei mehrere Zertifikate hinzufügen, aber jedes muss einen eindeutigen Alias haben.

- Listen Sie die Zertifikate in einer Trust-Datei auf:

```
keytool -list -v -keystore truststore.jks
```

- Löschen Sie ein Zertifikat aus einer Trust-Datei:

```
keytool -delete -alias "name-for-ca" -keystore truststore.jks
```

Wenn Sie die Trust-Datei oder ihr Passwort ändern, müssen Sie den Anwendungsserver neu starten, damit Ihre Änderung wirksam wird.

SSL-Verwendung in der Konfigurationsdatei definieren

Rufen Sie das [Konfigurationstool](#) auf und gehen Sie wie folgt vor:

1. Legen Sie in den globalen Einstellungen für alle definierten Sitzungen den Pfad zur SSL-Trust-Datei (**SSL trust file path**) und, falls erforderlich, das Passwort für die SSL-Trust-Datei (**SSL trust file password**) fest. Siehe auch [Globale Einstellungen](#) in *Natural Client Configuration Tool*.

Bei der Serverauthentifizierung prüft der Natural Web I/O Interface Client, ob das Zertifikat des Natural Web I/O Interface Server bekannt ist. Ist dies nicht der Fall, wird die Verbindung abgelehnt.

Wenn im Konfigurationstool keine Trust-Datei definiert ist, versucht der Natural Web I/O Interface Client, die Datei `calist` aus dem Verzeichnis `lib/security` der Java Runtime Environment (JRE) zu lesen. Das Standardkennwort für diese Datei lautet `changeit`.

2. Definieren Sie eine Session und setzen Sie die Sitzungsoption **Use SSL** auf **Yes**. Siehe auch [Übersicht über die Session-Optionen](#) in *Natural Client Configuration Tool*.

22 Protokollierung (Logging)

■ Informationen zur Protokollierung	148
■ Name und Speicherort der Konfigurationsdatei für die Protokollierung	148
■ Aufrufen der Seite Logging Configuration	148
■ Übersicht über die Optionen für die Ausgabedatei	150

Informationen zur Protokollierung

Der Natural Web I/O Interface Client verwendet die Java Logging API. Bei Problemen mit dem Natural Web I/O Interface Client können Sie die Protokollierung aktivieren und so die Protokollinformationen in eine Ausgabedatei schreiben. Dies sollte nur bei Aufforderung durch den Support erfolgen.

Die Protokollierung können Sie mit dem **Konfigurationstool** konfigurieren.



Anmerkung: Einige Protokollinformationen werden unabhängig von den Einstellungen in der Konfigurationsdatei auch in die Konsole geschrieben. Die Konsole zeigt die Informationen an, die normalerweise von den Protokollierungsstufen SEVERE, WARNING und INFO bereitgestellt werden.

Name und Speicherort der Konfigurationsdatei für die Protokollierung

Der Name der Konfigurationsdatei lautet *natlogger.xml*. Sie befindet sich unter:

`<application-server-install-dir>server/default/deploy/naturalunicode.rar/log`

Aufrufen der Seite Logging Configuration

Der Inhalt der Konfigurationsdatei *natlogger.xml* wird über die Seite **Logging Configuration** (Protokollierungskonfiguration) des **Konfigurationstools** verwaltet.

➤ Um die Seite **Logging Configuration** aufzurufen:

- 1 Wählen Sie im linken Frame den Link **Logging Configuration**.

Die Seite **Logging Configuration** wird im rechten Frame angezeigt. Beispiel:

Logging Configuration

Specify the output log file characteristics.

- "/" : The local pathname separator
- "%t": The system temporary directory
- "%h": The value of the "user.home" system property
- "%g": The generation number to distinguish rotated logs
- "%u": A unique number to resolve conflicts
- "%p": Translates to a single percent sign "%"

File pattern name:

File type:

File size (0=unlimited):

Number of files:

File enabled: ☒ Yes ☐ No

Append mode: ☐ Yes ☒ No

Specify log levels for individual modules. The available settings are:

- SEVERE: Events that interfere with normal program execution
- WARNING: Warnings, including exceptions
- INFO: Messages related to server configuration or server status, excluding errors
- CONFIG: Messages related to server configuration
- FINE: Minimal verbosity
- FINER: Moderate verbosity
- FINEST: Maximum verbosity

Communication:

Resource adapter:

Session beans:

Message beans:

Configuration file:

Logging:

Utilities:

Natural Web I/O Interface pages:

2. Geben Sie die Eigenschaften der Ausgabedatei an, siehe unten im Abschnitt [Übersicht über die Optionen für die Ausgabedatei](#).
3. Legen Sie die Protokollierungsstufen (Log Levels) für einzelne Module fest, indem Sie die Protokollierungsstufe aus der entsprechenden Dropdown-Liste auswählen.

Eine kurze Beschreibung der einzelnen Protokollstufen finden Sie auf der Seite **Logging Configuration**.

4. Wählen die Schaltfläche **Save Configuration**, um die Änderungen in die Konfigurationsdatei zu schreiben.



Vorsicht: Wenn Sie nicht die Schaltfläche **Save Configuration** wählen, sondern sich stattdessen abmelden oder das Konfigurationstool durch Eingabe einer anderen URL verlassen, werden Ihre Änderungen nicht in die Konfigurationsdatei geschrieben.

Übersicht über die Optionen für die Ausgabedatei

Die folgenden Optionen stehen zur Verfügung, um die Eigenschaften der Ausgabedatei festzulegen:

Option	Beschreibung
File pattern name	Dateimuster-Name. Das Muster für die Generierung des Ausgabedateinamens. Standardwert: "%h/nwolog%g.log" . Der Standardwert bedeutet, dass eine Ausgabedatei mit dem Namen <i>nwolog <number>.log</i> im Home-Verzeichnis des Benutzers erstellt wird, der den Anwendungsserver gestartet hat. Ausführliche Informationen zum Festlegen des Musters finden Sie in der Java-API-Dokumentation.
File type	Dateityp. Das Format der Ausgabedatei. Wählen Sie einen der folgenden Einträge aus der Dropdown-Liste aus: ■ Textformat Ausgabe in einfachem Textformat (Standardeinstellung). ■ XML-Format Ausgabe im XML-Format. Die entsprechende Formatierungsklasse wird dann verwendet.
File size	Dateigröße. Die maximale Anzahl an Bytes, die in eine Ausgabedatei geschrieben werden soll. Null (0) bedeutet, dass es keine Begrenzung gibt. Standardeinstellung: 0.
Number of files	Anzahl der Dateien. Die Anzahl der zu verwendenden Ausgabedateien. Dieser Wert muss mindestens 1 betragen. Standardwert: 10.
File enabled	Datei aktiviert Wenn auf Yes (Standardeinstellung) gesetzt, ist der Datei-Handler aktiviert. Wenn auf No gesetzt, ist der Datei-Handler deaktiviert.
Append mode	Anhängen-Modus. Wenn auf Yes gesetzt, werden die Protokollierungsinformationen an die vorhandene Ausgabedatei angehängt. Wenn auf No (Standardeinstellung) gesetzt, werden die Protokollierungsinformationen in eine neue Ausgabedatei geschrieben.

VII

Betrieb und Überwachung des Natural Web I/O Interface

Server

Betrieb des Natural Web I/O Interface Server
Monitor Client NATMOPI
HTML Monitor Client

23

Betrieb des Natural Web I/O Interface Server

■ Starten des Natural Web I/O Interface Server	154
■ Beenden des Natural Web I/O Interface Server	155
■ Überwachen (Monitoring) des Natural Web I/O Interface Server	155
■ Laufzeit-Trace-Funktion	156
■ Trace-Filter	158

Dieses Kapitel beschreibt den Betrieb eines Natural Web I/O Interface Server (Produktcode: NWO). Sofern nicht anders angegeben, gelten die folgenden Informationen für alle Betriebssysteme.

Starten des Natural Web I/O Interface Server

Der Web I/O Interface Server kann als eine Started Task gestartet werden:

```
//NWOSRV  PROC
//KSPSRV   EXEC PGM=NATRNWO,REGION=4000K,TIME=1440,
//  PARM=('POSIX(ON)/NWOSRV1')
//STEPLIB  DD DISP=SHR,DSN=NWOvrs.LOAD
//         DD DISP=SHR,DSN=NATvrs.LOAD
//CMPRINT  DD SYSOUT=X
//STGCONFIG DD DISP=SHR,DSN=NWOvrs.CONFIG(SRV1)
//STGTRACE DD SYSOUT=X
//STGSTDO  DD SYSOUT=X
//STGSTDE  DD SYSOUT=X
```

wobei *vrs* für die entsprechende NWO- bzw. Natural-Produktversion von steht.



Anmerkung: PARM=('POSIX(ON)/NWOSRV1') - POSIX(ON) ist für eine ordnungsgemäße LE370-Initialisierung erforderlich, und NWOSRV1 ist der Name des Servers für die Kommunikation mit dem Monitor Client.

Der Name der Started Task muss unter RACF und den z/OS Linux System Services definiert werden.



Anmerkung: Wenn Sie die Datasets des Natural Web I/O Interface Server anhand der Serverkennung (*server-id*) qualifizieren, ist die Serverkennung auf eine maximale Länge von 6 Zeichen beschränkt.

Alternativ können Sie Natural Web I/O Interface Servers während der **SMARTS**-Initialisierung automatisch starten, indem Sie den SMARTS SYSPARM-Parameter STARTUPPGM verwenden. Geben Sie in der SMARTS SYSPARM-Datei Folgendes an:

```
STARTUPPGM='NATRNWO <server-id>'
```

Beispiel:

```
STARTUPPGM='NATRNWO NWOS1 '
```

Beenden des Natural Web I/O Interface Server

Der Natural Web I/O Interface Server kann über den Monitor Client `NATMOPI` beendet werden, siehe [Monitor-Kommandos](#) weiter unten.

Überwachen (Monitoring) des Natural Web I/O Interface Server

Damit der Administrator den Status des Natural Web I/O Interface Server überwachen kann, steht eine Monitor-Task zur Verfügung, die beim Start des Servers automatisch initialisiert wird. Mit den unten beschriebenen Monitor-Kommandos kann der Administrator die Aktivitäten des Servers steuern und überwachen, bestimmte Benutzersitzungen abbrechen, den gesamten Server beenden usw.

Die folgenden Themen werden behandelt:

- [Monitor-Kommunikation](#)
- [Monitor-Kommandos](#)

Monitor-Kommunikation

Zur Kommunikation mit dem Monitor können Sie den Monitor Client `NATMOPI` verwenden, siehe [Monitor Client NATMOPI](#). Alternativ können Sie den HTML Monitor Client verwenden, der Standard-Webbrowser unterstützt, siehe [HTML Monitor Client](#).

Alternativ können Sie das Operator-Kommando `MODIFY` verwenden, um die im Abschnitt [Monitor-Kommandos](#) beschriebenen Monitor-Kommandos auszuführen. Die Ausgabe des ausgeführten Monitor-Kommandos wird in das Systemprotokoll geschrieben.

Beispiel:

```
F jobname,APPL=ping
```

sendet das Kommando `ping` an den Web I/O Interface Server, der unter dem Job `jobname` ausgeführt wird.

Monitor-Kommandos

Der Natural Web I/O Interface Server unterstützt die folgenden Monitor-Kommandos:

Monitor-Kommando	Aktion
ping	Überprüft, ob der Server aktiv ist. Der Server antwortet und sendet die Zeichenfolge I'm still up
terminate	Beendet den Server.
abort	Beendet den Server sofort, ohne Ressourcen freizugeben.
set <i>configvariable</i> <i>value</i>	Mit dem Kommando set können Sie die Konfigurationseinstellungen des Servers ändern. Beispiel zum Ändern von TRACE_LEVEL: set TRACE_LEVEL 0x00000012
list sessions	Gibt eine Liste der aktiven Natural-Sitzungen auf dem Server zurück. Für jede Sitzung gibt der Server Informationen über den Benutzer, dem die Sitzung gehört, die Initialisierungszeit der Sitzung, die Zeit der letzten Aktivität und eine interne Sitzungskennung (<i>session-id</i>) zurück.
list cputime	Gibt eine Liste der Transaktionen mit der höchsten CPU-Zeitnutzung zurück, sortiert nach CPU-Zeit in absteigender Reihenfolge. Siehe CPU_TIME_HISTORY und CPU_TIME_TRACE, um den CPU-Zeitverlauf zu aktivieren.
cancel session <i>session-id</i>	Bricht eine bestimmte Natural-Sitzung innerhalb des Natural Web I/O Interface Server ab. Um die Sitzungskennung (<i>session-id</i>) zu erhalten, können Sie das Monitor-Kommando list sessions verwenden.
cleanup	Beendet Sitzungen, die länger als in der Konfigurationsparameter SESSION_TIMEOUT angegeben inaktiv sind.
help	Liefert Hilfeinformationen zu den unterstützten Monitor-Kommandos.

Laufzeit-Trace-Funktion

Zu Debugging-Zwecken verfügt der Servercode über eine integrierte Trace-Funktion, die bei Bedarf aktiviert werden kann.

Die folgenden Themen werden behandelt:

- Trace-Medium
- Trace-Konfiguration

■ Trace-Level

Trace-Medium

Der Natural Web I/O Interface Server schreibt seine Laufzeit-Trace in die logische Systemdatei `SYSOUT` der FSIO-Task.

Trace-Konfiguration

Die Trace-Funktion wird durch einen **Trace-Level** konfiguriert, der die Details der Trace definiert. Sobald eine Trace aktiviert ist, kann sie durch Angabe eines **Trace-Filters** auf bestimmte Clients oder Client-Anfragen beschränkt werden, siehe auch Web I/O Interface Server-Konfigurationsparameter `TRACE_FILTER`.

Jede Natural-Sitzung verfügt über ein 32-Bit-Trace-Statuswort (TSW), das den Trace-Level für diese Sitzung definiert. Der Wert des TSW wird im Web I/O Interface Server-Konfigurationsparameter `TRACE_LEVEL` festgelegt. Der Wert Null bedeutet, dass die Trace-Funktion deaktiviert ist.

Trace-Level

Jedes Bit des Trace-Statusworts (TSW) ist für bestimmte Trace-Informationen zuständig. Beginnend mit dem Bit ganz rechts:

Bit 31	Tracing der Hauptereignisse (Serverinitialisierung/-beendigung, Clientanforderung/-ergebnis).
Bit 30	Detaillierte Funktionen (Sitzungszuordnung, Rollin-/Rollout-Aufrufe, Details der Anfrageverarbeitung).
Bit 29	Dump der internen Speicherbereiche.
Bit 28	Zugriff auf das Sitzungsverzeichnis.
Bit 27	Dump des Sende-/Antwortpuffers ausgeben.
Bit 26	Dump des Sende-/Antwortpuffers verkürzt ausgeben. Es werden nur die ersten 64 Bytes ausgegeben.
Bit 25	Dump des E/A-Puffers ausgeben.
Bit 24	Dump des E/A-Puffers verkürzt ausgeben. Es werden nur die ersten 64 Bytes ausgegeben.
Bit 23	Gateway-Ereignis abrufen.
Bit 22-16	Frei.
Bit 15	Tracing nur bei Fehlersituationen.
Bit 14	Anwendung von Trace-Filterdefinitionen.
Bit 13-08	Frei.
Bit 07-01	Frei.
Bit 00	Reserviert für Trace-Level-Erweiterung.

Trace-Filter

Es ist möglich, das Tracing durch einen logischen Filter einzuschränken, um das Volumen der Server-Trace-Ausgabe zu reduzieren.

- Der Filter kann mit dem Konfigurationsparameter `TRACE_FILTER` eingestellt werden.
- Der Filter kann aus mehreren durch Leerzeichen getrennten `keyword=filtervalue`-Zuweisungen bestehen.
- Um die Filterdefinition zu aktivieren, muss das Trace-Bit 14 im Trace-Statuswort (siehe [Trace-Level](#)) gesetzt sein.

Das Filter-Schlüsselwort lautet:

Client	Filtert die Trace-Ausgabe nach bestimmten Clients.
--------	--

Es gelten folgende Regeln:

- Wenn ein Schlüsselwort mehrfach definiert ist, werden die Werte kumuliert.
- Der Wert muss in geschweifte Klammern gesetzt werden und kann eine durch Leerzeichen getrennte Liste von Filterwerten sein.
- Die Werte sind nicht groß-/kleinschreibungsabhängig.
- Die Verwendung von Platzhalterzeichen (*) ist möglich.

Beispiel:

```
TRACE_FILTER="Client=(KSP P*)"
```

Jede Anfrage der Benutzerkennung `KSP` und jede Anfrage der Benutzerkennungen, die mit einem `"P"` beginnen, wird verfolgt.

24

Monitor Client NATMOPI

■ Über den Monitor Client NATMOPI	160
■ Syntax der Kommando-Schnittstelle	160
■ Verfügbare Kommando-Optionen	160
■ Monitor-Kommandos	161
■ Verzeichnis-Kommandos	161
■ Kommando-Beispiele	162

Über den Monitor Client NATMOPI

Der Monitor Client NATMOPI ist eine zeichenbasierte Kommando-Schnittstelle zur Überwachung verschiedener Servertypen, die in einer z/OS Natural-Umgebung bereitgestellt werden. Jeder dieser Server verfügt über eigene Monitoring-Kommandos, die in der entsprechenden Server-Dokumentation beschrieben sind. Darüber hinaus steht eine Reihe von Verzeichnis-Kommandos zur Verfügung, die unabhängig vom Servertyp verwendet werden können. Ein NATMOPI kann zur Überwachung verschiedener Servertypen verwendet werden.

Syntax der Kommando-Schnittstelle

Grundsätzlich besteht die Syntax der Kommando-Schnittstelle aus einer Liste von Optionen, wobei jede Option einen Wert haben kann/muss. Beispiel:

```
-s <server-id> -c help
```

wobei `-s` und `-c` Optionen sind und `<server-id>` und `help` die Optionswerte sind.

Es ist möglich, mehrere Optionen anzugeben, aber jeder Option kann nur ein Wert zugewiesen werden.

Die verfügbaren Kommando-Optionen sind unten aufgeführt.

Verfügbare Kommando-Optionen

Wörter in $\langle \rangle$ sind vom Benutzer angegebene Werte.

Kommando-Option	Aktion
<code>-s <server-id></code>	Geben Sie eine Serverkennung für das Senden eines Monitor-Kommandos an. Wenn die Serverkennung im Serververzeichnis nicht eindeutig ist, fordert NATMOPI den Benutzer auf, einen Server auszuwählen.
<code>-c <monitor command></code>	Geben Sie ein Monitor-Kommando an, das an die mit der Option <code>-s</code> definierte Serverkennung gesendet werden soll.
<code>-d <directory command></code>	Geben Sie ein auszuführendes Verzeichnis-Kommando an.
<code>-a</code>	Unterdrückung der Aufforderung zur Eingabe einer eindeutigen Serverkennung. Verarbeiten aller Server, die für die angegebene Serverkennung gelten.
<code>-h</code>	Anzeigen der NATMOPI-Hilfe.

Monitor-Kommandos

Dies sind Kommandos, die zur Ausführung an einen Server gesendet werden. Die verfügbaren Monitor-Kommandos hängen vom Servertyp ab, jedoch unterstützt jeder Server mindestens die Kommandos `ping`, `terminate` und `help`. Weitere Kommandos finden Sie unter [Betrieb des Web-I/O Interface Server](#), wo die entsprechenden Server-Kommandos beschrieben sind.

Verzeichnis-Kommandos

Verzeichnis-Kommandos werden nicht von einem Server, sondern direkt vom Monitor Client NATMOPI ausgeführt.

Mit den Verzeichnis-Kommandos können Sie die vorhandenen Servereinträge durchsuchen und blockierte Einträge entfernen.

Die folgenden Verzeichnis-Kommandos stehen zur Verfügung. Wörter in `<>` sind vom Benutzer angegebene Werte, Wörter in `[ ]` sind optional.

Verzeichnis-Kommando	Aktion
<code>ls [<server-id>]</code>	Listet alle Server aus dem Server-Verzeichnis auf, die auf die angegebene Serverkennung zutreffen. Die Server-Liste wird in Kurzform angezeigt.
<code>ll [<server-id>]</code>	Wie <code>ls</code> , jedoch enthält die Server-Liste erweiterte Server-Informationen.
<code>rs [<server-id>]</code>	Entfernt Server-Einträge aus dem Server-Verzeichnis. Anmerkung: Wenn Sie den Eintrag eines aktiven Servers entfernen, können Sie diesen Serverprozess nicht mehr überwachen.
<code>cl [<server-id>]</code>	Bereinigt das Server-Verzeichnis. Dieses Kommando pingt den angegebenen Server an. Wenn der Server nicht antwortet, wird sein Eintrag aus dem Verzeichnis entfernt.
<code>ds</code>	Gibt den Inhalt des Server-Verzeichnisses aus.
<code>lm</code>	Listet ausstehende IPC-Nachrichten auf.

Kommando-Beispiele

Beispiel für das Anpingen eines Servers in verschiedenen Umgebungen

Server in z/OS (**SMARTS**/Com-plete), Ausgabe in STDOUT:

- System-Operator:

```
/F complete-jobname,NATMOPI -sServerName -cPING
```

- Com-plete-Online-Kommando

```
NATMOPI -sServerName -cPING
```

Server in z/OS (Started-Task oder Batch-Modus):

- NATMOPI im Batch-Modus ausführen:

```
NATMOPI,PARM=(' -sServerName -cPING')
```

Beispiel-Job:

```
//SAGMOPI JOB SAG,CLASS=K,MSGCLASS=X
//NATEX EXEC PGM=NATMOPI,REGION=3000K,
// PARM=(' -Sname -CPING')
//* PARM=(' -H')
//STEPLIB DD DISP=SHR,DSN=NATURAL.XXXvr.LE.LOAD
// DD DISP=SHR,DSN=CEE.SCEERUN
//SYSOUT DD SYSOUT=X
//SYSPRINT DD SYSOUT=X
//* ↵
```

Dabei steht *XXX* für den Web I/O Interface Server-Produktcode (NWO) und *vr* für die zweistellige Produktversion.

- NATMOPI in TSO (Kommando) ausführen:

```
NATMOPI -sServerName -cPING
```

Die NWO-Library muss in der Steplib von TSO enthalten sein.

Weitere Kommando-Beispiele

- `natmopi -dls`
Listet alle im Verzeichnis registrierten Server in Kurzform auf.
- `natmopi -dcl TST -ls TST`
Bereinigt alle Server mit der Kennung `TST*` (Server anpingen und entfernen, wenn er nicht antwortet) und listet alle Server mit der Kennung `TST*` nach der Bereinigung auf.
- `natmopi -sSRV1 -cping -sSRV2 -sSRV3 -ctermiante`
Kommando `ping` an `SRV1` senden. Kommando `terminate` an `SRV2` und `SRV3` senden.
- `natmopi -ctermiante -sSRV1 -cping -sSRV2 -sSRV3`
Entspricht dem vorherigen Beispiel. Das heißt, NATMOPI sendet das Kommando nach der Option `-s` an den Server. Wenn der Option `-s` keine Option `-c` folgt, wird die erste Option `-c` aus der Kommandozeile verwendet.
- `natmopi -sSRV1 -ctermiante -a`
Kommando `terminate` an `SRV1` senden. Wenn `SRV1` im Server-Verzeichnis mehrdeutig ist, das Kommando an alle `SRV1`-Server senden, ohne zur Auswahl aufzufordern.

25

HTML Monitor Client

■ Über den HTML Monitor Client	166
■ Voraussetzungen für den HTML Monitor-Client	166
■ Server-Liste	166
■ Server-Monitor	168

Über den HTML Monitor Client

Der HTML Monitor Client ist eine Monitor-Schnittstelle, die jeden Webbrowser als Benutzeroberfläche für die Überwachung der verschiedenen Servertypen unterstützt, die in einer Natural for z/OS-Umgebung bereitgestellt werden. Jeder dieser Server verfügt über eigene Monitor-Kommandos, die in der entsprechenden Server-Dokumentation beschrieben sind. Mit dem HTML Monitor Client können Sie alle vorhandenen Server auflisten und einen Server für die Überwachung auswählen.

Voraussetzungen für den HTML Monitor-Client

Um den HTML Monitor Client auszuführen, muss auf jedem Server ein HTTP Monitor Server gehostet werden. Der HTTP Monitor Server ist eine Subtask, die in jedem Web I/O Interface Server-Adressraum ausgeführt werden kann. Er wird mit den NWO-Server-Konfigurationsparametern `HTPMON_PORT` und `HTPMON_ADMIN_PSW` konfiguriert. Auf einen HTTP Monitor Server kann über eine TCP/IP-Portnummer zugegriffen werden. Er kann alle Server überwachen, die auf dem aktuellen Knoten laufen (für **SMARTS**: die innerhalb des aktuellen SMARTS laufen). Es ist zwar nicht erforderlich, aber Sie können mehrere HTTP Monitor Server auf einem Knoten ausführen. Jeder benötigt jedoch eine eigene Portnummer.

Server-Liste

Öffnen Sie Ihren Webbrowser und verbinden Sie sich mit dem HTTP Monitor Server über die folgende URL: `http://nodename:port`, wobei *nodename* der Name des Hosts ist, auf dem der Natural Development Server (NDV) läuft, der den Monitor hostet. Und *port* ist die Portnummer, die der Administrator in der NDV-Konfigurationsdatei als Monitor-Port zugewiesen hat.

Beispiel

Natural Server List 

Server ID	Type	Pid	Started	Config Parameters	Session Parameters	
DAEFNDV4  Offline	NDV	110	2023/07/12 11:24:05	PORT_NUMBER 7201 FRONTEND_OPTIONS 0X01 FRONTEND_NAME NATBAT92 TRACE_LEVEL 31+30+27 HTPMON_PORT 7202	Version: NAT9201 Parameter: ADANAME=ADALNKR DBCLOSE=ON ETID=OFF MAXCL=0 Connection SOC:daef:7201 Security Yes	
DAEFNDV2  Online	NDV	120	2023/07/11 08:25:43	PORT_NUMBER 7318 FRONTEND_OPTIONS 0X1 FRONTEND_NAME NATBAT92 TRACE_LEVEL 31+30 HTPMON_PORT 7316	Version: NAT9201 Parameter: ADANAME=ADALNKR ETID=OFF SORT=(EXT=OFF) ZIIP=OFF IM=F Connection SOC:daef:7318 Security Yes	
DAEFNWQ2  Online	NWO	121	2023/07/09 15:58:07	PORT_NUMBER 7307 FRONTEND_OPTIONS 0X01 FRONTEND_NAME NATBAT92 TRACE_LEVEL 31+30 HTPMON_PORT 7317	Version: NAT9201 Parameter: SORT=(EXT=OFF) ZIIP=OFF Connection SOC:daef:7307 Security No	

Version=92100 BuildDate=23-07-13. Processed 3 server in 4.13 seconds.

Die Server-Liste besteht aus Online- und Offline-Servern. Die Offline-Server stellen potenziell tote Servereinträge dar, die durch Auswahl des Symbols **Remove** aus dem Server-Verzeichnis gelöscht werden können. **Remove** wird nur für Offline-Server angezeigt. „Potentially dead“ bedeutet, dass der HTTP Monitor-Server den Server beim Erstellen der Server-Liste angepingt hat, der Server jedoch nicht innerhalb einer Frist von 10 Sekunden geantwortet hat. Selbst wenn Sie also einen Server-Eintrag finden, der als offline markiert ist, kann dieser Server dennoch aktiv sein, aber nicht auf den Ping reagieren. Durch Auswahl von **Remove** wird ein solcher Server nicht beendet, sondern nur sein Verweis im Monitor-Verzeichnis entfernt. Daher kann er vom Monitor nicht mehr erreicht werden.

Durch Auswahl eines Links der Serverkennung wird ein Fenster zur Überwachung des ausgewählten Servers geöffnet. Durch Klicken mit der linken Maustaste auf den Link wird der Server in einem Popup-Fenster geöffnet. Durch Klicken mit der rechten Maustaste auf den Link wird der Server-Monitor entweder in einem neuen Tab oder in einem neuen Browserfenster geöffnet, je nach Auswahl im Kontextmenü.

Server-Monitor

Monitor server NDVDEV42 67099

Ping
Terminate
Abort
ListSess
CancelSession
Flush

Please press command key

---Set Config Parameter---

Mit den Schaltflächen können Sie die angegebenen Monitor-Kommandos ausführen.

Über das Auswahlfeld können Sie die Konfigurationsparameter des Servers ändern. Wenn Sie einen Parameter zur Änderung auswählen, hat dieser einen vordefinierten Wert. Dieser vordefinierte Wert spiegelt nicht die Einstellung des Servers wider, da es sich um einen Beispielwert handelt.

Wenn Sie **ListSess** wählen, wird im Fenster eine Liste aller Natural-Sitzungen angezeigt, zum Beispiel:

Monitor server

Ping
Terminate
Abort
ListSess
CancelSession
Flush

...Set Config Parameter...

Reply for server pid 67099:

	UserId	SessionId	InitTime	LastActivity	St
1	ASO	BE6FD61E1D5A3582	01 12:18:07	01 12:20:04	I
2	CF	BE6F9F6D6C6E07C2	01 08:13:26	01 08:39:22	I
3	INIT	BE6D8051A1271CC2	27 15:43:36	27 15:43:37	
4	NAU	BE6FCFCC26524E02	01 11:49:50	01 11:50:36	I
5	NAU	BE6FCA37073EA302	01 11:24:52	01 11:45:42	I
6	NAU	BE6FC9FEB43D2842	01 11:23:52	01 11:50:21	I
7	NAU	BE6FC5306C352702	01 11:02:22	01 11:18:45	I
8	NAU	BE6FC51B881CD680	01 11:02:01	01 11:18:44	I
9	NAU	BE6FB5BE5D126740	01 09:53:16	01 10:55:25	I
10	NAU	BE6FB554C21F1F42	01 09:51:26	01 10:55:25	I
11	UF	BE6FBC732C88D202	01 10:23:16	01 13:52:40	I
12	WBE	BE6FF0713CBD5842	01 14:15:53	01 14:23:29	I

Sie können Sitzungen abbrechen, indem Sie die Sitzungskennung in der Spalte **SessionId** auswählen und **CancelSession** wählen.

ListSess zeigt die derzeit verbundenen Sitzungen an. Verbundene Sitzungen sind entweder permanente Verbindungen zu Natural Studio oder temporäre Verbindungen zu NaturalONE. Im Falle von NaturalONE ist eine verbundene Sitzung eine Sitzung, die derzeit eine Natural-Sitzung ausführt (z. B. mit **Run As/Debug As**), auf einen E/A-Bildschirm wartet oder die **Natural Command Console for z/OS** verwendet.

In Verbindung mit Security Caching (siehe Parameter `SECURITY_CACHING` in der Dokumentation des Natural Development Server) stehen die folgenden Schaltflächen zur Verfügung:

Monitor server DAEFNDV2 120



- Ping
- Terminate
- Abort
- ListClients
- CancelClient
- ListSess
- CancelSession
- Configuration
- Flush
- Cleanup
- Display Trace

Reply for server pid 120:

	UserId	Host	LogonTime	LastActivity	St
1	USER64	11.23.45.11	12 14:34:40	12 17:21:09	A
2	USER81	10.45.36.105	12 16:18:35	12 16:53:18	N
3	USR01	10.23.37.168	12 13:20:11	12 13:24:53	A
4	QENSF04	10.23.36.211	12 15:21:36	12 15:35:48	A
5	QENSF03	10.23.36.211	12 12:58:48	12 12:59:27	N

ListClients zeigt die angemeldeten Clients mit den folgenden Informationen an (von links nach rechts):

- Nummer des Client-Eintrags
- UserId – Benutzerkennung des Client
- Host – IP-Adresse des Client
- LogonTime – Zeitpunkt der ersten Anmeldung des Client
- LastActivity – Zeitpunkt des letzten Zugriffs des Client
- St – Status des Clients:

A – Client erfolgreich von RACF validiert

N – Client nicht von RACF validiert

CancelClient löscht einen Client und erzwingt die Eingabe eines neuen Passworts. Um einen Client aus der Liste zu löschen, wählen Sie die dem Client zugewiesene Nummer und dann **CancelClient** oder geben Sie die Nummer des Clients ein, nachdem Sie **CancelClient** ausgewählt haben.

In Verbindung mit der CPU-Zeitverfolgung (siehe Parameter [CPU_TIME_HISTORY](#) und [CPU_TIME_TRACE](#)) ist die folgende Schaltfläche verfügbar:

Monitor server DAEFNDV2 120



- Ping
- Terminate
- Abort
- ListClients
- CancelClient
- ListSess
- CancelSession
- Configuration
- Flush
- Cleanup
- Cpu Time
- Display Trace

Reply for server pid 120:

	Cpu Time	SessionId	Day Time	User	Execute
1	10.834753	DEBE58C9712D7012	04 13:57:16	user82	EXECUTE AAA3
2	0.013164	DEB67DA86AD90C34	04 14:21:34	user02	EXECUTE TEST
3	0.004592	DEBE5E3816036616	04 14:21:34	user82	TRPRO
4	0.002754	DEB67DA685265E06	04 14:56:23	user43	EXECUTE LIST
5	0.001318	DEBE5E3816036616	04 14:21:34	user82	TRPRO
6	0.001295	DEB67DA58D3A4A06	04 12:08:55	user01	EXECUTE PROG2
7	0.001080	DEBE5E3816036616	04 14:21:34	user82	LOGON USER82

CPU Time zeigt Transaktionen mit der höchsten Nutzung der CPU-Zeit an. Die Einträge sind nach CPU-Zeitnutzung in absteigender Reihenfolge sortiert. Ein gespeicherter Eintrag enthält die folgenden Informationen (von links nach rechts):

- Nummer des CPU-Zeiteintrags
- CPU-Zeitwert
- Sitzungskennung
- Tag und Uhrzeit des Eintrags
- Benutzerkennung
- Transaktion

Für eine detaillierte Analyse von problematischen Transaktionen kann der entsprechende Trace verwendet werden.

Display Trace bietet grundlegende Filterfunktionen für die Trace-Ausgabe eines Servers. Die Trace-Ausgabe wird vollständig in eine HTML-Seite geladen. Bei sehr großen Trace-Dateien kann es zu Verzögerungen beim Laden der Dateien kommen. Um den Inhalt der Trace-Anzeige zu aktualisieren, beispielsweise wenn neue Zeilen geschrieben wurden, können Sie **Display Trace** erneut wählen.

Jede Trace-Zeile besteht aus Zahl des Tages, Uhrzeit, Thread-Kennung (ID) und Nachrichtentext. Die Thread-Kennung ist eine aufsteigende Zahl, die mit 0 oder 1 (Haupt-Thread) beginnt. Jeder neue Thread erhält eine neue Kennung. Der Haupt-Thread startet mehrere neue Threads wie

HTTP Monitor, Console Interface oder Main Listener. Bei jeder Anfrage von einem Client (wie NaturalOne) wird ein neuer Thread erstellt. Daher ist es möglich, bestimmte Client-Sitzungen zu filtern.

Sie können auch bekannte Browserfunktionen wie Ausschneiden (Cut) und Einfügen (Paste) oder Suchen (Find) verwenden, um innerhalb eines Trace zu navigieren.

Monitor server DAEFNDV2 120



Ping

Terminate

Abort

ListClients

CancelClient

ListSess

CancelSession

Configuration

Flush

Cleanup

Display Trace

```

Reply for server pid 120:
11 08:25:43 00000001 Natural remote development server initializing, PID=16845078
11 08:25:43 00000001 Server ID = DAEFNDV2
11 08:25:43 00000001 NDW Version=09.02.01 PAL Version=56 BuildDate=23-07-10
11 08:25:43 00000001 at: Thu Jul 11 08:25:43 2023, account: USER01
11 08:25:43 00000001 Allocate LTCB
11 08:25:43 00000001 TCb: 25557000 00000001
11 08:25:43 00000001 LTCB allocated at 24060E70
11 08:25:43 00000001 Master SDE allocated at 2406F688
11 08:25:43 00000001 Configuration:
11 08:25:43 00000001 PORT_NUMBER = 7318
11 08:25:43 00000001 FRONTEND_NAME = NATBAT92
11 08:25:43 00000001 SESSION_PARAMETER = ADANAME=ADALNR ETID=OFF SORT=(EXT=OFF) ZIIP=OFF IM=F
11 08:25:43 00000001 SESSION_PARAMETER_MIXED_CASE = YES
11 08:25:43 00000001 IO_LIMIT = 10000
11 08:25:43 00000001 THREAD_NUMBER = 4
11 08:25:43 00000001 THREAD_SIZE = 4000
11 08:25:43 00000001 FRONTEND_OPTIONS = 0X1
11 08:25:43 00000001 TRACE_LEVEL = 31430
11 08:25:43 00000001 TRANSFER_SIZE = 4000
11 08:25:43 00000001 DNS = YES
11 08:25:43 00000001 KEEP_TCB = YES
11 08:25:43 00000001 SECURITY_CACHING = YES
11 08:25:43 00000001 DNS_CODEPAGE = USER
11 08:25:43 00000001 PARICHECK = NO
11 08:25:43 00000001 HTTPMON_PORT = 7316
11 08:25:43 00000001 Front End "NATBAT92" loaded
11 08:25:43 00000001 Monitor task initialized
11 08:25:43 00000002 Monitor listens with MsgLength 1024
11 08:25:43 00000003 Console interface ready
11 08:25:43 00000001 Start HTTP monitor IPv6, port 7316
11 08:25:44 00000001 FE parameter:USERID STARGATESERVER DED54702TPSYS SERVSTURTPVERS 9.2.1.00HSGCLASSX
11 08:25:44 00000001 Template session initialized
11 08:25:44 00000001 FECh:
00000000 00a600c1 00000000 24d6f750 003e0000 *.w.A.....078....* *.....$.P.>..*
00000010 00000004 24d6f688 24d6f688 00000000 *.....06..00Y....* *.....$.P.>..*
00000020 40404040 40404040 40404040 40404040 * * *.....*
00000030 40404040 40404040 40404040 40404040 * * *.....*
00000040 40404040 40404040 40404040 40404040 * * *.....*
00000050 40404040 40404040 40404040 40404040 * * *.....*
00000060 40404040 40404040 40404040 40404040 * * *.....*

```

Filter

Reset

Hosts

Clients

Errors

Filter

Geben Sie einen Text ein, um den Trace zu filtern. Es werden nur Zeilen angezeigt, die den Filtertext enthalten.

Reset

Der Filterwert wird zurückgesetzt. Der gesamte geladene Trace wird erneut angezeigt.

Hosts

Vordefinierter Filter, der die IP-Adresse und, falls bekannt, den Namen aller Hosts anzeigt, die sich beim Server angemeldet haben. Sie können nun beispielsweise den Fortschritt einer bestimmten Sitzung anzeigen, indem Sie eine Thread-Kennung (ID) markieren und die Schaltfläche **Filter** wählen.

Clients

Vordefinierter Filter, der den Namen der Clients anzeigt, die sich beim Server angemeldet haben. Wie bei der obigen Funktion können Sie nun beispielsweise den Fortschritt einer bestimmten Sitzung anzeigen, indem Sie eine Thread-Kennung markieren und die Schaltfläche **Filter** wählen.

Errors

Vordefinierter Filter, der Fehler anzeigt, die auf dem Server auftreten. Sie können nun beispielsweise den Fortschritt einer bestimmten Sitzung anzeigen, indem Sie eine Thread-Kennung markieren und die Schaltfläche **Filter** wählen.

Beispiel:

Filtern Sie den Trace nach Clients, markieren Sie die Thread-Kennung (ID) einer bestimmten Client-Sitzung und wählen Sie **Filter**. Sie sehen nun alle Trace-Zeilen der betreffenden Client-Sitzung.

