
REFLECTION ZFE

Benutzerhandbuch

February 2017

© 2017 Attachmate Corporation, a Micro Focus company. Alle Rechte vorbehalten.

Die im Lieferumfang dieses Micro Focus-Softwareprodukts enthaltene Dokumentation darf ohne schriftliche Genehmigung von Micro Focus in beliebiger Form und mit beliebigen Hilfsmitteln weder vollständig noch teilweise vervielfältigt, übertragen, kopiert oder in eine andere Sprache übersetzt werden. Die Inhalte dieses Dokuments sind auch dann urheberrechtlich geschützt, wenn dieses Dokument nicht mit Software geliefert wird, für die eine Endbenutzerlizenzvereinbarung besteht.

Die Informationen in diesem Dokument dienen nur zu Informationszwecken, können ohne vorherige Bekanntmachung geändert werden und sind nicht als Verpflichtung seitens Micro Focus anzusehen. Micro Focus übernimmt keinerlei Verantwortung oder Haftung für Fehler oder Ungenauigkeiten im Informationsgehalt dieses Dokuments.

Attachmate, das Attachmate-Logo und Reflection sind eingetragene Marken von Micro Focus in den USA. Alle anderen im vorliegenden Dokument genannten Marken- oder Firmennamen dienen ausschließlich Identifikationszwecken und sind das Eigentum der jeweiligen Inhaber.

<http://www.attachmate.com> (<http://www.attachmate.com>)

Inhalt

Über Reflection ZFE	7
1 Einrichten von Installationsoptionen	9
Anpassen der Werte für Wartezeiten in Sitzungen	9
Einrichten von Terminal ID Manager für Reflection ZFE	9
Einrichten der Nutzungsüberwachung für Reflection ZFE	10
Automatisches Starten und Beenden von Diensten	11
Ändern von Anschlüssen	12
Einrichten von Automated Single Sign-On for Mainframe	13
2 Herstellen von sicheren Verbindungen	15
Sichern des Verbindungspunkts Webbrowser zu Reflection ZFE-Sitzungsserver	16
Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver und Verwaltungskomponente zu MSS	17
Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver zu Host	18
Installieren von Unlimited Strength Policy Jars	18
Konfigurieren des Speicherorts für den Schlüsselspeicher auf dem Reflection ZFE-Server	18
Konfigurieren des Schlüsselspeichers in MSS	19
Konfigurieren einer Reflection ZFE-Terminalsitzung	19
Konfigurieren der X.509-Authentifizierung	21
Konfigurieren von Single Sign-on über IIS	23
Aktivieren der FIPS-Sicherheit	23
So aktivieren Sie den FIPS-Modus:	23
Aktivieren von SSL	24
3 Konfigurieren einer Verbindung	25
Herstellen einer Verbindung zum Host.	25
Allgemeine Verbindungseinstellungen	26
3270- und 5250-Verbindungseinstellungen	27
VT-Verbindungseinstellungen	29
UTS-Verbindungseinstellungen	31
Testen der Kriterien für Terminal ID Manager	32
4 Konfigurieren der Sitzungsoptionen	33
Konfigurieren von Anzeigeeinstellungen	33
Host-Bildschirmfarben anpassen	33
Konfigurieren von Hotspots	34
Konfigurieren der Bildschirmabmessungen für VT-und UTS-Hosts	35
Einrichten von Cursoroptionen	35
Einrichten weiterer Anzeigeeoptionen	36
Zuordnen von Tasten	36
3270-, 5250-, VT- und UTS-Tastaturbelegung	37
Konfigurieren von Makros für Benutzer	45
Übertragen von Dateien vom und zum Host	45

Festlegen von Optionen zum Kopieren und Einfügen	46
Einrichten von Regeln für Benutzereinstellungen	47
Anpassen von Bildschirmen	47
Aktivieren von benutzerdefinierten Steuerelementen	48
Verwenden von serverseitigen Ereignissen	48
5 Verwenden von Sitzungen	51
Verwalten Ihrer Sitzungen	51
Arbeiten mit Kurztasten	51
Kopieren und Einfügen	53
Aufzeichnen von Makros für die Sitzung	53
Abmelden	54
6 Druckvorgang	55
7 Erstellen von Makros	57
Verfahren?	58
Aufzeichnen eines Makros	58
Bearbeiten eines Makros	58
Ausführen eines Makros	59
Anhalten eines Makros	59
Löschen eines Makros	59
Anzeigen eigener Makros	60
Fehlersuche für eigene Makros	61
8 Verwenden der Makro-API	65
Makro-API-Objekte	65
Attribute	66
AttributeSet	68
Farbe	69
ControlKey	70
DataCell	75
Dimension	76
Field	76
FieldList	78
OIA	80
OIAStatus	80
AutoSignon	81
Position	82
PresentationSpace	82
Session	85
SessionType	86
StatusSet	87
User Interface	88
Wait	89
Beispielmakros	91
Grundlegende Hostinteraktion	91
Benutzerinteraktion	93
Durchlaufen von Daten	93
Aufrufen eines Webdienstes	94
Arbeiten mit Datenzellen und Attributen	95
Verwenden von Feldern und Feldlisten	96
Makro für die automatische Anmeldung für Mainframes	97

9 Datei- und Datenübertragung	99
Mainframe-Dateiübertragung	99
Übertragen von Dateien.	99
Fehlerbehebung für Dateiübertragungen.	100
10 Protokollierung	101
11 Fehlerbehebung	103

Über Reflection ZFE

Der Reflection ZFE-Web-Client bietet einen browserbasierten HTML5-Zugriff auf 3270-, 5250-, VT- und UTS-Hostanwendungen. Reflection ZFE erfordert keine Änderungen an Ihren Desktops: Sie müssen weder Software bereitstellen noch Patches anwenden oder Konfigurationen durchführen. Sie können Benutzern plattformunabhängigen Zugriff auf alle Hostanwendungen gewähren.

Der Webclient wird mit vollständiger Sitzungssicherung ausgeführt und verwendet SSL/TLS für die sichere Kommunikation mit Ihren Mainframe-Systemen.

Reflection ZFE enthält die folgenden Komponenten:

- ◆ **Micro Focus Host Access Management and Security Server**

Reflection ZFE verwendet zum Erstellen und Verwalten von Hostsitzungen den Verwaltungsserver Management and Security Server (MSS). Die Verwaltungskonsole ist als zentralisierte Website mit Verwaltungstools und einer eigenen Dokumentation ausgestattet. Über diesen zentralen Zugriffspunkt können Sie Terminalsitzungen konfigurieren, sichern, bereitstellen und überwachen. Verwenden Sie eine kompatible MSS-Version, oder installieren Sie die in der Reflection ZFE-Installation enthaltene Version.

- ◆ **Sitzungsserver**

Der Sitzungsserver ist ein NT-Dienst oder UNIX-Daemon, der das Modul für die Ausführung von Hostsitzungen bereitstellt. Dabei können mehrere Sitzungsserver gleichzeitig verschiedene Sitzungen versorgen, um einen schlanken und zügigen Zugriff auf Ihre Hostdaten zu gewähren.

Reflection ZFE-Komponenten für MSS – Für die Verwendung von Reflection ZFE müssen die Reflection ZFE-Verwaltungskomponenten auf Management and Security Server installiert sein. Bei der Remote-Version von Management and Security Server prüft die Installation, ob die entsprechenden Komponenten installiert wurden. Sollte dies nicht der Fall sein, müssen Sie die Installation der Komponenten selbst vornehmen.

- ◆ **Web-Client**

Der Web-Client ist ein Terminalemulator, auf den über einen Browser zugegriffen werden kann. Nach der Zuweisung können Benutzer ganz einfach über eine beliebige Plattform und von einem beliebigen Standort aus auf autorisierte Sitzungen zugreifen.

- ◆ **Dokumentation**

Für die Verwaltungskonsole von Reflection ZFE-Web-Client und Management and Security Server stehen über die Benutzeroberfläche ausführliche Dokumentationen zur Verfügung. Außerdem sind die Dokumentationen im HTML- oder PDF-Format auf der [Micro Focus-Supportseite](#) erhältlich.

Erste Schritte für Administratoren

Eine Übersicht über unterstützte Plattformen und Systemanforderungen finden Sie im [technischen Hinweis 2837](#) (in englischer Sprache).

- ◆ Wenn Sie nach der Installation von Reflection ZFE Fragen zur Bereitstellung haben, finden Sie dazu Informationen unter [Einrichten der Installationsoptionen](#).

- ♦ Unter [Erste Schritte](#) können Sie außerdem einen Beispiel-Workflow durchlaufen. Diese Anleitung beginnt mit der Administratoranmeldung bei der Verwaltungskonsole von Micro Focus Host Access Management and Security Server und schließt mit der authentifizierten Endbenutzerverbindung mit einer Hostsitzung.
- ♦ Sie erstellen und autorisieren Sitzungen für bestimmte Benutzer mithilfe der Verwaltungskonsole. Dies ist die Verwaltungskomponente für MSS, die über das Reflection ZFE-Startmenü aufgerufen wird.
- ♦ Sie können für den Zugriff auf Reflection ZFE auch einen URL verwenden (z. B. `https://sessionserver:7443/zfe`). Alternativ dazu können Sie über den Sitzungsmanager der Verwaltungskonsole auf den Reflection ZFE-Web-Client zugreifen, sofern Sie über Administratorrechte verfügen.
- ♦ Zunächst wird in der Webanwendung eine Verbindungsseite für die Eingabe hostspezifischer Informationen und weiterer Voreinstellungen für Sie und Ihre Benutzer angezeigt.

Erste Schritte für Endbenutzer

- ♦ Für den Zugriff auf Reflection ZFE-Sitzungen verwenden Sie die von Ihrem Administrator bereitgestellte URL-Adresse. Die Adresse könnte ungefähr so aussehen: `https://sessionserver:7443/zfe`.
- ♦ Eine Kurzanleitung sowie Anweisungen zur Navigation in Reflection ZFE finden Sie unter [Verwenden von Sitzungen](#).
- ♦ Endbenutzermakros werden von Einzelpersonen für die Sitzungen erstellt, zu denen ihnen Zugriff gewährt wurde. Nach der Erstellung der Sitzung erteilt der Administrator die Genehmigung für die Erstellung von Makros.
- ♦ Wenn Sie diese Genehmigung erhalten haben, können Sie Bildschirmfarben anpassen, Hotspots konfigurieren, Tasten zuordnen und auf weitere Benutzervoreinstellungen zugreifen.

1 Einrichten von Installationsoptionen

Mithilfe verschiedener Konfigurationsmöglichkeiten können Sie nach der Installation sicherstellen, dass Reflection ZFE erfolgreich ausgeführt wird.

- ♦ „Anpassen der Werte für Wartezeiten in Sitzungen“, auf Seite 9
- ♦ „Einrichten von Terminal ID Manager für Reflection ZFE“, auf Seite 9
- ♦ „Einrichten der Nutzungsüberwachung für Reflection ZFE“, auf Seite 10
- ♦ „Automatisches Starten und Beenden von Diensten“, auf Seite 11
- ♦ „Ändern von Anschlüssen“, auf Seite 12
- ♦ „Einrichten von Automated Single Sign-On for Mainframe“, auf Seite 13

Verwandte Themen

[Management and Security Server Administratorhandbuch](#)
[Fehlerbehebung für Reflection ZFE-Verbindungen](#)
[Konfigurieren der Sitzungsoptionen](#)

Anpassen der Werte für Wartezeiten in Sitzungen

Die Standard-Wartezeit für eine inaktive Reflection ZFE-Sitzung beträgt 30 Minuten. Das bedeutet, dass eine Sitzung, die nicht abgemeldet wurde und inaktiv ist, nach 30 Minuten beendet wird. Sie können diese Einstellung auf dem Server konfigurieren.

- 1 Öffnen Sie dazu die Datei `<install_location>Micro Focus\ReflectionZFE\sessionserver\webapps\zfe|WEB-INF\web.xml`.

- 2 Passen Sie den Wert für die Wartezeit in Sitzungen an:

```
<session-config> <session-timeout>30</session-timeout> <!--In minutes-Minimum values of 5--> <cookie-config> <max-age>604800</max-age> <!--1 week in seconds--> </cookie-config> </session-config>
```

- 3 Starten Sie den Server neu.

Einrichten von Terminal ID Manager für Reflection ZFE

Management and Security Server stellt das Terminal ID Manager-Add-On zum Zusammenfassen von Terminalkennungen in Pools, zum Verfolgen der Kennungsnutzung und zum Verwalten der Wartezeiten bei Inaktivität für bestimmte Benutzer bereit, wodurch die Terminalkennungsressourcen effizienter genutzt und die Betriebskosten erheblich gesenkt werden können.

Für das Terminal ID Manager-Add-On ist eine separate Lizenz erforderlich.

Bevor Sie mit der Konfiguration von Terminal ID Manager für Reflection ZFE beginnen, überprüfen Sie, ob die Funktion für MSS aktiviert ist. Ausführliche Anweisungen zu diesem Thema finden Sie im [Installationshandbuch für MSS](#).

TIPP: Wenn MSS und Reflection ZFE auf demselben Computer installiert sind und beide den Anschluss 80 verwenden, ist keine weitere Konfiguration erforderlich.

Konfigurieren von Terminal ID Manager für Reflection ZFE

Stellen Sie die richtige Adresse für Terminal ID Manager bereit, um Terminal ID Manager für Reflection ZFE zu konfigurieren.

- 1 Öffnen Sie die Datei `sessionserver/conf/container.properties`.
- 2 Aktualisieren Sie `id.manager.server.url=http://localhost:80/tidm` gemäß der in Management and Security Server konfigurierten Adresse für Terminal ID Manager.
- 3 Starten Sie den Reflection ZFE-Sitzungsserver neu.

Einrichten der Nutzungsüberwachung für Reflection ZFE

Management and Security Server stellt Nutzungsüberwachungsfunktionen für Reflection ZFE-Hostsitzungen bereit.

Bevor Sie mit der Konfiguration der Nutzungsüberwachung für Reflection ZFE beginnen, überprüfen Sie, ob die Funktion für MSS aktiviert ist. Ausführliche Anweisungen zu diesem Thema finden Sie im [Installationshandbuch für MSS](#).

In Reflection ZFE wird die Nutzungsüberwachung für alle durch den Reflection ZFE-Sitzungsserver generierten Emulationssitzungen global eingerichtet. Die Einstellungen werden in der Datei `sessionserver/conf/container.properties` konfiguriert.

Tabelle 1-1 Optionen für die Nutzungsüberwachung

Eigenschaft	Beschreibung
<code>metering.enabled</code>	Aktiviert bzw. deaktiviert die Nutzungsüberwachung mit den Werten "true" und "false". Wenn der Wert nicht auf "true" gesetzt ist, wird die Nutzungsüberwachung deaktiviert.
<code>metering.host.required</code>	Gibt an, ob die Sitzung auch dann eine Verbindung zum Host herstellen kann, wenn der Nutzungsüberwachungsserver nicht erreichbar ist. "True" bedeutet, dass keine Verbindungen zu Sitzungen hergestellt werden können, wenn der Nutzungsüberwachungsserver nicht erreichbar ist. "False" bedeutet, dass auch bei nicht erreichbarem Nutzungsüberwachungsserver Verbindungen zu Sitzungen hergestellt werden können.
<code>metering.server.url</code>	Gibt den Namen bzw. die Adresse des Nutzungsüberwachungsservers, des Anschlusses, des Protokolls und des Webapp-Kontexts an. Die Syntax lautet "host:port protocol context". Sie entspricht der Syntax, die der MSS-Server in der Datei <code>MssData/serverconfig.props</code> für die Registrierung der Nutzungsüberwachungsserver verwendet hat. Dabei muss im Bereich host:port des URL das Zeichen „:“ manuell geschützt werden. Beispiel: <code>test990.attachmate.com\:8080</code> .

```
#Beispielergänzungen für sessionserver/conf/container.properties
metering.enabled=true metering.host.required=false
metering.server.url=10.10.11.55\:80|http|meter
```

HINWEIS: Wenn Sie versuchen, eine Verbindung herzustellen, während alle Lizenzen in Verwendung sind, wird die Sitzung getrennt. Um herauszufinden, ob die Verbindung durch den Host oder den Nutzungsüberwachungsdienst getrennt wurde, schauen Sie in der Datei `Reflection ZFE/sessionserver/logs/server.log` nach.

Automatisches Starten und Beenden von Diensten

Alle Serverkomponenten werden als Dienste installiert und können während der Installation für den Start konfiguriert werden.

Wenn Sie auf Linux-, Solaris- oder AIX-Plattformen arbeiten, befolgen Sie die nachstehenden Schritte, um den Sitzungsserver so einzurichten, dass er beim Systemstart automatisch gestartet wird.

Erstellen Sie mithilfe Ihres Installationsverzeichnisses eine Datei mit der Bezeichnung `zfe` und dem folgenden Inhalt:

```
#!/bin/sh
#
#This script manages the service needed to run the session server
#chkconfig:235 19 08
#description:Manage the Reflection ZFE session server

###BEGIN INIT INFO
# Provides:          zfe
# Required-Start:    $all
# Required-Stop:     $all
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Description:       Start the Reflection ZFE Session Server
### END INIT INFO

INSTALL_DIR=<enter installation directory>
BIN_DIR=$INSTALL_DIR/sessionserver/bin
case "$1" in
start)
echo "Starting Reflection ZFE Session Server"
$BIN_DIR/server start

RETVAL=0
;;
stop)
echo "Stopping Reflection ZFE Session Server"
$BIN_DIR/server stop

RETVAL=0
;;
status) echo "Current Reflection ZFE Session Server status"
$BIN_DIR/server status
```

```

RETVAL=0
;;
restart) echo "Restart Reflection ZFE Session Server"
$BIN_DIR/server restart

RETVAL=0
;;
*)
echo "Usage: $0 (start|stop|status|restart) "

RETVAL=1
;;
esac
exit $RETVAL

```

Wählen Sie anschließend Ihre Plattform aus, und führen Sie die relevanten Schritte durch.

Plattform	Gehen Sie wie folgt vor
Linux	<ol style="list-style-type: none"> 1. Kopieren Sie die Datei in das Verzeichnis <code>/etc/init.d</code>. 2. Legen Sie die Dateiberechtigungen fest. Führen Sie unter Verwendung des Werts 755 den Befehl <code>chmod</code> aus. Beispiel: <code>chmod 755 zfe</code> 3. Führen Sie <code>chkconfig</code> aus, um das Initialisierungsskript hinzuzufügen. Beispiel: <code>/sbin/chkconfig --add zfe</code>
Solaris	<ol style="list-style-type: none"> 1. Kopieren Sie die Datei in das Verzeichnis <code>/etc/init.d</code>. 2. Legen Sie die Dateiberechtigungen fest. Führen Sie unter Verwendung des Werts 755 den Befehl <code>chmod</code> aus. Beispiel: <code>chmod 755</code> 3. Erstellen Sie in <code>/etc/rc.d/rc3.d</code> eine symbolische Verknüpfung. Beispiel: <code>ln -s/etc/init.d/zfe/etc/rc3.d/S99zfe</code>
AIX	<ol style="list-style-type: none"> 1. Kopieren Sie die Datei nach <code>/etc/rc.zfe</code>. Beispiel: <code>cp zfe /etc/rc.zfe</code> 2. Legen Sie die Dateiberechtigungen fest. Führen Sie <code>chmod</code> aus. Beispiel: <code>chmod +x /etc/rc.zfe</code> 3. Fügen Sie am Ende der Datei <code>/etc/rc.tcpip</code> die folgenden Zeilen ein: <code>start/etc/rc.zfe " " "start"</code>

Ändern von Anschlüssen

Die Reflection ZFE-Sitzungsserver- und MSS-Anschlüsse können gemäß Ihren Netzwerkanforderungen angepasst werden. Die von Reflection ZFE verwendeten Standardanschlüsse lauten:

Tabelle 1-2 Reflection ZFE- und MSS-Standardanschlüsse

Sitzungsserver	HTTP – 7070 HTTPS – 7443
Reflection ZFE-Komponenten für MSS	HTTP – 7080 HTTPS – 7444
Management and Security Server	HTTP – 80 HTTPS – 443

So ändern Sie die Standardanschlüsse:

Tabelle 1-3 Ändern von Standardanschlüssen

Komponente	Anweisungen
Reflection ZFE-Sitzungsserver	<p>Die Anschlüsse des Reflection ZFE-Sitzungsservers werden unter <code>sessionserver/conf/container.properties</code> festgelegt und können dort auch angepasst werden.</p> <pre>servletengine.port=7070 servletengine.ssl.port=7443</pre> <p>Zum Deaktivieren des Anschlusses setzen Sie den Wert für den Anschluss auf 0. Einen nicht sicheren SSL-Anschluss können Sie deaktivieren, indem Sie den Wert von 7070 auf 0 ändern.</p>
Management and Security Server	<p>Der von MSS für die Herstellung einer HTTPS-Verbindung verwendete SSL-Anschluss ist standardmäßig auf 443 gesetzt. Starten Sie Management Server, wenn Sie die Anschlussnummer ändern müssen. Dadurch wird die Standarddatei <code>PropertyDS.xml</code> erstellt. Öffnen Sie anschließend im Verzeichnis "MssData" die Datei "PropertyDS.xml". Ändern Sie im nachfolgenden Abschnitt den Wert von 443 in die entsprechende Anschlussnummer, und starten Sie dann Management Server neu.</p> <pre><CORE_PROPERTY NAME="sslport"><STRING>443</STRING></pre>

Einrichten von Automated Single Sign-On for Mainframe

Automated Sign-On for Mainframe ist ein Add-On-Produkt für Management and Security Server, mit dem sich Endbenutzer bei einem Terminalemulationsclient authentifizieren können und automatisch bei einer Hostanwendung im z/OS-Mainframe angemeldet werden.

Ausführliche Informationen zur Konfiguration dieser Funktion finden Sie im [Management and Security Server Administratorhandbuch für Automated Sign-On for Mainframe](#).

- 1 Installieren und konfigurieren Sie das Add-On Automated Sign-On for Mainframe für Management and Security Server. [Hier](#) finden Sie ausführliche Anweisungen.
- 2 Nachdem Sie Management and Security Server eingerichtet haben, öffnen Sie die Verwaltungskonsole, um Sitzungen hinzuzufügen und diesen Sitzungen Benutzer zuzuweisen. Während dieses Vorgangs können Sie die [zusätzlichen Konfigurationsschritte](#) abschließen, die für die Implementierung von Automated Sign-On erforderlich sind.
- 3 Ein Reflection ZFE-Makro sendet den Mainframe-Benutzernamen und das Weiterleitungsticket des Benutzers an die Hostanwendung. Der Benutzer ist dann automatisch angemeldet.
Unterstützung bei der Erstellung des Makros:
 - ♦ Die Makro-API enthält das [AutoSignon](#)-Objekt, das die Methoden bereitstellt, die für die Erstellung einer Reflection ZFE-Anmeldung zur Verwendung mit der Funktion Automated Sign-On for Mainframe erforderlich sind.
 - ♦ Sie können auch das [Makro für die automatische Anmeldung für Mainframes](#)-Beispielmakro referenzieren. Dieses Makro verwendet das AutoSignon-Objekt für die Erstellung eines Makros, das mithilfe der Anmeldeinformationen eines Benutzers ein Weiterleitungsticket vom Digital Certificate Access Server (DCAS) abrufen.

Verwandte Themen

- ◆ [Verwenden der Makro-API](#)
- ◆ [Makro-API-Objekte](#)
- ◆ [Beispielmakros](#)

2 Herstellen von sicheren Verbindungen

Wenn Sie Ihre Legacyhosts für Benutzer auf der anderen Seite der Unternehmens-Firewall öffnen, z. B. für Geschäftspartner, entfernte Benutzer, mobile Vertriebstteams u. a., müssen Sie die Informationen vor Sicherheitsrisiken schützen. Mit Reflection ZFE bieten Sie allen Ihren Benutzern einen sicheren Web-an-Host-Zugang. Reflection ZFE bietet zusammen mit Management and Security Server HTTPS-Verbindungen und eine Vielzahl an Autorisierungs- und Authentifizierungsoptionen.

Eine Standardinstallation von Reflection ZFE enthält drei Hauptverbindungspunkte, die hinsichtlich der Sicherheit berücksichtigt werden müssen:

- ♦ [Webbrowser zu Reflection ZFE-Sitzungsserver](#)
- ♦ [Reflection ZFE-Sitzungsserver zu Reflection Host Access Management and Security Server \(MSS\)](#)
- ♦ [Reflection ZFE-Sitzungsserver zu Ihren Legacy-Hostsystemen](#)

Für die Sicherung dieser Verbindungen sind Anweisungen verfügbar.

Über Java Keytool und Zertifikate

Reflection ZFE und MSS verwenden für die Verwaltung von Schlüsseln und Zertifikaten das Java Schlüssel- und Verwaltungswerkzeug. Mit diesem Werkzeug können Sie öffentliche und private Schlüsselpaare und Zertifikate verwalten. Schlüssel und Zertifikate werden in einem Schlüsselspeicher gespeichert, der standardmäßig als Datei implementiert ist.

Ziel.....	Verwenden Sie.....
Zertifikate auflisten	<code>keytool -list -keystore keystore.jks</code>
Zertifikate löschen	<code>keytool -delete-alias mydomain -keystore keystore.jks</code>
Ein Zertifikat exportieren	<code>keytool -export -alias mydomain -file mydomain.cer -keystore keystore.jks</code>
Ein Zertifikat importieren	<code>keytool -importcert -file <path to certificate> -alias <some-alias> -keystore servletcontainer.jks -storetype jceks -storepass not-secure</code>
Eigenständiges Zertifikat anzeigen	<code>keytool -printcert -v -file mydomain.crt</code>

Weitere Informationen finden Sie in der Dokumentation zum [Java Schlüssel- und Verwaltungstool](#).

Verwandte Themen

- ♦ [Sichern des Verbindungspunkts Webbrowser zu Reflection ZFE-Sitzungsserver](#)
- ♦ [Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver und Verwaltungskomponente zu MSS](#)
- ♦ [Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver zu Host](#)

Sichern des Verbindungspunkts Webbrowser zu Reflection ZFE-Sitzungsserver

HINWEIS: Die hier angegebenen Dateipfade beziehen sich auf eine Standardinstallation. Wenn Sie Reflection ZFE an einem anderen Speicherort installiert haben, passen Sie den Pfad entsprechend an.

Wenn eine HTTPS-Verbindung zum Sitzungsserver hergestellt wurde, authentifiziert sich dieser gegenüber dem Clientbrowser mithilfe eines Serverzertifikats. Der Client vergleicht das Serverzertifikat mit den Einträgen im Speicher für vertrauenswürdige Zertifikate. Wenn das Zertifikat bzw. Stammzertifikat vertrauenswürdig ist, kann der Verbindungsaufbau fortgesetzt werden. Gilt das Zertifikat jedoch als nicht vertrauenswürdig, gibt der Browser eine Warnung aus und fordert Sie auf, die Verbindung zu bestätigen.

Selbstsignierte Zertifikate werden standardmäßig vom Sitzungsserver generiert und dazu verwendet, sich beim Client zu identifizieren. Wenn Sie mit einem HTTPS-URL eine sichere Webbrowser-Verbindung zum Sitzungsserver initiieren, während das selbstsignierte Zertifikat verwendet wird, erhalten Sie vom Browser eine Warnmeldung mit der Information, dass das Zertifikat als nicht vertrauenswürdig gilt. Hier weisen Sie den Browser an, dem Zertifikat zu vertrauen und mit der Herstellung einer sicheren Verbindung fortzufahren.

Wenn Sie den Sitzungsserver so konfigurieren, dass er ein vertrauenswürdiges Zertifikat verwendet, ist es nicht mehr notwendig, eine solche Anweisung an den Browser zu geben. Das erforderliche Zertifikat, das in der Regel von einer allgemein anerkannten Zertifizierungsstelle (Certificate Authority, CA) signiert ist, wird von dem für die ZFE-Installation zuständigen Administrator bereitgestellt.

Die Schlüssel- und Zertifikatskette muss in einem Schlüsselspeicher im JCEKS-Format oder im PKCS12-Format mit hoher Verschlüsselungsebene (PBE-SHA1-3DES) enthalten sein. Die Kennwörter für den Schlüssel und den Schlüsselspeicher müssen identisch sein.

So konfigurieren Sie den Reflection ZFE-Sitzungsserver für die Verwendung des Schlüsselspeichers:

1. Öffnen Sie die Datei `sessionserver/conf/container.properties`, und fügen Sie die folgenden drei Zeilen ein. Verwenden Sie dabei entweder Schrägstriche oder doppelte umgekehrte Schrägstriche:

```
servletengine.ssl.keystore=full path to keystore
servletengine.ssl.keystoretype=format name of keystore, either JCEKS or PKCS12
servletengine.ssl.keystorepassword=password for the keystore file you specified
```

2. Starten Sie den Sitzungsserver neu.

Sie können das Standardverhalten ändern und den Clientbrowser so einstellen, dass er beim Start keine unsichere HTTP-Verbindung zum ZFE-Sitzungsserver herstellt. Gehen Sie folgendermaßen vor:

Öffnen Sie die Datei `sessionserver/conf/container.properties`, setzen Sie die Eigenschaft `servletengine.port` auf 0, und starten Sie den Sitzungsserver neu.

Verwandte Themen

- ♦ [Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver und Verwaltungskomponente zu MSS](#)
- ♦ [Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver zu Host](#)

Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver und Verwaltungskomponente zu MSS

HINWEIS: Die hier angegebenen Dateipfade beziehen sich auf eine Standardinstallation und setzen voraus, dass sich `Java\bin` unter Ihrem Dateipfad befindet. Wenn Sie Reflection ZFE an einem anderen Speicherort installiert haben, passen Sie den Pfad entsprechend an.

Diese Anweisungen gelten sowohl für den Sitzungsserver als auch für die Verwaltungskomponente und erfordern eine Änderung an der Datei `container.properties`, die hier gespeichert ist:

- ♦ `sessionserver/conf/container.properties`
- ♦ `managementserver/conf/container.properties`

Die Datei `<component-path>/container.properties` enthält den URL aus Management and Security Server (MSS), der vom Reflection ZFE-Sitzungsserver und von der Verwaltungskomponente verwendet wird:

```
management.server.url=http://my-company.com:80/mss
```

Sie können während der Installation festlegen, dass ein sicherer Kommunikationskanal zwischen dem Reflection ZFE-Sitzungsserver und MSS konfiguriert werden soll. Das hat zur Folge, dass der Installationsvorgang das MSS-Zertifikat abrufen und den Reflection ZFE-Sitzungsserver konfigurieren. Die Verwaltungskomponente muss manuell konfiguriert werden.

Befolgen Sie die nachstehenden Schritte, um die Konfiguration nach Abschluss der Installation manuell durchzuführen:

1. Ändern Sie die Eigenschaft `management.server.url` in der Datei `<component-path>/conf/container.properties`, um das HTTPS-Protokoll zu verwenden und den richtigen MSS-Anschluss festzulegen.
2. Stellen Sie mithilfe des Browsers eine Verbindung zum HTTPS-URL von Management and Security Server her, und weisen Sie den Browser an, das Zertifikat zu speichern.
3. Importieren Sie unter Verwendung des nachstehenden Befehls das Zertifikat in den entsprechenden Reflection ZFE-Schlüsselspeicher (der Befehl kann abhängig von bestimmten Werten anders lauten) im Verzeichnis `<component-path>/etc`: `keytool -importcert -file <path-to-the-MSS-certificate> -alias <some-alias> -keystore servletcontainer.jks -storetype jceks -storepass not-secure`
4. Starten Sie den entsprechenden Dienst neu.

Diese Anweisungen verwenden **not-secure** als Standardkennwort des Schlüsselspeichers. Sie können das Kennwort des Schlüsselspeichers ändern, indem Sie im Verzeichnis `<component-path>/etc` den folgenden Befehl ausführen:

```
keytool -storepasswd -new new_password -keystore servletcontainer.jks -storetype jceks -storepass not-secure.
```

Verwandte Themen

- ♦ [Sichern des Verbindungspunkts Webbrowser zu Reflection ZFE-Sitzungsserver](#)
- ♦ [Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver zu Host](#)

Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver zu Host

Befolgen Sie die nachstehenden grundlegenden Schritte zum Konfigurieren einer TLS-Verbindung zwischen dem Reflection ZFE-Sitzungsserver und dem Host, der TLS unterstützt:

1. Installieren Sie Unlimited Strength Policy Jars von Oracle.
2. Konfigurieren Sie auf dem Reflection ZFE-Sitzungsserver einen Speicherort für den Schlüsselspeicher. (Optional)
3. Konfigurieren Sie den Schlüsselspeicher mithilfe des MSS-Verwaltungsservers.
4. Konfigurieren Sie eine Reflection ZFE-Terminalsitzung für TLS.

Installieren von Unlimited Strength Policy Jars

Die TLS/SSL-Verschlüsselung zwischen dem Reflection ZFE-Sitzungsserver und dem Hostcomputer erfordert die Unlimited-Strength-Policy-Dateien von Oracle oder IBM. Wenn Sie die Standardinstallation verwendet haben, sind diese Dateien bereits installiert. Bei Bedarf finden Sie die Dateien hier:

- ♦ Oracle Java 8 – <http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html>
- ♦ IBM Java 8 – <https://www14.software.ibm.com/webapp/iwm/web/preLogin.do?source=jcesdk>

Informationen zur Installation der Dateien finden Sie in der Oracle-Readme-Datei, die im Downloadpaket enthalten ist.

Konfigurieren des Speicherorts für den Schlüsselspeicher auf dem Reflection ZFE-Server

Dieser Schritt ist nicht erforderlich, wenn Sie für Reflection ZFE die Standardinstallation verwenden. Einige Leistungsanpassungen erfordern allerdings möglicherweise ein lokales Schlüsselspeicherverzeichnis.

Reflection ZFE erstellt standardmäßig einen MSS-Parameter, der auf das lokale Verzeichnis `keystore` verweist, welches sich unterhalb des aktuellen Arbeitsverzeichnisses des Reflection ZFE-Sitzungsservers befindet (in der Regel `sessionserver/bin/keystore`). Sie können ein alternatives Schlüsselspeicherverzeichnis festlegen, indem Sie die Java-Systemeigenschaft `haapi.ecl.keystore.location=<path_to_dir>` definieren.

Konfigurieren des Schlüsselspeichers in MSS

Damit eine Reflection ZFE-Sitzung den TLS-Host bei einem Verbindungsaufbau als vertrauenswürdig einstuft, muss das öffentliche Zertifikat des Hosts mithilfe von Reflection Management and Security Server (MSS) zu einem vertrauenswürdigen Schlüsselspeicher hinzugefügt werden. Die Reflection ZFE-Sitzung ruft dieses Zertifikat bei der ersten Verbindungsherstellung ab.

So konfigurieren Sie den Schlüsselspeicher mit den öffentlichen Zertifikaten vertrauenswürdiger TLS-Hosts:

1. Stellen Sie eine Verbindung zu dem System her, auf dem MSS installiert ist.
2. Kopieren Sie die Datei mit den öffentlichen Zertifikaten des neuen vertrauenswürdigen Hosts in den Ordner `MssData/certificates`. Bei einer automatisierten Windows-Installation befindet sich diese Datei unter `C:\ProgramData\Micro Focus\mss\MssData`. Bei der Datei kann es sich um ein base64-kodiertes DER-Zertifikat oder um ein binäres Reflection-Verwaltungsserver-Zertifikat handeln.
3. Melden Sie sich bei MSS an, zum Beispiel über `http://mycompany.com/mss/AdminStart.html`.
4. Klicken Sie im unteren rechten Bereich der Verbindungsliste auf **Administrative WebStation**.
5. Klicken Sie in Administrative WebStation auf den Abschnitt „Einrichtung für Sicherheit“, und öffnen Sie anschließend die Registerkarte „Zertifikate“.
6. Klicken Sie auf der Registerkarte „Zertifikate“ auf die Verknüpfung zu **Vom Terminalemulator-Applet als vertrauenswürdig eingestufte Zertifikate anzeigen oder bearbeiten**.
7. Auf der Seite „Zertifikate“ werden die importierten Zertifikate oben angezeigt, während vertrauenswürdige Zertifikate (CA-Zertifikate) im unteren Abschnitt der Seite aufgelistet sind. Um ein Zertifikat für einen neuen vertrauenswürdigen Host zu importieren, klicken Sie auf **Importieren**.
8. Geben Sie auf der Seite „Vertrauenswürdige Zertifikate importieren“ den Namen der Zertifikatdatei ein, die Sie nach `MssData/certificates` kopiert haben, sowie gegebenenfalls das Kennwort der öffentlichen Zertifikatdatei und einen Anzeigenamen zur Identifizierung des Zertifikats auf der MSS-Seite „Zertifikate“.
9. Klicken Sie auf **Senden**.

Wenn das Zertifikat zum vertrauenswürdigen Schlüsselspeicher des MSS-Servers hinzugefügt wurde, kehren Sie zu der Liste der Zertifikate zurück und Sie sollten den neuen Host sehen.

Konfigurieren einer Reflection ZFE-Terminalsitzung

Um eine Verbindung zu dem neuen vertrauenswürdigen Host herzustellen, konfigurieren Sie wie gewohnt eine Reflection ZFE-Sitzung, und legen Sie im Dialogfeld „Einstellungen“ als Sicherheitsprotokoll TLS/SSL fest. Achten Sie darauf, für die Verbindung den richtigen TLS-Anschluss anzugeben.

Verwenden von Secure Shell (SSH)

Secure Shell bietet verschlüsselte Kommunikation zwischen dem Client und einem VT-Host.

MSS enthält eine Liste der bekannten Hosts mit den öffentlichen Schlüsseln der Hosts, mit denen Sie sich über SSH verbinden können. SSH-Verbindungen können nur mit Hosts hergestellt werden, die bereits von einem Administrator als vertrauenswürdig eingestuft wurden.

Bei der ersten Herstellung einer SSH-Verbindung mit einem Host über eine Reflection ZFE-Sitzung wird die Datei mit den bekannten Hosts von Management and Security Server in den Reflection ZFE-Sitzungsserver geladen.

Wenn Sie versuchen, mithilfe von SSH eine Sitzung im Sitzungsmanager zu erstellen oder zu bearbeiten, erhalten Sie eine Benachrichtigung, sofern der Schlüssel nicht als vertrauenswürdig anerkannt wird. Sie werden gefragt, ob Sie den Schlüssel als vertrauenswürdig einstufen und fortfahren möchten.

- ♦ Wenn Sie Ja eingeben, wird der Host als vertrauenswürdig eingestuft und zur Liste mit den bekannten Hosts hinzugefügt, und Sie werden aufgefordert, das SSH-Hostkennwort einzugeben.
- ♦ Wenn Sie nicht Ja eingeben, wird der Host weiterhin als nicht vertrauenswürdig eingestuft, und die Sitzung wird getrennt.

Sie können die SSH-Datei mit den bekannten Hosts auch manuell konfigurieren, indem Sie eine SSH-Verbindung zwischen einer Reflection ZFE-Sitzung und dem Host herstellen und den Schlüsselfingerabdruck des Remotehosts zur Liste mit den bekannten Hosts in MSS hinzufügen.

Konfigurieren der Datei mit den bekannten Hosts für SSH-Verbindungen in MSS

So konfigurieren Sie die Datei mit den bekannten Hosts für SSH-Verbindungen in MSS:

1. Stellen Sie eine Verbindung zu dem System her, auf dem MSS installiert ist, und navigieren Sie zum Zertifikatordner des Servers: `C:\ProgramData\Micro Focus\Mss\MssData\certificates` (Windows) oder `/var/opt/microfocus/mss/Mssdata/certificates` (UNIX).
2. Kopieren Sie die Datei mit den öffentlichen Zertifikaten des neuen SSH-Hosts in den oben angegebenen Ordner `MssData/certificates` (Windows) oder `/etc/ssh/ssh_host_rsa_key.pub` (UNIX). Die einzigen gültigen Arten öffentlicher Schlüssel für MSS `known_hosts`-Einträge sind `ssh-rsa` und `ssh-dss`.

Mögliche Formate für öffentliche Schlüssel des Hosts sind OpenSSH, Base64-encode, DER oder PFX. Die Datei muss diesem Format entsprechen: Hostname, IP-Adresse, Schlüsselart, Schlüssel. Der öffentliche Schlüssel kann beispielsweise so aussehen: `alpsuse132, 10.117.16.232 ssh-rsa AAAAB3NzaC1yc2EAAAADAQAB.....`
3. Melden Sie sich bei MSS an (zum Beispiel über `http://mycompany.com/mss/AdminStart.html`).
4. Klicken Sie im unteren rechten Bereich der Verbindungsliste auf **Administrative WebStation**.
5. Klicken Sie in „Administrative WebStation“ auf „Einrichtung für Sicherheit“, und öffnen Sie die Registerkarte „Secure Shell“.
6. Klicken Sie unter **Liste bekannter Secure Shell-Hosts verwalten** auf die Verknüpfung **Liste der bekannten Secure Shell-Hosts anzeigen oder ändern**. Die Seite **Bekannte Secure Shell-Hosts** wird angezeigt
7. Klicken Sie auf **Importieren**. Die Seite **Bekanntes Host importieren** wird angezeigt.
8. Geben Sie den Namen der Datei mit dem öffentlichen Schlüssel, den Namen des Hosts sowie optional das Kennwort der Datei mit den öffentlichen Schlüsseln und die IP-Adresse des Hosts ein. Der eingegebene Name des Hosts muss mit dem Schlüsselnamen identisch sein. Beispiel: Wenn der Name des Schlüssels `hostname.example.com` lautet, dürfen Sie nicht nur `hostname` eingeben.
9. Klicken Sie auf **Senden**.

Nachdem der öffentliche Schlüssel in die Reflection-Datei mit den bekannten Schlüsseln importiert wurde, kehren Sie zur Seite „Bekannte Secure Shell-Hosts“ zurück, und der neue Host wird in der Liste angezeigt.

- ♦ [Sichern des Verbindungspunkts Webbrowser zu Reflection ZFE-Sitzungsserver](#)
- ♦ [Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver und Verwaltungskomponente zu MSS](#)

Konfigurieren der X.509-Authentifizierung

Mit der X.509-Clientauthentifizierung können sich Clients über Zertifikate anstatt mit Benutzername und Kennwort bei Servern authentifizieren, indem der Standard X.509 PKI (Public Key Infrastructure) verwendet wird.

Voraussetzung

- ♦ Überprüfen Sie mithilfe des unter [Sichern des Verbindungspunkts Reflection ZFE-Sitzungsserver und Verwaltungskomponente zu MSS](#) beschriebenen Verfahrens für die manuelle Konfiguration, ob im Zertifikatspeicher ein vertrauenswürdigen Zertifikat installiert wurde. Das Verfahren kann abhängig vom Betriebssystem und vom Browser variieren.

Grundlegende Schritte:

1. Fügen Sie Reflection ZFE-Zertifikate zum MSS Trusted Subsystem hinzu.
2. Aktivieren Sie das MSS Trusted Subsystem.
3. Installieren Sie das Zertifikat der Zertifizierungsstelle auf MSS und Reflection ZFE.
4. Starten Sie die Server neu.
5. Konfigurieren Sie X.509 in der Verwaltungskonsolle von Management and Security Server.

1. Schritt. Hinzufügen der Reflection ZFE-Zertifikate zum MSS Trusted Subsystem

In diesem Schritt wird sichergestellt, dass MSS den Verbindungen des Reflection ZFE-Sitzungsservers und der Verwaltungskomponenten vertrauen kann.

1. Fügen Sie das Zertifikat des Reflection ZFE-Sitzungsservers zum MSS Trusted Subsystem hinzu:

```
Fügen Sie unter <MSS_install_directory>\server\etc dieses Zertifikat hinzu: keytool -
importcert -file
<RZFE_install_directory>\sessionserver\etc\servletcontainer.cer -alias
zfesessionserver -keystore system.jks -storetype jceks -storepass not-secure
```

2. Fügen Sie das Zertifikat der Verwaltungskomponente zum MSS Trusted Subsystem hinzu:

```
Fügen Sie unter <MSS_install_directory>\server\etc dieses Zertifikat hinzu: keytool -
importcert -file
<RZFE_install_directory>\managementserver\etc\servletcontainer.cer -alias
zfmgtserver -keystore system.jks -storetype jceks -storepass not-secure
```

Schritt 2. Aktivieren des MSS Trusted Subsystems

1. Öffnen Sie die Datei <MSS_install_directory>\server\conf\container.properties.
2. Fügen Sie `servletengine.system.ports= <unused network port>` hinzu, und speichern Sie die Datei.

3. Schritt. Installieren Sie das Zertifikat der Zertifizierungsstelle auf MSS und Reflection ZFE

Ihr Zertifikat der Zertifizierungsstelle kann bereits im vertrauenswürdigen Speicher von MSS enthalten sein. Dies kommt häufig bei bekannten Zertifizierungsstellen vor. In dem Fall können Sie diesen Schritt überspringen.

Überprüfung:

Öffnen Sie Administrative WebStation, klicken Sie auf „Einrichtung für Sicherheit“, und öffnen Sie die Registerkarte „Zertifikate“. Öffnen Sie **Von Management and Security Server als vertrauenswürdig eingestufte Zertifikate anzeigen oder bearbeiten**, um eine Liste der verfügbaren Zertifikate anzuzeigen.

Wenn Ihr Zertifikat nicht aufgeführt ist, installieren Sie Ihre signierende Zertifizierungsstelle auf MSS und auf dem Reflection ZFE-Sitzungsserver.

♦ Installation auf MSS:

1. Kopieren Sie das Zertifikat zum MSS-Server: %PROGRAMDATA%\Micro Focus\MSS\MSSData\certificates.
2. Öffnen Sie Administrative WebStation. Öffnen Sie im Startmenü „Micro Focus Host Access Management and Security Server | Verwaltungsserver | Administrative WebStation“.
3. Klicken Sie unter **Administrative WebStation** im linken Bereich auf **Einrichtung für Sicherheit** und anschließend auf die Registerkarte **Zertifikate**. Klicken Sie im Bereich **Liste der vertrauenswürdigen Zertifikate für Management and Security Server verwalten** auf **Von Management and Security Server als vertrauenswürdig eingestufte Zertifikate anzeigen oder bearbeiten**.
4. Klicken Sie im Bereich „Vertrauenswürdige Zertifikate importieren“ auf **Importieren**. Füllen Sie dann die Felder aus, um auf Ihr Zertifikat zu zeigen, das Kennwort festzulegen und dem Zertifikat einen Anzeigenamen zuzuweisen.
5. Klicken Sie auf **Senden**, und überprüfen Sie, ob das Zertifikat aufgeführt wird.

♦ Installation des Zertifikats auf dem Reflection ZFE-Sitzungsserver und der Verwaltungskomponente:

1. Importieren Sie unter <RZFE_install_directory>\sessionserver\etc dieses Zertifikat:
`keytool -importcert -file <path to certificate> -alias zfesessionserver -keystore servletcontainer.jks -storetype jceks -storepass not-secure`
2. Importieren Sie unter <RZFE_install_directory>\managementserver\etc dieses Zertifikat:
`keytool -importcert -file <path to certificate> -alias zfemgtserver -keystore servletcontainer.jks -storetype jceks -storepass not-secure`

4. Schritt. Starten Sie alle Server neu

Starten Sie alle Server neu, damit die Konfiguration wirksam wird.

5. Schritt. Konfigurieren von X.509 mit LDAP-Failover auf der Verwaltungskonsole von MSS

Nachdem die Zertifikate eingerichtet sind, können Sie X.509 mit LDAP-Failover unter **Verwaltungskonsole von Management and Security Server | Einrichtung der Zugangssteuerung** aktivieren. Beschreibungen der Konfigurationsoptionen finden Sie in der Onlinehilfe zur Verwaltungskonsole.

Konfigurieren von Single Sign-on über IIS

Diese Option verwendet den Microsoft IIS-Webserver. Es ist keine weitere Einrichtung erforderlich, wenn Sie das automatische Installationsprogramm von Management and Security Server verwendet und für den Installationsvorgang die Integration in ISS ausgewählt haben. Weitere Informationen zur Installation von Konfigurationen finden Sie im [Management and Security Server Administratorhandbuch](#).

Aktivieren von Reflection ZFE für die Verwendung mit Single Sign-On über IIS

Um Reflection ZFE für die Verwendung dieser Authentifizierungsmethode zu aktivieren, fügen Sie die folgende Eigenschaft in die Datei `<install_dir>/sessionserver/conf/container.properties` ein:

```
management.server.iis.url=<url>
```

Der Wert dieser Eigenschaft besteht aus der Adresse und dem Anschluss des IIS-Webserver sowie dem Pfad / MSS. Beispiel: `http://server/mss`. Geben Sie den Hostnamen ohne Domännennamen an, damit die Domänenanmeldeinformationen an IIS übergeben werden.

Wenn Sie über IIS eine Proxyverbindung für Reflection ZFE herstellen möchten und IIS Single Sign-On verwenden, müssen Sie in derselben `container.properties`-Datei eine weitere Eigenschaft festlegen:

```
servletengine.iis.url=<url>
```

Der Wert weist die gleiche Form auf wie der oben genannte URL, verwendet jedoch die Reflection ZFE-Adresse. Beispiel: `http://server/zfe`. Für diesen URL ist es nicht erforderlich, eine Kurzform des Hostnamens zu verwenden. Weitere Informationen finden Sie im [technischen Hinweis 2859: Using the IIS Reverse Proxy with Reflection ZFE](#) (in englischer Sprache).

Nachdem Sie diese Konfiguration abgeschlossen haben, wählen Sie unter **Verwaltungskontrolle von Management and Security Server | Einrichtung der Zugangssteuerung** die Authentifizierungsoption aus. Beschreibungen der Konfigurationsoptionen finden Sie in der Onlinehilfe zur Verwaltungskontrolle.

Verwandte Themen

- ♦ [Herstellen von sicheren Verbindungen](#)
- ♦ [Technischer Hinweis 2859: Using the IIS Reverse Proxy with Reflection ZFE](#) (in englischer Sprache)

Aktivieren der FIPS-Sicherheit

Die geprüften Verschlüsselungsmodule des Federal Information Processing Standard (FIPS) 140-2 werden von der US-Regierung als Standard für Sicherheitsbestimmungen verwendet. Reflection ZFE unterstützt diesen Standard. Sie können den FIPS-Modus problemlos aktivieren, indem Sie auf dem Sitzungsserver und in der Verwaltungskomponente eine Eigenschaftseinstellung ändern.

So aktivieren Sie den FIPS-Modus:

Die Reflection ZFE-Verwaltungskomponente und der Sitzungsserver enthalten die Datei `container.conf`:


```
<install_directory>\sessionserver\conf\container.conf  
<install_directory>\managementserver\conf\container.conf
```

HINWEIS: Passen Sie in beiden Speicherorten die Eigenschaft an, und starten Sie die Server neu.

1. Öffnen Sie die Datei `container.conf`.
2. Passen Sie die nachstehende Eigenschaft wie folgt an:
`Dcom.attachmate.integration.container.BSAFE.FIPS.enabled=true.`
3. Starten Sie den Server neu.

Verwandte Themen

- ♦ [Technischer Hinweis 2400, Attachmate Products with FIPS 140-2 Validated Crypto Modules](#) (in englischer Sprache)
- ♦ [Technischer Hinweis 2783, Security Updates and Reflection ZFE](#) (in englischer Sprache)

Aktivieren von SSL

SSL 3.0 ist aufgrund bekannter Sicherheitsrisiken in der JDK 8-Datei „`java.security`“ standardmäßig deaktiviert.

Wenn für Hosts jedoch unbedingt SSL 3.0 erforderlich ist, können Sie das Protokoll mithilfe der nachstehenden Schritte aktivieren:

- 1 Beenden Sie alle Anwendungen und Dienste, die SSL 3.0 verwenden.
- 2 Öffnen Sie die Datei `<install_dir>/jre/lib/security/java.security` in einem Texteditor.
- 3 Entfernen Sie die Zeile `jdk.tls.disabledAlgorithms=SSLv3`, oder kommentieren Sie die Zeile aus.

Verwandte Themen

- ♦ [Konfigurieren einer Verbindung](#)

3 Konfigurieren einer Verbindung

Reflection ZFE unterstützt die Hosts IBM 3270 und 5250 sowie VT- und UTS-Hosttypen.

Ihre Benutzer können über von Ihnen erstellte und konfigurierte Sitzungen auf den Host zugreifen. Sitzungen werden von einem Administrator in der Verwaltungskonsolle von Management and Security Server erstellt. Wenn Sie über die Verwaltungskonsolle eine Sitzung starten, wird in einem separaten Browserfenster der Bereich **Verbindung** des Web-Clients geöffnet.

- ♦ „Herstellen einer Verbindung zum Host“, auf Seite 25
- ♦ „Allgemeine Verbindungseinstellungen“, auf Seite 26
- ♦ „3270- und 5250-Verbindungseinstellungen“, auf Seite 27
- ♦ „VT-Verbindungseinstellungen“, auf Seite 29
- ♦ „UTS-Verbindungseinstellungen“, auf Seite 31
- ♦ „Testen der Kriterien für Terminal ID Manager“, auf Seite 32

Herstellen einer Verbindung zum Host

Im Bereich **Verbindung** des Web-Clients konfigurieren Sie die Verbindungsoptionen des ausgewählten Hosttyps. Die Optionen können abhängig vom Hosttyp voneinander abweichen.

Nachdem Sie eine Sitzung erstellt und in der Verwaltungskonsolle gestartet haben, wird die Benutzeroberfläche des Web-Clients angezeigt. Hier legen Sie neben den Verbindungsoptionen auch Ihre eigenen Voreinstellungen und die der anderen Sitzungsbenutzer fest, um die Kommunikation mit dem Host anzupassen.

HINWEIS: Sie können im folgenden Format eine direkte Verbindung zu einer bestimmten Reflection ZFE-Sitzung erstellen:

```
<rzfe-session-server:port>/?name=<Sitzungsname>
```

Benutzer können auf diese Verbindung zugreifen und direkt eine Reflection ZFE-Websitzung starten. Wenn die angegebene Sitzung beim Öffnen bereits vorhanden ist, wird keine neue Sitzung gestartet.

- 1 Wählen Sie aus der Dropdownliste **Typ** den Hosttyp für Ihre Verbindung aus.
 - 2 Geben Sie an, zu welchem Host eine Verbindung hergestellt werden soll. Sie können den vollständigen Hostnamen oder seine vollständige IP-Adresse angeben.
 - 3 Geben Sie die Nummer des Ports ein, den Sie verwenden möchten.
 - 4 Geben Sie die für die Hostverbindung benötigten Informationen ein.
 - 5 Speichern Sie die Verbindungseinstellungen.
-

Verwandte Themen

- ♦ [VT-Verbindungseinstellungen](#)
- ♦ [UTS-Verbindungseinstellungen](#)
- ♦ [Einrichten von Terminal ID Manager für Reflection ZFE](#)

Allgemeine Verbindungseinstellungen

Diese Optionen gelten für alle unterstützten Hosttypen.

♦ Beim Start verbinden

Sitzungen sind standardmäßig so konfiguriert, dass sie beim Erstellen oder Öffnen einer Sitzung automatisch eine Verbindung zum Host herstellen. Sie können jedoch auch eine Sitzung einrichten, die nicht automatisch eine Verbindung zum Host aufbaut. Wählen Sie **Nein** aus, um eine manuelle Verbindung zum Host herzustellen.

♦ Protokoll

Wählen Sie aus der Dropdownliste das Protokoll aus, das für die Kommunikation mit dem Host verwendet werden soll. Um eine Hostverbindung herzustellen, müssen der Reflection ZFE-Web-Client und der Hostrechner dasselbe Netzwerkprotokoll verwenden. Die verfügbaren Werte hängen von dem Host ab, mit dem Sie eine Verbindung herstellen. Dazu gehören:

Tabelle 3-1 Protokollbeschreibungen

Protokoll	Beschreibung
TN3270	TN3270 ist eine Form des Telnet-Protokolls. Dieses Protokoll definiert eine bestimmte Anzahl von Spezifikationen für die allgemeine Kommunikation zwischen Desktopcomputern und Hostsystemen. Es verwendet TCP/IP als Transportprotokoll zwischen Desktopcomputern und IBM-Mainframes.
TN3270E	TN3270E oder Telnet Erweitert ist für Benutzer von TCP/IP gedacht, die über ein Telnet-Gateway mit RFC 1647-Implementierung eine Verbindung zum IBM-Mainframe herstellen. Mit dem Protokoll TN3270E können Sie den Verbindungsgerätenamen (auch LU-Name genannt) angeben. Ferner verfügen Sie über Standardunterstützung für die Tasten ATTN und SYSREQ sowie die SNA-Antwortbehandlung. Wenn Sie mit Telnet Erweitert eine Verbindung zu einem Gateway aufbauen, das dieses Protokoll nicht unterstützt, wird stattdessen das Standardprotokoll TN3270 verwendet.
TN5250	TN5250 ist eine Form des Telnet-Protokolls. Dieses Protokoll definiert eine bestimmte Anzahl von Spezifikationen für die allgemeine Kommunikation zwischen Desktopcomputern und Hostsystemen. Es verwendet TCP/IP als Transportprotokoll zwischen Desktopcomputern und AS/400-Computern.
Secure Shell (VT)	Das Konfigurieren von SSH-Verbindungen empfiehlt sich zum Gewährleisten einer sicheren, verschlüsselten Kommunikation zwischen Ihrem Computer und einem zuverlässigen VT-Host über ein unsicheres Netzwerk. Mit SSH-Verbindungen wird neben der Authentifizierung von Clientbenutzer und Hostcomputer auch die Verschlüsselung aller Daten sichergestellt.
Telnet (VT)	Für die Bereitstellung sicherer Verbindungen sind Telnet Secure Socket Layer (SSL) und Transport Layer Security (TLS) verfügbar.
INT1 (UTS)	Ermöglicht den Zugriff auf Unisys 1100/1200-Hosts über das TCP/IP-Netzwerkprotokoll.

Verwandte Themen

- ♦ [3270- und 5250-Verbindungseinstellungen](#)
- ♦ [VT-Verbindungseinstellungen](#)
- ♦ [UTS-Verbindungseinstellungen](#)
- ♦ [Einrichten von Terminal ID Manager für Reflection ZFE](#)

3270- und 5250-Verbindungseinstellungen

3270- und 5250-Hoststypen erfordern neben den allgemeinen Konfigurationseinstellungen die nachstehenden spezifischen Einstellungen.

♦ **Gerätename**

Wenn Sie als Protokoll TN3270, TN3270E oder TN5250 ausgewählt haben, geben Sie den Gerätenamen an, der bei der Verbindung der Sitzung zum Host verwendet werden soll. Der Gerätename ist auch unter der Bezeichnung Host-LU oder Pool bekannt. Wenn Sie für die Sitzung keinen Gerätenamen angeben, weist der Host der Sitzung dynamisch einen Namen zu. Wenn ein Gerätename in einem Makro festgelegt ist, wird diese Einstellung überschrieben.

Wenn Sie **Terminal ID Manager** ausgewählt haben, können Sie mit dieser Anwendung zur Laufzeit Kennungen zu Clientanwendungen zuordnen. Mit Terminal ID Manager verwalten Sie in Pools zusammengefasste Kennungen für verschiedene Hoststypen. Eine Kennung besteht aus Verbindungsdaten, die für eine einzelne Hostsitzung eindeutig sind. Um Terminal ID-Manager verwenden zu können, muss ein Terminal ID-Manager-Server konfiguriert sein. Weitere Informationen finden Sie unter [Terminal ID Manager](#) im Management and Security Server Administratorhandbuch.

Wenn Sie Terminal ID Manager verwenden möchten und den Terminal ID Manager-Server konfiguriert haben, können Sie anhand der nachstehenden Optionen die Kriterien zum Abrufen einer Kennung konfigurieren. Eine Kennung wird nur dann zurückgegeben, wenn alle angegebenen Kriterien erfüllt sind.

HINWEIS: Beachten Sie, dass Sie durch Angabe eines Kriteriums festlegen, dass die Kennung nur zugewiesen werden soll, wenn eine Kennung mit dem angegebenen Wert gefunden wurde. Die Kennungsanforderung ist nur erfolgreich, wenn die hier ausgewählte Gruppe von Kriterien genau mit einer Kriteriengruppe übereinstimmt, die für mindestens einen Kennungspool in Terminal ID-Manager festgelegt wurde.

Tabelle 3-2 Kriterien für Terminal ID Manager

Kriterium	Beschreibung
Poolname	Definieren Sie dieses Attribut, und geben Sie den Namen des Pools ein, um die Kennungssuche auf einen Pool einzugrenzen.
Client-IP-Adresse	Die IP-Adresse des Clientrechners wird in die Anforderung einer Kennung mit einbezogen.
Hostadresse	Die Adresse des für die Sitzung konfigurierten Hosts wird in die Anforderung einer Kennung mit einbezogen.

Kriterium	Beschreibung
Hostanschluss	Der Anschluss des für die Sitzung konfigurierten Hosts wird in die Anforderung einer Kennung mit einbezogen.
Name der Sitzung	Wenn Sie diese Option wählen, muss die Kennung für die exklusive Verwendung durch die Sitzung konfiguriert sein.
Sitzungstyp	Der Sitzungstyp (z. B. IBM 3270, IBM 5250 oder UTS) ist immer in Anforderungen für eine Kennung enthalten.
Benutzername	<p>Mit diesem Kriterium können Sie gewährleisten, dass ausschließlich zur exklusiven Verwendung durch bestimmte Benutzer erstellte Kennungen zugewiesen werden. Der Name des aktuellen Benutzers entspricht dem Benutzer, dem die Sitzung zur Laufzeit zugewiesen ist. Der Name muss in einer Kennung gefunden werden, um zugewiesen werden zu können.</p> <p>Für die Konfiguration einer auf Benutzernamen basierenden Sitzung ist ein Standardplatzhalter verfügbar: tidm-setup.</p> <p>Wenn ein Administrator Sitzungen mithilfe von tidm-setup konfiguriert, muss Terminal ID Manager Kennungen für diesen Platzhalter bereitstellen. Sie können den Standardnamen mit einem eigenen Namen überschreiben, indem Sie die Datei <code>Micro Focus/ReflectionZFE/sessionserver/conf/container.properties</code> wie folgt anpassen:</p> <pre>id.manager.user.name=custom-username</pre> <p>Dabei wird „custom-username“ durch den gewünschten Namen ersetzt.</p>
Anwendungsname (UTS)	Der Name der Hostanwendung wird in die Anforderung einer Kennung mit einbezogen.

Um das Verhalten bei der Verbindungsherstellung festzulegen, wenn Terminal ID Manager für die betreffende Sitzung keine Kennung zuordnen kann, verwenden Sie **Bei nicht zugeordneter Kennung**:

- ♦ **Verbindungsversuch fehlschlagen lassen** – Wenn diese Option aktiviert ist, versucht die Sitzung nicht, eine Verbindung herzustellen, wenn eine Kennung nicht zugeordnet ist.
- ♦ **Verbindungsversuch zulassen** – Wenn diese Option aktiviert ist, versucht die Sitzung, eine Verbindung herzustellen, wenn eine Kennung nicht zugeordnet ist. Der Versuch kann jedoch vom Host abgelehnt werden. Bei einigen Hosttypen können Benutzer ohne Kennung eine Verbindung herstellen.

Klicken Sie auf [Test](#), um zu bestätigen, dass Terminal ID Manager mithilfe der ausgewählten Kriterien und Werte eine Kennung bereitstellen kann.

- ♦ **Terminalmodell**

Wählen Sie das Terminalmodell aus, mit dem Sie eine Verbindung herstellen. (nur für 3270-Hosttypen)

- ♦ **Hostzeichensatz**

Wählen Sie den 3270- oder 5250-Hostzeichensatz aus, der verwendet werden soll. Bei dieser Einstellung wird für die Konvertierung von Hostzeichen (EBCDIC) in PC-Zeichen (ANSI) eine Konvertierungstabelle ausgewählt. Diese Einstellung sollte mit dem nationalen Zeichensatz übereinstimmen, der von Ihrem Hostsystem verwendet wird. Falls sie nicht übereinstimmt, könnten einige Zeichen, z. B. Modifikationszeichen (Betonungszeichen), falsch angezeigt werden. Schlagen Sie in Ihrer Hostdokumentation die Definitionen der in den betreffenden Zeichensätzen enthaltenen Zeichen nach. Der Standardwert ist Englisch (US) (037).

- ♦ **TLS/SSL-Sicherheit**

Mit SSL- und TLS-Protokollen können ein Server und ein Client eine sichere, verschlüsselte Verbindung über ein öffentliches Netzwerk aufbauen. Wenn Sie mithilfe von SSL/TLS Verbindungen herstellen, authentifiziert ZFE den Server, bevor eine Sitzung geöffnet wird. Alle Daten, die zwischen ZFE und dem Host übertragen werden, sind mit der ausgewählten Verschlüsselungsstufe verschlüsselt. Folgende Optionen stehen zur Auswahl:

Tabelle 3-3 TLS/SSL-Beschreibungen

Sicherheitsoptionen	Beschreibung
TLS 1.2, 1.1, 1.0 und SSL 3.0	Ermöglicht die Verbindung über TLS 1.2, TLS 1.1, TLS 1.0 oder SSL 3.0, abhängig von den Fähigkeiten des Hosts oder Servers, zu dem Sie eine Verbindung herstellen.
TLS 1.2, 1.1, 1.0	Wählen Sie diesen Wert, um nur TLS und nicht SSL für die Verbindung zu verwenden. Als Teil des TLS-Protokolls gleicht der Client den Server- oder Hostnamen mit dem Namen im Serverzertifikat ab. Aus diesem Grund muss bei TLS-Verbindungen der allgemeine Name im Serverzertifikat mit dem Namen des Hosts oder Proxyserver übereinstimmen.
SSL 3.0	<p>Wählen Sie diesen Wert, um SSL und nicht TLS für die Verbindung zu verwenden. Dies ist nicht empfehlenswert; allerdings unterstützen manche Hosts TLS nicht. Wenn Sie diese Option verwenden möchten, müssen Sie einen weiteren Konfigurationsschritt durchführen:</p> <p>SSL 3.0 ist in der JDK 8-Datei <code>java.security</code> standardmäßig deaktiviert. Wenn SSL 3.0 unbedingt erforderlich ist, können Sie das Protokoll deaktivieren. Dazu entfernen Sie in der Datei <code>java.security</code> den Wert „SSLv3“ aus der Eigenschaft <code>jdk.tls.disabledAlgorithms</code>, oder Sie setzen die Sicherheitseigenschaft auf "true", bevor JSSE initialisiert wurde.</p> <p>Ein Beispiel: Sie müssen die Zeile <code>jdk.tls.disabledAlgorithms=SSLv3</code> in der Datei <code><install_dir>\Micro Focus\ReflectionZFE\jre\jre\lib\security\java.security</code> entfernen oder auskommentieren.</p>

HINWEIS: Informationen zum Hinzufügen vertrauenswürdiger Zertifikate, zu Schlüsselspeichern, zur Verwendung von SSK sowie weitere ausführliche Sicherheitsinformationen finden Sie Im Abschnitt zu [Herstellen von sicheren Verbindungen](#).

VT-Verbindungseinstellungen

VT-Hosts erfordern neben den allgemeinen Verbindungseinstellungen die folgenden zusätzlichen Einstellungen:

Tabelle 3-4 Konfigurationsoptionen für VT-Sitzungen

VT-Einstellungen	Beschreibung
Terminalkennung	Diese Einstellung legt fest, welche Antwort Reflection ZFE an den Host sendet, wenn die primären Geräteattribute angefordert werden. Anhand der Antwort kann der Host erkennen, welche Terminalfunktionen ausgeführt werden können. Die für die jeweilige Terminalkennung von Reflection ZFE gesendete Antwort entspricht genau der Antwort des VT-Terminals, da einige Anwendungen ganz bestimmte Geräteattribute erfordern. Diese Einstellung für die Terminalkennung ist unabhängig von der Option im Feld „Terminaltyp“. Die Optionen lauten: VT220, VT420, VT100, DEC-VT100 und VT52.
Lokales Echo	Automatisch (Standard). Hier wird festgelegt, wie Reflection ZFE auf das Echo eines Telnet-Hosts antworten soll: Das Programm versucht, mit dem Host ein Remoteecho auszuhandeln, führt jedoch die Befehle des Hosts aus. „Ja“ bedeutet, dass Reflection mit dem Host ein lokales Echo aushandelt, aber immer ein Echo sendet. „Nein“ bedeutet, dass Reflection ZFE mit dem Host ein Remoteecho aushandelt, aber kein Echo sendet.
Hostfenstergröße festlegen	Ja (Standard). Wenn dieses Kontrollkästchen aktiviert ist, wird die Anzahl der Spalten und Zeilen bei jeder Änderung an den Host gesendet. Dadurch kann der Telnet-Host auch bei einer Änderung der Fenstergröße den Cursor richtig steuern.
Binär-Modus	Nein (Standard). Telnet gibt einen 7-Bit-Datenpfad zwischen dem Host und dem Terminal vor. Diese Art von Datenpfad ist mit bestimmten nationalen Zeichensätzen nicht kompatibel. Allerdings lassen viele Hosts auch 8-Bit-Zeichen zu, ohne dass dabei das achte Bit den Wert null annehmen muss. In bestimmten Fällen muss der Host jedoch explizit angewiesen werden, im 8-Bit-Modus zu arbeiten. Dies geschieht durch das Aktivieren dieses Kontrollkästchens.
LF nach CR senden	Nein (Standard). Ein „echter“ Telnet-Host erwartet, dass eine vom Terminal gesendete Zeile durch die Steuerzeichenfolge „CrNu“ (Wagenrücklauf/Null) abgeschlossen wird. Bei einigen im Internet verfügbaren Hosts handelt es sich jedoch nicht um echte Telnet-Hosts. Diese Hosts erwarten zur Kennzeichnung von Zeilenenden nach dem CR-Zeichen ein Lf-Zeichen (Line Feed = Zeilenvorschubzeichen). Wenn Sie eine Verbindung zu einem solchen Host herstellen möchten, wählen Sie „Ja“.
Hostzeichensatz	Der Standardwert für den Hostzeichensatz hängt von dem jeweils emulierten Terminaltyp ab. Diese Einstellung zeigt den aktuellen Terminalstatus des VT-Hostzeichensatzes an, der durch den Host geändert werden kann. Die entsprechende, in dem Modell gespeicherte Standardeinstellung lautet „DEC Supplemental“.
Automatische Antwort	Nein (Standard). Diese Einstellung gibt an, ob der Antworttext (der über die Eigenschaft „Answerback“ eingerichtet wird) nach dem Aufbau einer Verbindung automatisch an den Host gesendet werden soll.
Antworttext	Mithilfe dieser Einstellung können Sie im Textfeld einen Antworttext eingeben, wenn der Host eine Antwort auf ein ENQ-Zeichen erwartet.
Anzeigespalten	Gibt an, wie viele Spalten angezeigt werden (Standard sind 80 Spalten)
Bildschirmzeilen	24 Zeilen (Standard) Mit dieser Einstellung wird die Anzahl der angezeigten Zeilen angegeben.

UTS-Verbindungseinstellungen

UTS-Hosts erfordern neben den allgemeinen Verbindungseinstellungen die folgenden zusätzlichen Einstellungen:

Tabelle 3-5 Konfigurationsoptionen für UTS INT1-Sitzungen

UTS INT1-Optionen	Beschreibung
Anwendung	<p>Der Name der Hostanwendung oder des Hostbetriebsmodus, auf die bzw. den zugegriffen wird.</p> <p>Dieses Wort bzw. diese Wortgruppe wird bei der ersten Verbindungsherstellung zum Host vom lokalen Computer an den Host gesendet. Wenn Sie ein Hostterminal verwenden, wäre dies der \$\$OPEN-Name der Anwendung. Der Name der Anwendung entspricht in der Regel dem Namen der Umgebung. Die Namen können allerdings auch voneinander abweichen. Ein Beispiel: Der Name der Umgebung lautet MAPPER, und der Name der Anwendung lautet UDSSRC. Während einer Terminalemulationssitzung geben Sie in der Eingabeaufforderung \$\$OPEN MAPPER ein. Wenn die Verbindung hergestellt wurde, sendet INT1 den Namen UDSSRC an den Host.</p>
TSAP	<p>Der gewünschte Transport Service Access Point (TSAP), bis zu 32 Zeichen (z. B. TIPCSU für TIP-Verbindungen, RSDCSU für Demand-Verbindungen). Ein TSAP ist nur dann erforderlich, wenn Sie im IP-Router-Modus eine Verbindung zu einem Host-LAN-Controller (HLC) oder verteilten Kommunikationsprozessor (DCP, Distributed Communications Processor) herstellen. Wenn Sie nicht sicher sind, welchen Wert Sie verwenden müssen, wenden Sie sich an Ihren Hostadministrator.</p>
Ausgangstransaktion	<p>Das Zeichen oder das Wort bzw. die Wortgruppe, das bzw. die der lokale Computer an den Host sendet, wenn zum ersten Mal eine Verbindung mit dem Host hergestellt wird (bis zu 15 Zeichen). Dieser optionale Parameter wird in der Regel mit TIP verwendet. Sie können beispielsweise ^ eingeben, um MAPPER auszuführen. Dieser Parameter kann außerdem für die Übertragung von Kennworten verwendet werden.</p>
Starttransaktion	<p>Wenn Sie eine Ausgangstransaktion konfigurieren, werden die Daten standardmäßig gesendet, sobald die Verbindung zur Sitzung hergestellt wurde. Sie können selbst entscheiden, wann eine Ausgangstransaktion gesendet wird, indem Sie die Ausgangstransaktion mithilfe einer bestimmten Zeichenfolge auslösen.</p> <p>Wenn Sie beispielsweise erst dann die Daten der Ausgangstransaktion senden möchten, wenn die Anmeldung erfolgreich war, geben Sie eine Zeichenfolge ein, die zum Identifizieren einer erfolgreichen Anmeldung verwendet wird.</p> <p>Sie können diese Option zusammen mit Ausgangstransaktion senden verwenden.</p>
Ausgangstransaktion senden	<p>Sie können festlegen, wann die Ausgangstransaktion gesendet werden soll:</p> <ul style="list-style-type: none">◆ Sofort (Standard).◆ Wenn das Zeichen für den Anfang des Eintrags empfangen wird – Diese Einstellung ist hilfreich, wenn mehrzeilige Transaktionen abgeschlossen sein müssen, bevor die Zeichenfolge gesendet wird.◆ Nach angegebenen Millisekunden

UTS INT1-Optionen	Beschreibung
Terminalkennung	<p>Wählen Sie, ob Sie selbst eine Terminalkennung angeben oder dazu Terminal ID Manager verwenden möchten. Wenn Sie dies selbst übernehmen möchten, geben Sie die Kennung im Feld Terminalkennung angeben ein.</p> <p>Wenn Sie Terminal ID Manager verwenden auswählen, werden Sie aufgefordert, die Attribute der Terminalkennung auszuwählen, die Sie für den Abruf einer Kennung verwenden möchten. Weitere Informationen finden Sie unter Terminal ID Manager-Attribute.</p> <p>Klicken Sie zum Testen der Attribute auf Testen.</p>
Terminalkennung angeben	<p>Die Terminalkennung (in der Regel bis zu acht alphanumerische Zeichen) wird für die diesem Pfad zugeordnete Kommunikationssitzung verwendet. Jede Terminalkennung (auch TID oder PID genannt) muss für den jeweiligen Host eindeutig sein.</p>

Testen der Kriterien für Terminal ID Manager

Terminal ID Manager gibt zur Laufzeit Kennungen an Clientanwendungen aus. Verwenden Sie diese Testoption, um zu bestätigen, dass Terminal ID Manager mithilfe der ausgewählten Kriterien und Werte eine Kennung bereitstellen kann.

Die Kriterien für die aktuelle Sitzung werden im Bereich „Verbindung“ angegeben, nachdem Sie entweder über den Gerätenamen (3270- und 5250-Hosttypen) oder das Feld „Terminalkennung (UTS)“ die Option **Terminal ID Manager verwenden** ausgewählt haben. Standardmäßig werden die ausgewählten Kriterien für die aktuelle Sitzung angezeigt.

Klicken Sie auf **Kennung abrufen**, um zu bestätigen, dass der ID-Manager eine Kennung bereitstellen kann, die mit den konfigurierten und ausgewählten Kriterien und Werten übereinstimmt. Der Test gibt den Namen einer verfügbaren Kennung zurück, die die ausgewählten Attributkriterien erfüllt.

Testen weiterer Kriterien und Werte

In diesem Bereich können Sie Kriterien testen, die sich von denen für die aktuelle Sitzung unterscheiden.

1. Wählen Sie beliebige Einträge aus der Liste „Sitzungstyp“ aus, und geben Sie die zu testenden Kriterien an. Sie können alternative Werte testen, die Sie in einer Terminal ID Manager-Beispielanfrage verwenden möchten.
2. Klicken Sie auf **Kennung abrufen**, um zu bestätigen, dass Terminal ID Manager eine Kennung bereitstellen kann, die mit den ausgewählten Kriterien und Werten übereinstimmt. Der Test gibt den Namen einer verfügbaren Kennung zurück, die die ausgewählten Werte erfüllt.

Verwandte Themen

- ◆ [Konfigurieren der Sitzungsoptionen](#)
- ◆ [Kriterien für Terminal ID Manager](#)

4 Konfigurieren der Sitzungsoptionen

Neben den Verbindungseinstellungen können Sie auch andere Anzeigeeinstellungen personalisieren, Bildschirme anpassen, Tasten belegen und verschiedene Funktionen zum Kopieren und Einfügen festlegen.

- ♦ „Konfigurieren von Anzeigeeinstellungen“, auf Seite 33
- ♦ „Zuordnen von Tasten“, auf Seite 36
- ♦ „Konfigurieren von Makros für Benutzer“, auf Seite 45
- ♦ „Übertragen von Dateien vom und zum Host“, auf Seite 45
- ♦ „Festlegen von Optionen zum Kopieren und Einfügen“, auf Seite 46
- ♦ „Einrichten von Regeln für Benutzereinstellungen“, auf Seite 47
- ♦ „Anpassen von Bildschirmen“, auf Seite 47

Konfigurieren von Anzeigeeinstellungen

Sie können Vorder- und Hintergrundfarben für den Host anpassen und Hotspots einrichten, um die Navigation in der Hostanwendung zu erleichtern. Diese Einstellungen gelten für die von Ihnen konfigurierte Sitzung.

Verfahren?

- ♦ [Host-Bildschirmfarben anpassen](#)
- ♦ [Einrichten von Cursoroptionen](#)
- ♦ [Einrichten weiterer Anzeigeeinstellungen](#)
- ♦ [Konfigurieren der Bildschirmabmessungen für VT- und UTS-Hosts](#)
- ♦ [Konfigurieren von Hotspots](#)
- ♦ [Zuordnen von Tasten](#)
- ♦ [Festlegen von Optionen zum Kopieren und Einfügen](#)

Host-Bildschirmfarben anpassen

Sie können die Farbe Ihres Bildschirms und das Aussehen der verschiedenen Hostattribute im Terminalfenster anpassen. Sie können für jedes Element die Vorder- und Hintergrundfarben für 3270-, 5250-, VT- und UTS-Hostverbindungen auswählen. Farben werden mithilfe der Farbpalette oder durch Eingabe des Hex-Code-Formats angegeben.

Die verfügbaren Hex-Farben werden auf zahlreichen Websites angeboten. Ein Beispiel dazu finden Sie unter [w3schools.com HTML Color Picker](http://w3schools.com/html/color-picker)

Abhängig vom Typ der Hostverbindung können Sie verschiedene Optionen sehen.

Verfügbare Optionen für UTS-Hosts:

- ♦ **Farbinformationen vom Host verwenden** – Deaktivieren Sie diese Option, um anstelle der durch den Host angegebenen Farben die hier angegebenen Farben zu verwenden.

- ♦ **Blinken Ein** – Wenn Sie das Blinken ausschalten möchten, deaktivieren Sie diese Option.
- ♦ **Attribut zur Bearbeitung auswählen** – In UTS-Emulationen werden Farben direkt vom Host eingestellt. Sie können für Text mit spezifischen Bildschirmanzeigeeoptionen unterschiedliche Farben angeben. Dazu sind die folgenden Kombinationen verfügbar:

Einfach, Unterstrichen (UND), Durchgestrichen (STK), Linkes Spaltentrennzeichen (LCS), Steuerungsseite und Statuszeile (OIA).

- ♦ **Videointensität**

Die Videointensitäten Blinken, Schwach, Geschützt und Invertiert werden für die Erstellung zusätzlicher Kombinationen mit den Attributen kombiniert. Sie können beispielsweise alle Vordergrund- und Hintergrundfarben mit Schwach + Blinken + Unterstrichen oder Invertiert + Geschützt + Durchgestrichen + Unterstrichen verknüpfen.

Wenn Sie eine Videointensität (oder eine Kombination verschiedener Intensitäten) auswählen, werden diese Intensitäten mit dem Wert aus der Dropdownliste „Attribut“ kombiniert und bilden so eine einzelne Farbzuoordnung.

Verfügbare Optionen für VT-Hosts

- ♦ **Blinken Ein** – Wenn Sie das Blinken ausschalten möchten, deaktivieren Sie diese Option.
- ♦ **Fettschrift Ein** – Zeigt Textabschnitte mit Fettattributen im Terminalfenster als fett formatierten Text an. Wenn fett formatierte Zeichen als Normaltext angezeigt werden sollen, deaktivieren Sie diese Option.
- ♦ **Unterstreichen ein** – Zeigt Text mit Unterstrich an.

So passen Sie für alle Hosttypen die Farben an:

- 1 Klicken Sie im linken Navigationsbereich auf **Anzeige**.
- 2 Klicken Sie unter **Farbzuoordnung** auf das Feld „Hintergrundfarbe“, um die Farbpalette zu öffnen. Wählen Sie aus der Farbpalette die gewünschte Hintergrundfarbe für den Host aus. Alternativ dazu geben Sie die Nummer der gewünschten Hex-Farbe ein.
- 3 Wählen Sie aus der Dropdownliste die Standard-Hostfarbe aus, die Sie ändern möchten.
- 4 Öffnen Sie die Farbpalette für den **Vordergrund**, um eine Farbe auszuwählen, die dem Elementtext zugewiesen werden soll. Sie können den gewünschten Hex-Code auch manuell eingeben. Wählen Sie **Hintergrund** aus, um die neue Farbe dem Feld „Hintergrund“ zuzuweisen.
- 5 Klicken Sie auf **Speichern**, um die Registerkarte „Anzeige“ zu schließen und mit der Konfiguration der Hostverbindung fortzufahren.

Über **Standardwerte wiederherstellen** löschen Sie alle vorgenommenen Änderungen und setzen die Werte für die Farben auf die standardmäßigen Hosteinstellungen zurück.

Konfigurieren von Hotspots

Hotspots sind Schaltflächen, die in Terminalsitzungen über häufig verwendeten Hostbefehlen eingeblendet werden. Wenn Sie Hotspots verwenden, können Sie die Terminalsitzung statt mit der Tastatur per Maus oder Fingerdruck steuern. Der Hotspot überträgt eine Terminaltaste oder einen Befehl zum Host. Hotspots sind standardmäßig für die Verwendung von 3270-, 5250- und VT-Befehlen konfiguriert.

Sie sind standardmäßig aktiviert und werden angezeigt, können jedoch für bestimmte Sitzungen deaktiviert oder ausgeblendet werden.

- ♦ **Hotspots aktivieren**

Wählen Sie **Nein** aus, um Hotspots in einer Sitzung zu deaktivieren.

♦ **Hotspots anzeigen**

Wählen Sie **Nein** aus, um Hotspots auf dem Bildschirm auszublenden. Die Hotspots sind weiterhin funktionsfähig.

Tabelle 4-1 Hotspots für 3270-Hosts

Hotspot	Beschreibung
PF1...PF24	Sendet die Tastenwerte PF1...PF24 zum Host
PA1, PA2 oder PA3	Sendet die Tastenwerte PA1, PA2 oder PA3 zum Host
Eingabe	Sendet die Eingabetaste an den Host
Mehr	Sendet die Löschtaste an den Host

Tabelle 4-2 Hotspots für 5250-Hosts

Hotspot	Beschreibung
Eingabe	Sendet die Eingabetaste an den Host
Mehr...	Sendet die Nach-oben-blättern-Taste an den Host (blättert eine Seite nach oben)
PF1 – PF24	Sendet die Tastenwerte PF1...PF24 zum Host

Tabelle 4-3 Hotspots für 3270-Hosts

Hotspot	Beschreibung
F1 – F20	Sendet die Tastenwerte F1...F20 zum Host

Konfigurieren der Bildschirmabmessungen für VT-und UTS-Hosts

Als Administrator können Sie die Anzahl der Spalten und Zeilen für VT- und UTS-Sitzungen auswählen.

- 1 Öffnen Sie die Registerkarte „Anzeige“.
- 2 Geben Sie unter **Abmessungen** die Anzahl der Spalten und Zeilen an, die in den Bildschirmen enthalten sein sollen. Der Standardwert ist 80 Spalten mal 24 Zeilen.
- 3 Klicken Sie auf **Speichern**.

Einrichten von Cursoroptionen

Mit den Optionen unter „Cursor“ konfigurieren Sie die Darstellung und das Verhalten von Cursor und Lineal.

Option	Funktion....
Cursortyp	<ul style="list-style-type: none"> ◆ Unterstrich zeigt den Cursor als Unterstrich an. ◆ Vertikaler Balken zeigt den Cursor als vertikalen Balken an. ◆ Block (Voreinstellung) zeigt den Cursor als Block in invertierter Darstellung an.
Linealtyp	<ul style="list-style-type: none"> ◆ Vertikal zeigt ein vertikales Lineal an der Cursorposition an. ◆ Horizontal zeigt ein horizontales Lineal an der Cursorposition an. ◆ Fadenkreuz zeigt ein horizontales und ein vertikales Lineal an der Cursorposition an.
Cursorfarbe	Klicken Sie auf das Feld „Farbe“, um die Farbpalette zu öffnen. Wählen Sie aus der Farbpalette die gewünschte Farbe des Cursors und des Lineals aus. Alternativ dazu geben Sie die Nummer der gewünschten Hex-Farbe ein.
Cursorblinken	In der Standardeinstellung blinkt der Cursor (als Block oder Unterstrich dargestellt). Deaktivieren Sie diese Option, um den Cursor ohne Blinken anzuzeigen.

Einrichten weiterer Anzeigoptionen


Einige Anzeigoptionen sind hostspezifisch.

Option	Funktion....
Pixelverhältnis beibehalten	Wählen Sie diese Option aus, um die Proportionen des Hostbildschirms unabhängig von der Größe des Browserfensters beizubehalten. Die Proportionen beschreiben das proportionale Verhältnis zwischen der Breite und der Höhe eines Bildes.
OIA anzeigen	Wählen Sie diese Option aus, um Vorgangs- und Statusmeldungen im Operatorinformationsfeld (OIA, Operator Information Area) im unteren Terminalfensterbereich anzuzeigen. Die OIA-Anzeige ist standardmäßig aktiviert.
Mausklick auf Fenster ignorieren aktivieren	Wenn das Terminalfenster durch einen Mausclick aktiviert wird, geben Sie mit dieser Option an, ob Aktionen wie das Aktualisieren der Terminal-Cursorposition, das Aufheben der Auswahl oder das Ausführen eines Hotspots ebenfalls durchgeführt werden. Diese Aktionen werden standardmäßig nicht durchgeführt.
Invertierte Darstellung (VT)	Mit dieser Option werden die Vordergrund- und Hintergrundfarben umgekehrt, wenn der VT-Host eine invertierte Video-Escape-Sequenz sendet. Wenn diese Option deaktiviert ist, werden die vom Host gesendeten invertierten Videosequenzen ignoriert.
Automatischer Umbruch (VT)	Wenn dieses Kontrollkästchen aktiviert ist, werden Zeichen am rechten Rand automatisch umbrochen und auf der nächsten Zeile fortgesetzt. Wenn das Kontrollkästchen deaktiviert ist, werden Zeichen beim Erreichen des rechten Rands der Anzeige nicht umbrochen. Das Zeichen am rechten Rand wird durch neue Zeichen überschrieben, bis ein Wagenrücklaufzeichen eingegeben wird.

Zuordnen von Tasten

Sie können Tastenkombinationen definieren, über die während einer Sitzung jede zuweisbare Aktion ausgeführt werden kann. Über die Option „Tastaturbelegung anzeigen“ können Sie die Standardtastaturbelegung für jeden Hosttyp und die zugeordneten benutzerdefinierten Tasten für die jeweilige Sitzung anzeigen.

TIPP: Browser verwenden Tastenkombinationen, um Zeitaufwand und Mausclicks zu reduzieren. Dies sollte bei der Zuordnung von Tastenkombinationen berücksichtigt werden. So wird beispielsweise über die Tastenkombination Strg+F1 die Internet Explorer-Hilfe und die Seite für die UTS-Steuerung geöffnet. Unter [Handy Keyboard Shortcuts](#) (in englischer Sprache) erhalten Sie einen Überblick über die in den verschiedenen Browsern verwendeten Tastenkombinationen.

- 1 Öffnen Sie die Registerkarte **Tastenbelegungen verwalten**.
- 2 Klicken Sie unter „Tasten“ auf die entsprechenden Werte, um die Taste oder Tastenkombination hinzuzufügen, die Sie zum Auslösen der gewünschten Aktion im Feld „Tasten“ verwenden möchten.
- 3 Wählen Sie aus dem Dropdownfeld „Belegung“ die Aktion aus, die Sie der Tastenauswahl zuordnen möchten.
- 4 Klicken Sie auf das blaue Häkchen, um die Zuordnung abzuschließen und die Tastaturbelegung zur Sitzung hinzuzufügen.
- 5 Klicken Sie auf , um den Bereich „Tastaturbelegung“ zu schließen und mit der Konfiguration der Hostverbindung fortzufahren. Klicken Sie auf **Verbinden**, um die Konfiguration zu speichern.

Nachdem Sie die Tastaturbelegungen für die Sitzung eingerichtet haben, klicken Sie im linken Bereich auf „Tastaturbelegung anzeigen“, um auf die aktualisierte Belegung zuzugreifen, die auch die benutzerdefinierten Tastaturbelegungen für Fettschrift enthält. Klicken Sie auf die Spaltenüberschrift, um die Liste nach „Taste“ oder „Belegung“ zu sortieren.

Auch für Kopier-/Einfügevorgänge werden verschiedene Tastenkombinationen verwendet. So wird beispielsweise auf einem VT-Hostbildschirm über die Tastenkombination **Strg+Umschalt+A** die Aktion „Alles auswählen“ ausgelöst. Unter [Festlegen von Optionen zum Kopieren und Einfügen](#) finden Sie eine Liste mit Tastaturbelegungen für Kopier-/ und Einfügevorgänge.

Verwandte Themen

[IBM 3270-Tastaturbelegungen](#)

[IBM 5250-Tastaturbelegungen](#)

[VT-Tastaturbelegungen](#)

[UTS-Tastaturbelegungen](#)

[Arbeiten mit Kurztasten](#)

3270-, 5250-, VT- und UTS-Tastaturbelegung

Die folgenden Tabellen zeigen die Standardtasten, Tastennamen und die Beschreibungen für die verschiedenen Host-Tastaturbelegungen.

Tabelle 4-4 IBM 3270-Tastaturbelegung

Taste	Zuordnungen zu	Beschreibung
Strg+F1	Abruf	Sendet die Abruftaste an den Host
Umschalt+Tabulat or	Rücktabulator	Bewegt den Cursor in das vorherige ungeschützte Feld

Taste	Zuordnungen zu	Beschreibung
Strg+F2	Löschen	Entfernt den Bildschirminhalt und sendet die Löschtaste an den Host
Alt+Pfeil nach links	Cursor nach links – doppelt	Bewegt den Cursor um zwei Positionen nach links
Alt+Pfeil nach rechts	Cursor nach rechts – doppelt	Bewegt den Cursor um zwei Positionen nach rechts
Strg+F3	Cursorauswahl	Simuliert eine Lightpen-Auswahl im aktuellen Feld
Alt+Entf	Wort löschen	Löscht drei Zeichen aus dem aktuellen Feld
Strg+5	Duplizieren	Fügt das DUP-Zeichen an der Cursorposition ein
Eingabe	Eingabe	Sendet die Eingabetaste an den Host
Ende	Feldende löschen	Löscht alle Daten ab der aktuellen Cursorposition bis zum aktuellen Feldende
Alt+F5	Eingabe löschen	Löscht alle Daten aus allen ungeschützten Feldern des aktuellen Bildschirms
Strg+Alt+F	Feldtrennzeichen	Schaltet die Anzeige der Feldtrennzeichen ein bzw. aus
Strg+6	Feldmarkierung	Fügt das Feldmarkierungszeichen an der Cursorposition ein
Pos1	Pos1	Bewegt den Cursor in das erste ungeschützte Feld auf dem Bildschirm
Einf	Einfügen	Ändert den Einfügemodus
Umschalt+Eingabe	Neue Zeile	Bewegt den Cursor in das nächste ungeschützte Feld
Strg+1	PA1	Sendet die PA1-Taste an den Host
Bild nach oben	PA1	Sendet die PA1-Taste an den Host
Strg+2	PA2	Sendet die PA2-Taste an den Host
Bild nach unten	PA2	Sendet die PA2-Taste an den Host
Strg+3	PA3	Sendet die PA3-Taste an den Host
F1 – F10	PF1 – PF10	Sendet die PF1, PF2...PF10-Tasten an den Host
Alt+1	PF11	Sendet die PF11-Taste an den Host
F11	PF11	Sendet die PF11-Taste an den Host
Alt+2	PF12	Sendet die PF12-Taste an den Host
F12	PF12	Sendet die PF12-Taste an den Host
Umschalt+F1	PF13	Sendet die PF13-Taste an den Host
Umschalt+F2	PF14	Sendet die PF14-Taste an den Host
Umschalt+F3	PF15	Sendet die PF15-Taste an den Host
Umschalt+F4	PF16	Sendet die PF16-Taste an den Host
Umschalt+F5	PF17	Sendet die PF17-Taste an den Host
Umschalt+F6	PF18	Sendet die PF18-Taste an den Host

Taste	Zuordnungen zu	Beschreibung
Umschalt+F7	PF19	Sendet die PF19-Taste an den Host
Umschalt+F8	PF20	Sendet die PF20-Taste an den Host
Umschalt+F9	PF21	Sendet die PF21-Taste an den Host
Umschalt+F10	PF22	Sendet die PF22-Taste an den Host
Alt+3	PF23	Sendet die PF23-Taste an den Host
Umschalt+F11	PF23	Sendet die PF23-Taste an den Host
Alt+4	PF24	Sendet die PF24-Taste an den Host
Umschalt+F12	PF24	Sendet die PF24-Taste an den Host
Strg+P	Druck	Sendet den Bildschirminhalt an den Drucker
Esc	Zurücksetzen	Setzt Tastaturfehler zurück
Strg+S	Systemanforderung	Sendet die SYSTEM REQUEST-Taste an den Host

Table 4-5 IBM 5250-Keyboard

Tasten	Zuordnungen zu	Beschreibung
Esc	Abruf	Sendet die Abruftaste an den Host
Strg+F2	Löschen	Entfernt den Bildschirminhalt und sendet die Löschtaste an den Host
Strg+F3	Cursorauswahl	Simuliert eine Lightpen-Auswahl im aktuellen Feld
Strg+Rücktaste	Rückschritt mit Löschen	Bewegt den Cursor um eine Position nach links
Strg+5	Duplizieren	Fügt das DUP-Zeichen an der Cursorposition ein
Strg+Ende	Feldende	Bewegt den Cursor an das Feldende
Ende	Feldende löschen	Löscht alle Daten ab der aktuellen Cursorposition bis zum aktuellen Feldende
Alt+Ende	Eingabe löschen	Löscht alle Daten aus allen ungeschützten Feldern des aktuellen Bildschirms
Alt+F5	Eingabe löschen	Löscht alle Daten aus allen ungeschützten Feldern des aktuellen Bildschirms
Strg+Eingabetaste	Feldende	Bewegt den Cursor aus einem Eingabefeld heraus
Nt+Minus	Feldende Minus	Bewegt den Cursor aus einem Feld für numerische Daten mit oder ohne Vorzeichen heraus
Strg+Minus	Feldende Minus	Bewegt den Cursor aus einem Feld für numerische Daten mit oder ohne Vorzeichen heraus
Nt+Plus	Feldende Plus	Bewegt den Cursor aus einem Feld für numerische Daten mit oder ohne Vorzeichen heraus
Strg+Plus	Feldende Plus	Bewegt den Cursor aus einem Feld für numerische Daten mit oder ohne Vorzeichen heraus

Tasten	Zuordnungen zu	Beschreibung
Strg+6	Feldmarkierung	Fügt das Feldmarkierungszeichen an der Cursorposition ein
Strg+H	Hilfe	Sendet die Hilfetaste an den Host
Strg+X	Hexadezimalmodus	Versetzt das Terminal in den Hexadezimalmodus
Pos1	Pos1	Bewegt den Cursor in das erste ungeschützte Feld auf dem Bildschirm
Einfügen	Einfügen	Ändert den Einfügemodus
Umschalt+Eingabe	Neue Zeile	Bewegt den Cursor in das nächste ungeschützte Feld
Strg+1	PA1	Sendet die PA1-Taste an den Host
Strg+2	PA2	Sendet die PA2-Taste an den Host
Strg+3	PA3	Sendet die PA3-Taste an den Host
F1 – F11	PF1 – PF11	Sendet die PF1, PF2....PF11-Tasten an den Host
Alt+1	PF11	Sendet die PF11-Taste an den Host
Alt+2	PF12	Sendet die PF12-Taste an den Host
F12	PF12	Sendet die PF12-Taste an den Host
Umschalt+1	PF13	Sendet die PF13-Taste an den Host
Umschalt+F2	PF14	Sendet die PF14-Taste an den Host
Umschalt+F3	PF15	Sendet die PF15-Taste an den Host
Umschalt+F4	PF16	Sendet die PF16-Taste an den Host
Umschalt+F5	PF17	Sendet die PF17-Taste an den Host
Umschalt+F6	PF18	Sendet die PF18-Taste an den Host
Umschalt+F7	PF19	Sendet die PF19-Taste an den Host
Umschalt+F8	PF20	Sendet die PF20-Taste an den Host
Umschalt+F9	PF21	Sendet die PF21-Taste an den Host
Umschalt+F10	PF22	Sendet die PF22-Taste an den Host
Alt+3	PF23	Sendet die PF23-Taste an den Host
Umschalt+F11	PF23	Sendet die PF23-Taste an den Host
Alt+4	PF24	Sendet die PF24-Taste an den Host
Umschalt+F12	PF24	Sendet die PF24-Taste an den Host
Strg+P	Drucken	Sendet den Bildschirminhalt an den Drucker
Strg	Zurücksetzen	Setzt Tastaturfehler zurück
Bild nach oben	Nach unten blättern	Sendet die Nach-unten-blättern-Taste an den Host
Bild nach unten	Nach oben blättern	Sendet die Nach-oben-blättern-Taste an den Host
Strg+Pos1	Feldanfang	Bewegt den Cursor an den Feldanfang

Tasten	Zuordnungen zu	Beschreibung
Strg+S	Systemanforderung	Sendet die SYSTEM REQUEST-Taste an den Host

Tabelle 4-6 VT-Tastaturbelegung

Tasten	Zuordnungen zu	Beschreibung
Strg+Untbr	Unterbrechungstaste	Sendet die Unterbrechungstaste an den Host
Strg+Eingabe	Eingabe	Sendet die Eingabetaste an den Host
Alt+F1	F1	Sendet die F1-Taste an den Host
Strg+F1	F11	Sendet die F11-Taste an den Host
Strg+F2	F12	Sendet die F12-Taste an den Host
Strg+F3	F13	Sendet die F13-Taste an den Host
Strg+F4	F14	Sendet die F14-Taste an den Host
Strg+F5	F15	Sendet die F15-Taste an den Host
Strg+F6	F16	Sendet die F16-Taste an den Host
Strg+F7	F17	Sendet die F17-Taste an den Host
Strg+F8	F18	Sendet die F18-Taste an den Host
Strg+F9	F19	Sendet die F19-Taste an den Host
Strg+F10	F20	Sendet die F20-Taste an den Host
Pos1	Suchen	Sendet die Suchen-Taste an den Host
F1	Anhalten	Sendet die Anhalten-Taste an den Host
Pause	Anhalten	Sendet die Anhalten-Taste an den Host
Einfg	Einfügen	Sendet die Einfügen-Taste an den Host
Strg+Einfg	Nt0	Sendet die Nt0-Taste des numerischen Tastenfelds an den Host
Strg+Ende	Nt1	Sendet die Nt1-Taste des numerischen Tastenfelds an den Host
Strg+Pfeil nach unten	Nt2	Sendet die Nt2-Taste des numerischen Tastenfelds an den Host
Strg+Bild nach unten	Nt3	Sendet die Nt3-Taste des numerischen Tastenfelds an den Host
Strg+Pfeil nach links	Nt4	Sendet die Nt4-Taste des numerischen Tastenfelds an den Host
Strg+Löschen	Nt5	Sendet die Nt5-Taste des numerischen Tastenfelds an den Host
Strg+Pfeil nach rechts	Nt6	Sendet die Nt6-Taste des numerischen Tastenfelds an den Host
Strg+Pos1	Nt7	Sendet die Nt7-Taste des numerischen Tastenfelds an den Host

Tasten	Zuordnungen zu	Beschreibung
Strg+Pfeil nach oben	Nt8	Sendet die Nt8-Taste des numerischen Tastenfelds an den Host
Strg+Bild nach oben	Ziffernblock 9	Sendet die Nt9-Taste des numerischen Tastenfelds an den Host
Strg+Alt-Plus	NtKomma	Sendet die NtKommataste des numerischen Tastenfelds an den Host
Strg+Plus	NtMinus	Sendet die NtMinustaste des numerischen Tastenfelds an den Host
Strg+Dezimal	NtPunkt	Sendet die NtPunkt-Taste des numerischen Tastenfeldes an den Host
Strg+Löschen	NtPunkt	Sendet die NtPunkt-Taste des numerischen Tastenfeldes an den Host
Bild nach unten	Nächster	Sendet die „Nächster-Bildschirm“-Taste an den Host
Strg+Pause	PF1	Sendet die PF1-Taste an den Host
Strg+Dividieren	PF2	Sendet die PF2-Taste an den Host
Strg+Multiplizieren	PF3	Sendet die PF3-Taste an den Host
Strg+Minus	PF4	Sendet die PF4-Taste an den Host
Bild nach oben	Vorherig	Sendet die „Vorhergehender Bildschirm“-Taste an den Host.
Löschen	Entfernen	Sendet die LösCHFunktion-Taste an den Host
Ende	Auswählen	Sendet die Selektieren-Taste an den Host.
Umschalt+F6	UDK6	Sendet die Benutzertaste 6 an den Host
Umschalt+F7	UDK7	Sendet die Benutzertaste 7 an den Host
Umschalt+F8	UDK8	Sendet die Benutzertaste 8 an den Host
Umschalttaste+F9	UDK9	Sendet die Benutzertaste 9 an den Host
Umschalttaste+F10	UDK10	Sendet die Benutzertaste 10 an den Host
Umschalt+Strg+F1	UDK11	Sendet die Benutzertaste 11 an den Host
Umschalt+Strg+F2	UDK12	Sendet die Benutzertaste 12 an den Host
Umschalt+Strg+F3	UDK13	Sendet die Benutzertaste 13 an den Host
Umschalt+Strg+F4	UDK14	Sendet die Benutzertaste 14 an den Host
Umschalt+Strg+F5	UDK15	Sendet die Benutzertaste 15 an den Host
Umschalt+Strg+F6	UDK16	Sendet die Benutzertaste 16 an den Host
Umschalt+Strg+F7	UDK17	Sendet die Benutzertaste 17 an den Host
Umschalt+Strg+F8	UDK18	Sendet die Benutzertaste 18 an den Host
Umschalt+Strg+F9	UDK19	Sendet die Benutzertaste 19 an den Host
Umschalt+Strg+F10	UDK20	Sendet die Benutzertaste 20 an den Host

Tabelle 4-7 UTS-Tastaturbelegung

Tasten	Zuordnungen zu	Beschreibung
F4	Änderungsbit löschen	Sendet die CLEARCHANGEBIT-Taste an den Host.
NtEingabe	Zeilenschaltung	Sendet eine Zeilenschaltung an den Host.
Strg+Bild nach unten	Anzeigeende löschen	Löscht den Text ab der Cursorposition bis zum Anzeigeende.
Strg+Bild nach oben	Anzeigeende/FCC löschen	Löscht alle Daten (einschließlich FCC-Informationen) von der Cursorposition bis zum Anzeigeende.
Strg+Ende	Feldende löschen	Löscht den Text ab der Cursorposition bis zum Feldende.
Strg+Umschalt+Ende	Zeilenende löschen	Löscht den Text ab der Cursorposition bis zum Zeilenende.
F7	FCC löschen	Löscht das Feldsteuerungszeichen.
Strg+Pos1	Pos1 löschen	Sendet die CLEAR_HOME-Taste an den Host.
Strg+H	Spaltentrennzeichen rechts	Sendet die COLUMN_SEP_RIGHT-Taste an den Host.
Strg+F1	Steuerungsseite	Sendet die CONTROL_PAGE-Taste an den Host.
Nt2	Cursor nach unten	Bewegt den Cursor eine Zeile nach unten.
Nt4	Cursor nach links	Bewegt den Cursor um eine Spalte nach links.
Nt6	Cursor nach rechts	Bewegt den Cursor um eine Spalte nach rechts.
Nt8	Cursor nach oben	Bewegt den Cursor eine Zeile nach oben.
Löschen	In Zeile löschen	Sendet die DELETE_IN_LINE-Taste an den Host.
Strg+Löschen	Auf Seite löschen	Sendet die DELETE_IN_PAGE-Taste an den Host.
Strg+Umschalt+Löschen	Zeile löschen	Löscht die Zeile an der Cursorposition.
Strg+Pfeil nach unten	Zeile duplizieren	Löscht die Zeile an der Cursorposition.
F8	FCC aktivieren	Aktiviert das Feldsteuerungszeichen.
Nt+-	Anzeigeende und übertragen	Sendet die EOD_AND_TRANSMIT-Taste an den Host.
Umschalt+Ende	Feldende	Bewegt den Cursor an das Feldende.
Ende	Zeilenende	Bewegt den Cursor an das Zeilenende.
Strg+Pfeil nach rechts	Seitenende	Bewegt den Cursor an das Seitenende.
Umschalt+Leertaste	Zeichen löschen	Löscht das Zeichen an der Cursorposition.
Strg+Umschalt+E	Euro-Zeichen	Sendet das Eurozeichen an den Host.
Strg+1 – Strg+9	F1 – F9	Sendet die F1- bis F9-Tasten an den Host.
Strg+0	F10	Sendet die F10-Taste an den Host.
Strg+-	F11	Sendet die F11-Taste an den Host.
Strg+=	F12	Sendet die F12-Taste an den Host.

Tasten	Zuordnungen zu	Beschreibung
Strg+Q	F13	Sendet die F13-Taste an den Host.
STRG+W	F14	Sendet die F14-Taste an den Host.
Strg+E	F15	Sendet die F15-Taste an den Host.
Strg+R	F16	Sendet die F16-Taste an den Host.
Strg+T	F17	Sendet die F17-Taste an den Host.
Strg+Y	F18	Sendet die F18-Taste an den Host.
Strg+U	F19	Sendet die F19-Taste an den Host.
Strg+I	F20	Sendet die F20-Taste an den Host.
Strg+O	F21	Sendet die F21-Taste an den Host.
Strg+P	F22	Sendet die F22-Taste an den Host
Umschalttaste+F3	FF	Sendet einen Papiervorschub an den Host.
F9	FCC erzeugen	Erzeugt ein Feldsteuerungszeichen.
Pos1	Pos1	Bewegt den Cursor in das erste Feld in Anzeige.
Strg+Umschalt+Leertaste	In Zeile einfügen	Sendet die INSERT_IN_LINE-Taste an den Host.
Ctrl+Leertaste	Auf Seite einfügen	Sendet die INSERT_IN_PAGE-Taste an den Host.
Strg+Umschalt+Einf	Zeile einfügen	Fügt eine neue Zeile in den Bildschirmspeicher ein.
Einf	Einfügemodus	Ändert den Einfügemodus.
F5	FCC suchen	Deaktiviert die Feldsteuerungszeichen und wechselt zum ersten Zeichen des nächsten Felds rechts vom Cursor.
F3	Warten-Meldung	Sendet die MESSAGE_WAIT-Taste an den Host.
Umschalttaste+F2	Neue Zeile	Bewegt den Cursor in eine neue Zeile.
NtUmschalt+2	Nächstes Feld	Bewegt den Cursor in das nächste Feld.
NtUmschalt+4	Nächstes Feld	Bewegt den Cursor in das nächste Feld
Bild nach unten	Bild nach unten	Sendet die Bild-ab-Taste an den Host.
Bild nach oben	Bild nach oben	Sendet die Bild-nach-oben-Taste an den Host.
NtUmschalt+6	Vorheriges Feld	Bewegt den Cursor in das vorherige Feld.
NtUmschalt+8	Vorheriges Feld	Bewegt den Cursor in das vorherige Feld.
Löschen	SOE-Zeichen	Sendet das SOE-Zeichen an den Host.
F12	SOE-Zeichen	Sendet das SOE-Zeichen an den Host.
Strg+Löschen	Tab setzen	Sendet die SET_TAB-Taste an den Host.
Strg+Tabulator	Tab setzen	Sendet die SET_TAB-Taste an den Host.
Umschalt+Pos1	Feldanfang	Bewegt den Cursor an den Feldanfang.
Strg+Pfeil nach links	Zeilenanfang	Bewegt den Cursor an den Zeilenanfang.

Tasten	Zuordnungen zu	Beschreibung
Strg+[Systemmodus	Sendet die SYSTEM_MODE-Taste an den Host.
Strg+J	Spaltentrennzeichen ändern	Ändert das Spaltentrennzeichen.
Strg+F12	Piepton für Warten-Meldung ändern	Sendet die TOGGLEMSGWAITBEEP-Taste an den Host.
Strg+L	Durchstreichen ändern	Ändert den Durchstreichungsmodus.
Strg+K	Unterstreichen ändern	Ändert den Unterstreichungsmodus.
Strg+Eingabetaste	Übertragen	Sendet den Anzeigehalt an den Host.
ScrollLock	Übertragen	Sendet den Anzeigehalt an den Host.
Nt++	Übertragen	Sendet den Anzeigehalt an den Host.
NtStrg+	Übertragen	Sendet den Anzeigehalt an den Host.
Esc	Seite „Entsperren“	Sendet die UNLOCK-Taste an den Host.
Strg+]	Workstationmodus	Sendet die WORKSTATION_MODE-Taste an den Host.

Konfigurieren von Makros für Benutzer

Im Bereich „Makro“ wählen Sie die auszuführenden Makros sowie den Zeitraum aus, in dem sie ausgeführt werden sollen.

- ♦ **Makro beim Start ausführen** – Wählen Sie aus, dass ein Makro beim Sitzungsstart automatisch ausgeführt wird.
- ♦ **Makro beim Verbinden ausführen** – Wählen Sie aus, dass ein Makro automatisch ausgeführt wird, wenn sich die Sitzung mit dem Host verbindet.
- ♦ **Makro beim Trennen der Verbindung ausführen** – Wählen Sie aus, dass ein Makro automatisch ausgeführt wird, wenn die Sitzung mit dem Host getrennt wird.

Verwandte Themen

[Verwenden der Makro-API](#)

[Beispielmakros](#)

[Erstellen von Makros](#)

Übertragen von Dateien vom und zum Host

Sie können Daten zwischen Ihrem Computer und einem 3270-Hostcomputer übertragen. Wählen Sie aus der Dropdownliste aus, welche IBM 3270-Betriebsumgebung der Host ausführt: CMS, TSO oder keine (Standardeinstellung).

Dateien können erst dann übertragen werden, wenn Sie mit dem Host verbunden sind. Nachdem Sie eine Verbindung hergestellt haben, wählen Sie im linken Bereich **IND\$FILE** aus, um das Dialogfeld „Dateiübertragung“ zu öffnen.

Festlegen von Optionen zum Kopieren und Einfügen

Sie können verschiedene Optionen für Kopier- und Einfügevorgänge festlegen.

Kopieroptionen

Markieren Sie Text, indem Sie die Maus über den gewünschten Bereich ziehen. Standardmäßig verwenden verschiedene Hosttypen unterschiedliche Auswahlmodi beim Kopieren von Text: IBM 3270-, 5250- und UTS-Hosts verwenden einen Blockauswahlmodus, während VT-Hosts einen linearen Auswahlmodus verwenden. Um zwischen den Auswahlmodi zu wechseln, halten Sie die **Alt**-Taste gedrückt, und wählen Sie den gewünschten Text aus.

- ♦ **Nur Eingabefelder kopieren** – Wählen Sie diese Option, um nur Daten aus Eingabefeldern zu kopieren. Daten aus geschützten Feldern werden beim Hinzufügen zur Zwischenablage durch Leerzeichen ersetzt.
- ♦ **Bei fehlender Auswahl gesamte Anzeige verwenden** – Diese Option wendet den Befehl „Kopieren“ auf die gesamte Terminalanzeige an, wenn nichts ausgewählt wurde.

Einfügeooptionen

Klicken Sie auf Einfügen, wenn Sie den Inhalt der Zwischenablage an der Cursorposition einfügen möchten.

- ♦ **Ursprüngliche Cursorposition nach dem Einfügen wiederherstellen** – Standardmäßig befindet sich der Hostcursor nach einem Einfügevorgang am Ende der Daten. Wählen Sie diese Option, um den Hostcursor auf seine Anfangsposition zurückzusetzen, nachdem der Einfügevorgang abgeschlossen ist.

Tastenkombinationen

Sie können bestimmte Tastenkombinationen verwenden, die verschiedenen Kopier-/ und Einfügevorgängen zugeordnet sind.

Tastenkombination	Hosttyp	Aktion
Strg+A	UTS, 3270, 5250	Alles auswählen
Umschalt+Pfeiltaste	UTS, 3270, 5250, VT	Größe der aktuellen Auswahl ändern
Strg+C	UTS, 3270, 5250	Kopieren
Strg+V	UTS, 3270, 5250	Einfügen
Strg+Umschalt+A	VT	Alles auswählen
Strg+Umschalt+C	VT	Kopieren
Strg+Umschalt+V	VT	Einfügen

Einrichten von Regeln für Benutzereinstellungen

Als Administrator können Sie auswählen, welche Optionen Benutzer für ihre Sitzungen konfigurieren dürfen. Die Optionen werden für die jeweilige Sitzung festgelegt. So können alle Benutzer, die Zugriff auf eine bestimmte Sitzung haben, ihre eigene Sitzungsinstanz konfigurieren.

- 1 Wählen Sie im linken Navigationsbereich die Option **Regeln für Benutzereinstellungen** aus.
- 2 Wählen Sie aus, welche Optionen die Benutzer konfigurieren können.
- 3 Klicken Sie auf **Speichern**.

Die einzelnen Konfigurationen gelten speziell für die Instanz der jeweiligen Sitzung und stehen nicht in Konflikt mit den Einstellungen der anderen Benutzer.

Verwandte Themen

[Konfigurieren von Anzeigeeinstellungen](#)
[Host-Bildschirmfarben anpassen](#)
[Konfigurieren von Hotspots](#)
[Zuordnen von Tasten](#)
[Konfigurieren von Makros für Benutzer](#)
[Übertragen von Dateien vom und zum Host](#)
[Aktivieren von benutzerdefinierten Steuerelementen](#)

Anpassen von Bildschirmen

Ihnen stehen zwei Optionen für die Anpassung der Bildschirme von Endbenutzern zur Verfügung:

- ♦ **Micro Focus Plus** – Aktiviert benutzerdefinierte Steuerelemente, um effizientere Abläufe und eine moderne und benutzerfreundliche Oberfläche zu bieten.

Mit dieser Option können Sie QuickInfos einfügen und nummerierte Listen mit fortschrittlichen Dropdownlisten ersetzen. Weiterhin können Sie Schaltflächen zur Hostoberfläche hinzufügen und zum Starten von Makros oder zum Durchführen anderer Optionen programmieren und die manuelle Datumseingabe durch einen Kalender ersetzen, aus dem Daten ausgewählt werden können.

- ♦ **Serverseitige Ereignisse** – Stellt prozeduralen Java-Code bereit, mit dem die Darstellung von Hostdaten erweitert und verbessert wird.

Mit serverseitigen Ereignissen können sie spezifische Ereignisse definieren und die Hostanwendung anhalten, ersetzen oder mit für die Sitzung bereitgestelltem Code unterbrechen. Außerdem können Sie die Optionen für die Fehlerbehandlung erweitern. Beispielsweise können Sie ein Ereignis hinzufügen, das Fehler erkennt und Code implementiert, um den Fehler abzufangen, zu kontrollieren und zu beheben.

Diese beiden Optionen werden im Bereich **Anpassung** konfiguriert.

- 1 Klicken Sie in der Symbolleiste auf **Einstellungen**, um den linken Navigationsbereich zu öffnen.
- 2 Klicken Sie auf **Anpassung**.

Verwandte Themen


[Aktivieren von benutzerdefinierten Steuerelementen](#)

[Verwenden von serverseitigen Ereignissen](#)

Aktivieren von benutzerdefinierten Steuerelementen

- 1 Klicken Sie im Bereich **Anpassung** auf **Plus aktivieren**.
- 2 Wählen Sie aus der Dropdownliste die gewünschte Transformationsdatei aus, oder laden Sie eine Datei aus einem anderen Speicherort. Transformationsdateien werden an der Dateinamenerweiterung `.rdar` erkannt.

Transformationsdateien (`.rdar`) werden im Rahmen eines Bildschirmeditor-Projekts generiert und stellen die Kriterien für benutzerdefinierte Steuerelemente bereit. Genaue Anweisungen zum Rumba-Bildschirmeditor finden Sie in den [Videos zum Bildschirmeditor](#).

- 3 Überprüfen Sie, ob die Einstellung der Anzahl an Millisekunden für die Hostaufbauverzögerung korrekt ist. Dies ist die Zeit, die der Server auf die Herstellung einer synchronen Verbindung wartet, bevor er davon ausgeht, dass der Host die Datenübertragung abgeschlossen hat.
- 4 Nachdem Sie zu Ihrer Sitzung zurückgekehrt sind, klicken Sie in der Symbolleiste auf , um die Optionen für benutzerdefinierte Steuerelemente anzuzeigen.

Endbenutzer können auf diese Option zugreifen, indem sie im Bereich **Benutzervoreinstellungen** die Funktion „Plus“ auswählen. Anschließend wird den Benutzern in der Symbolleiste das Plus-Symbol angezeigt, und sie können auf die Optionen für benutzerdefinierte Steuerelemente zugreifen.

Verwandte Themen

[Anpassen von Bildschirmen](#)

[Videos zum Bildschirmeditor](#)

[Einrichten von Regeln für Benutzereinstellungen](#)

Verwenden von serverseitigen Ereignissen

Über serverseitige Ereignisse können Sie prozeduralen Java-Code bereitstellen, mit dem die Darstellung von Hostdaten erweitert und verbessert wird.

- 1 Stellen Sie in Ihrer Entwicklungsumgebung die Anwendung Reflection ZFE SDK bereit.
- 2 Schreiben Sie den für die Aufgabe erforderlichen Java-Code, und kompilieren Sie den Code in einer JAR-Datei (Java Archive) in eine Java-Klasse.
- 3 Kopieren Sie die JAR-Datei in das Verzeichnis `install-dir\Microsoft\Focus\ReflectionZFE\sessionserver\webapps\zfe\WEB-INF\lib`, und starten Sie den Sitzungsserver neu.

Wenn das Ereignis auf mehreren Servern ausgeführt werden soll, kopieren Sie die JAR-Datei auf den jeweiligen Servern in das entsprechende Verzeichnis.

- 4 Geben Sie beim Hinzufügen und Konfigurieren der Sitzung (in der Verwaltungskonsole), die dem Ereignis zugewiesen werden soll, im Bereich **Anpassung** unter **Serverseitige Ereignisse** den vollständigen Klassennamen für das Ereignis an.
- 5 Starten Sie die Sitzung, und testen Sie das Ereignis.

Sie können die bereitgestellten Beispielergebnisse verwenden, die Sie bei den ersten Schritten unterstützen.

Verwandte Themen

[Anpassen von Bildschirmen](#)

5 Verwenden von Sitzungen

Alle Sitzungen, auf die Sie zugreifen können, sind in der Liste **Verfügbare Sitzungen** enthalten. Sitzungen werden anfänglich von Ihrem Systemadministrator erstellt und konfiguriert und können über einen verteilten URL (z. B. <https://<sessionserver>:7443/zfe>) abgerufen werden.

So öffnen Sie eine Sitzung

- 1 Klicken Sie auf die gewünschte Sitzung, um sie zu öffnen.
- 2 Interagieren Sie über die geöffnete Sitzung mit der Hostanwendung.
- 3 Sie können mehrere Instanzen einer konfigurierten Sitzung erstellen.

Verwandte Themen

- ♦ [Verwalten Ihrer Sitzungen](#)
- ♦ [Abmelden](#)

Verwalten Ihrer Sitzungen

Sie greifen auf Ihre Sitzungen über die Liste **Verfügbare Sitzungen** zu. Diese Sitzungen wurden Ihnen von Ihrem Administrator zugewiesen. Klicken Sie auf eine Sitzung, um sie zu öffnen.

Sie können gleichzeitig mehrere Sitzungen öffnen und über die oben am Bildschirm angeordneten Registerkarten problemlos zwischen den Sitzungen wechseln. Die aktuelle Sitzung befindet sich immer auf der äußeren linken Registerkarte und wird mit einem weißen Hintergrund und als fett formatierter Text angezeigt. Jede Sitzung bleibt 30 Minuten lang im aktiven Modus.

Bei der Interaktion mit der Sitzung haben Sie über die Symbolleiste Zugriff auf die verschiedenen verfügbaren Optionen. Sie können die Verbindung mit einer Sitzung trennen, die Sitzung schließen, „Kurzasten“ aktivieren und auf weitere Einstellungen zugreifen. Einige dieser Optionen sind nur verfügbar, wenn Ihr Administrator entsprechende Genehmigungen erteilt hat.

- ♦ „Arbeiten mit Kurztasten“, auf Seite 51
- ♦ „Kopieren und Einfügen“, auf Seite 53
- ♦ „Aufzeichnen von Makros für die Sitzung“, auf Seite 53
- ♦ „Abmelden“, auf Seite 54

Arbeiten mit Kurztasten

Klicken Sie in der Symbolleiste auf , um „Kurzasten“ anzuzeigen. „Kurzasten“ ist eine Terminaltastatur für die grafische Darstellung von Tasten auf einer Hosttastatur, mit der Sie schnell auf Terminaltasten zugreifen können. Durch Klicken auf die Terminaltasten der Kurztasten-Tastatur können Sie Terminaltastatureingaben an den Host senden. Wenn Sie den Mauszeiger über eine Taste bewegen, wird eine QuickInfo mit der Belegung angezeigt.

Table 5-1 3270-Kurztasten

Tasten	Beschreibung
PA-Tasten PA1 – PA3	Wählen Sie eine dieser Optionen aus, um die entsprechende PA-Taste an den Host zu senden.
PF-Tasten PF1 – PF24	Wählen Sie eine dieser Optionen aus, um die entsprechende PF-Taste an den Host zu senden.
Abruf	Wählen Sie diese Option aus, um die Abruftaste an den Host zu senden.
Eingabe	Wählen Sie diese Option aus, um die Eingabetaste an den Host zu senden.
Löschen	Wählen Sie diese Option aus, um von der aktuellen Cursorposition bis zum Ende des Eintrags alle Zeichen zu löschen.
Löschen	Setzt die Pufferpositionen für die aktive Partition auf null und den Antwortmodus auf die Standardeinstellung, sendet die Clear AID-Taste an den Host und verschiebt den Cursor in die obere linke Ecke.
Zurücksetzen	Entfernt das Eingabesperrsymbol und setzt den Einfügemodus zurück.
SYSRQ	Sendet die System Request-Taste an den Host.

Table 5-2 5250-Kurztasten

Tasten	Beschreibung
PF-Tasten PF1 – PF24	Wählen Sie eine dieser Optionen aus, um die entsprechende PF-Taste an den Host zu senden.
Abruf	Wählen Sie diese Option aus, um die Abruftaste an den Host zu senden.
Eingabe	Wählen Sie diese Option aus, um die Eingabetaste an den Host zu senden.
Löschen	Wählen Sie diese Option aus, um von der aktuellen Cursorposition bis zum Ende des Eintrags alle Zeichen zu löschen.
Löschen	Weist den Host an, sämtlichen vom Benutzer eingegebenen Text vom aktuellen Bildschirm zu löschen.
Zurücksetzen	Durch diese Option wird der Einfügemodus, der diakritische Modus oder der Hexadezimalmodus verlassen, Hilfe- und Systemanforderungsfunktionen werden beendet, Operatorfehlermeldungen werden gelöscht, die Anzeige für die Eingabesperre wird entfernt, und der Einfügemodus wird zurückgesetzt. Wählen Sie die Option „Zurücksetzen“ zweimal hintereinander, um den Plus CR-Modus zu beenden.
Nach oben blättern	Wählen Sie diese Option, wenn Sie um eine Seite des aktuellen Hostbildschirms nach unten blättern möchten. Diese Auswahl entspricht der Option „Bild nach unten“.
Nach unten blättern	Wählen Sie diese Option, wenn Sie um eine Seite des aktuellen Hostbildschirms nach oben blättern möchten. Diese Auswahl entspricht der Option „Bild nach oben“.

Tabelle 5-3 VT-Kurztasten

Tasten	Beschreibung
VT-Funktionstasten F1 – F20	Wählen Sie eine dieser Optionen aus, um die entsprechende VT-Funktionstaste an den Host zu senden.
PF-Tasten PF1 – PF24	Wählen Sie eine dieser Optionen aus, um die entsprechende PF-Taste an den Host zu senden.
Unterbrechungstaste	Sendet die Unterbrechungstaste an den Host.
Eingabe	Wählen Sie diese Option aus, um die Eingabetaste an den Host zu senden.
Suchen	Sendet die Suchen-Taste an den Host. <ESC>[1~
Einfügen	Ändert den Einfügemodus. <ESC>[2~
Entfernen	Sendet die LösCHFunktion-Taste an den Host. <ESC>[3~
Auswählen	Sendet die Selektieren-Taste an den Host. <ESC>[4~
Vorherig	Sendet die „Vorheriger Bildschirm“-Taste an den Host. <ESC>[5~
Nächster	Sendet die Suchen-Taste an den Host. <ESC>[6~

Kopieren und Einfügen

HINWEIS: Die Funktionen zum Kopieren und Einfügen werden in allen Browsern unterschiedlich behandelt. Einige Browser unterstützen die Verwendung von Tasten für Kopier- und Einfügevorgänge. Um diese Funktionen möglichst praktisch zu nutzen, sollten Sie Tastaturbefehle verwenden. Die Belegung der Tastaturbefehle kann je nach Betriebssystem voneinander abweichen. Unter Windows lautet sie: **STRG+C** für Kopiervorgänge und **STRG+V** für Einfügevorgänge.


So kopieren Sie Inhalte aus dem Terminal

- 1 Markieren Sie auf dem Terminalbildschirm den Bereich, der kopiert werden soll.
- 2 Klicken Sie in der Symbolleiste auf **Kopieren**, oder verwenden Sie den Tastaturbefehl **STRG+C**.

So fügen Sie Inhalte in den Terminalbildschirm ein

- 1 Bewegen Sie den Cursor zu der Position, an der Sie Inhalte einfügen möchten.
- 2 Verwenden Sie den Tastaturbefehl **STRG+V**, oder klicken Sie in der Symbolleiste auf **Einfügen**. Wenn Ihr Browser das Einfügen über die Symbolleiste nicht unterstützt, werden Sie aufgefordert, den Tastaturbefehl zu verwenden.

Aufzeichnen von Makros für die Sitzung

Klicken Sie in der Symbolleiste auf das Makro-Symbol , um mit der Erstellung eigener Makros zu beginnen. Wenn das Symbol nicht angezeigt wird, bitten Sie Ihren Administrator, die entsprechenden Genehmigungen einzurichten.

Unter [Erstellen von Makros](#) finden Sie Anweisungen zum Aufzeichnen von Makros.

Verwandte Themen

- ◆ [Erstellen von Makros](#)
- ◆ [Verwenden der Makro-API](#)

Abmelden

Öffnen Sie am oberen rechten Rand des Bildschirms die Dropdownliste für Ihren Benutzernamen, und wählen Sie **Abmelden** aus, um die Arbeit in der Hostanwendung zu beenden.

Verwandte Themen

- ◆ [Verwenden von Sitzungen](#)
- ◆ [Erstellen von Makros](#)

6 Druckvorgang

Sie können Bildschirminhalte an einen Drucker senden.

Die verfügbaren Einstellungen für die Einrichtung der Seite und der Ausrichtung hängen von den Einstellungen Ihres Browsers ab.

So drucken Sie den Bildschirminhalt

Die Inhalte des Terminalbildschirms werden über die Option „Bildschirminhalt drucken“ gedruckt. Die Symbolleiste oder sonstige Anzeigeinformationen sind in dem Druckauftrag nicht enthalten.

- 1 Navigieren Sie zu dem Bildschirm, den Sie drucken möchten.
- 2 Klicken Sie in der Symbolleiste auf „Bildschirm drucken“.
- 3 Verwenden Sie im Browser das Dialogfeld für Druckvorgänge, um die Einstellungen für den Drucker und die Einrichtung der Seite auszuwählen.

Verwandte Themen

[Anpassen von Bildschirmen](#)

[Mainframe-Dateiübertragung](#)

7 Erstellen von Makros

Ein Makro ist eine Folge von Tastatureingaben, die Sie aufzeichnen und ausführen können. Diese JavaScript-Makroprogramme werden für die Automatisierung von Interaktionen mit dem Terminal verwendet. Sie können mit allen unterstützten Geräten auf Makros zugreifen und sie ausführen.

Erweiterte Makros werden von Reflection ZFE als JavaScript aufgezeichnet und gespeichert, sodass Sie die aufgezeichneten Makros problemlos bearbeiten und ergänzen können. Sie können Makros für eine spätere Wiedergabe aufzeichnen oder Makros beim Programmstart und beim Verbinden bzw. Trennen der Verbindung zum Host ausführen lassen. Sie können Makros auch ganz neu schreiben, um komplexere Aufgaben auszuführen, die das Aufzeichnungsprogramm nicht erfassen kann.

Makros werden Benutzern auf zwei unterschiedliche Arten zur Verfügung gestellt: Sie werden von einem Administrator erstellt oder von Benutzern für ihre eigene persönliche Verwendung aufgezeichnet. Alle erweiterten Makros sind einer Sitzung zugeordnet und haben mit der Automatisierung der Hostinteraktion alle das gleiche Ziel. Der einzige Unterschied zwischen den beiden Bereitstellungsmethoden ist der Zugriff auf die Makros und die Verwaltung der Erstellung und Verfügbarkeit.

- ◆ **Durch Administratoren erstellte Makros**

Administratoren zeichnen Makros beim Erstellen der Sitzung auf. Die Makros gelten für eine bestimmte Sitzung und stehen allen Benutzern zur Verfügung, die über das Makro-Symbol in der Symbolleiste Zugriff auf die Sitzung haben. Administratoren können Makros für eine Wiedergabe beim Programmstart oder beim Verbinden bzw. Trennen der Verbindung zum Host zuweisen.

- ◆ **Durch Benutzer erstellte Makros**

Endbenutzermakros werden von Einzelpersonen für die Sitzungen erstellt, zu denen ihnen Zugriff gewährt wurde. Administratoren erteilen die Genehmigung für die Erstellung von Makros, indem Sie eine Regel für Benutzereinstellungen erstellen. Die Benutzer können mit eigenen Anmeldeinformationen oder über eine **Gast**-Rolle auf die Sitzung zugreifen. Makros, die von Gastbenutzern erstellt werden, stehen auch allen anderen Gastbenutzern zur Verfügung. Wenn sich Benutzer mit ihren Anmeldeinformationen anmelden, werden ihnen nur die selbst erstellten Makros angezeigt.

Erweiterte Makros werden in alphabetischer Reihenfolge in der Dropdownliste aufgelistet, die in der Symbolleiste abgerufen werden kann. Die von Endbenutzern erstellten Makros werden am Anfang der Liste angezeigt, gefolgt von einem aus drei grauen vertikalen Punkten bestehenden Symbol, das bei Auswahl die Optionen „Bearbeiten“ und „Löschen“ anzeigt. Die von Administratoren erstellten Makros werden hingegen ohne das Symbol angezeigt, da derartige Makros von Endbenutzern nicht angepasst werden können.

Verwandte Themen

[Wie arbeite ich mit Makros?](#)

Verfahren?

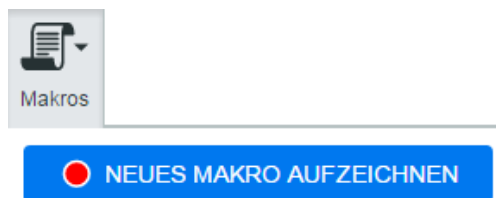
- ♦ „Aufzeichnen eines Makros“, auf Seite 58
- ♦ „Bearbeiten eines Makros“, auf Seite 58
- ♦ „Ausführen eines Makros“, auf Seite 59
- ♦ „Anhalten eines Makros“, auf Seite 59
- ♦ „Löschen eines Makros“, auf Seite 59
- ♦ „Anzeigen eigener Makros“, auf Seite 60
- ♦ „Fehlersuche für eigene Makros“, auf Seite 61


Aufzeichnen eines Makros

Wenn Sie die entsprechende Genehmigung von Ihrem Administrator erhalten haben, können Sie Makros aufzeichnen, um eine beliebige Folge von Hostaktionen zu automatisieren. Makros, die vom Administrator aufgezeichnet wurden und der aktuellen Sitzung zugewiesen sind, stehen Ihnen über die Makro-Dropdownliste zur Verfügung.

So zeichnen Sie ein Makro auf:

1. Klicken Sie in der Symbolleiste zunächst auf das Makro-Symbol und anschließend auf „Neues Makro aufzeichnen“.



2. Navigieren Sie durch die Hostanwendung, um die Schrittfolge aufzuzeichnen, die im Makro enthalten sein soll.
3. Klicken Sie in der Symbolleiste auf , um die Aufnahme anzuhalten. Der rote Punkt blinkt und zeigt somit an, dass die Aufzeichnung gerade durchgeführt wird.
4. Wenn Sie dazu aufgefordert werden, geben Sie dem Makro einen Namen.

Verwandte Themen

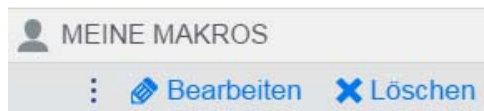
- ♦ [Ausführen eines Makros](#)
- ♦ [Bearbeiten eines Makros](#)
- ♦ [Fehlersuche für eigene Makros](#)


Bearbeiten eines Makros

Sie können selbst aufgezeichnete Makros bearbeiten. Diese Makros finden Sie in der Liste **Meine Makros**.

So bearbeiten Sie ein vorhandenes Makro:

1. Wählen Sie aus der Dropdownliste „Makro“ das zu bearbeitende Makro aus.




2. Klicken Sie auf die drei vertikalen Punkte, um das Feld zu erweitern.
3. Klicken Sie auf  [Bearbeiten](#), um den Makroeditor auszuführen.
Daraufhin wird der Makroeditor im linken Bereich geöffnet.
4. Nehmen Sie mit JavaScript die gewünschten Änderungen vor. Sie können das angepasste Makro über die Symbolleistensymbole im oberen Bereich des Editors ausführen und speichern.

Verwandte Themen

- ♦ [Ausführen eines Makros](#)
- ♦ [Aufzeichnen eines Makros](#)
- ♦ [Fehlersuche für eigene Makros](#)


Ausführen eines Makros

Um ein Makro auszuführen, wählen Sie es aus der Dropdownliste aus, und klicken Sie auf .

Verwandte Themen

- ♦ [Bearbeiten eines Makros](#)
- ♦ [Aufzeichnen eines Makros](#)
- ♦ [Fehlersuche für eigene Makros](#)

Anhalten eines Makros

Sie können ein Makro jederzeit über den Makroeditor oder die Symbolleiste anhalten. Klicken Sie dazu auf . Wenn Sie das Makro erneut ausführen möchten, kehren Sie zum Startbildschirm des Makros zurück.

Verwandte Themen

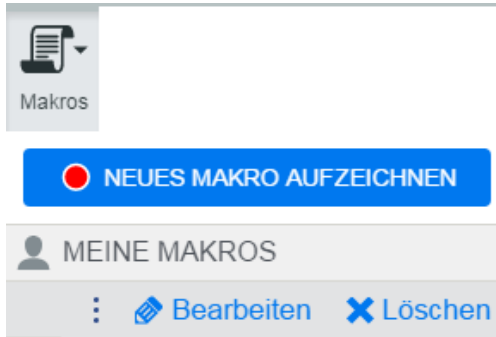
- ♦ [Bearbeiten eines Makros](#)
- ♦ [Aufzeichnen eines Makros](#)
- ♦ [Fehlersuche für eigene Makros](#)

Löschen eines Makros

1. Wählen Sie aus der Dropdownliste „Makro“ das zu löschende Makro aus.
2. Erweitern Sie das Feld, indem Sie auf das Symbol mit den drei vertikalen Punkten klicken.



3. Klicken Sie auf Löschen.



Verwandte Themen

- ◆ [Bearbeiten eines Makros](#)
- ◆ [Aufzeichnen eines Makros](#)
- ◆ [Fehlersuche für eigene Makros](#)

Anzeigen eigener Makros


Die Dropdownliste „Makro“ steht in der Symbolleiste allen Benutzern zur Verfügung, die zum Aufzeichnen von Makros berechtigt sind oder auf eine Sitzung zugreifen, in der Makros vorab von einem Administrator zur Verwendung in der entsprechenden Sitzung aufgezeichnet wurden.

Makros werden, je nach dem, wie sie aufgezeichnet wurden, unter **MEINE MAKROS** oder **MAKROS** aufgelistet.

Die der Sitzung zugewiesenen Makros werden allen Benutzern angezeigt, ganz gleich, ob sie sich mit ihren Anmeldeinformationen oder als Gast angemeldet haben. Die unter dem Titel „MEINE MAKROS“ aufgeführten Makros sind in alphabetischer Reihenfolge nach Name sortiert und werden den Benutzern angezeigt, die sie aufgezeichnet haben. Makros, die von einem Administrator aufgezeichnet und zu einer Sitzung hinzugefügt wurden, sind in alphabetischer Reihenfolge unter „MAKROS“ aufgelistet.



Makros

 NEUES MAKRO AUFZEICHNEN

 MEINE MAKROS

makro 1



 MAKROS

Neues Makro

Verwandte Themen

- ◆ [Bearbeiten eines Makros](#)
- ◆ [Aufzeichnen eines Makros](#)
- ◆ [Fehlersuche für eigene Makros](#)

Fehlersuche für eigene Makros

Makros werden in JavaScript geschrieben und im Browser ausgeführt. Daher sollte die Fehlersuche mit den in den Webbrowsern integrierten Werkzeugen durchgeführt werden. Moderne Browser sind mit einer Reihe äußerst wirksamer Werkzeuge zur Fehlersuche in JavaScript-Code ausgestattet. Sie können damit Haltepunkte positionieren, sich durch Code bewegen und Debug-Informationen ausgeben.

TIPP: In JavaScript wird die Groß- bzw. Kleinschreibung berücksichtigt. Achten Sie darauf, wenn Sie JavaScript-Code bearbeiten.

So führen Sie eine Fehlersuche für ein Makro durch:

1. Öffnen Sie das Makro für die Bearbeitung. Eine Anleitung dazu finden Sie unter [Bearbeiten eines Makros](#).
2. Öffnen Sie die Entwicklungswerkzeuge Ihres Browsers.

Tabelle 7-1 Browser-Unterstützung bei der Fehlersuche

Browser	Debugger öffnen
Mozilla Firefox 40.0.3	<ul style="list-style-type: none">♦ Öffnen Sie über die Symbolleiste das Menü, und wählen Sie „Entwickler“ aus.♦ Wählen Sie aus dem Menü „Web-Entwickler“ den Eintrag „Debugger“. Der Debugger wird im unteren Bereich geöffnet.
Google Chrome 45.0	<ul style="list-style-type: none">♦ Öffnen Sie über die Symbolleiste das Menü, und wählen Sie „Weitere Tools“ aus.♦ Wählen Sie „Entwicklungstools“, um den Debugger zu öffnen.
Microsoft Internet Explorer 11	<ul style="list-style-type: none">♦ Öffnen Sie über die Symbolleiste die „Einstellungen“, und wählen Sie „F12-Entwicklungstools“.♦ Öffnen Sie die Registerkarte „Debugger“.

Diese Anweisungen beziehen sich auf unterstützte Browser und hängen von den verwendeten Versionen ab.

3. Verwenden Sie eines der Werkzeuge im Makrocode, und führen Sie den Code aus.

♦ *debugger*

Den gründlichsten Ansatz für die Fehlersuche stellt die ‘debugger;’ -Anweisung dar. Wenn Sie diese Anweisungen in den Makrocode einfügen und den Makrocode ausführen, während die Entwicklungswerkzeuge des Browsers geöffnet sind, wird die Ausführung an diesen Zeilen angehalten. Sie können das Makro schrittweise ausführen und den Wert der lokalen Variablen sowie weitere zu prüfende Werte anzeigen.

Sie sollten mehrere ‘debugger;’-Anweisungen im Code platzieren, um zur richtigen Zeile zu gelangen. Aufgrund der asynchronen Eigenschaft von JavaScript kann die schrittweise Ausführung von Code herausfordernd sein. Wenn Sie mehrere ‘debugger;’-Anweisungen sorgfältig platzieren, können Sie diese Effekte jedoch abmildern.

Example 7-1 Debugger

```
-----  
var hostCommand = menuSelection + `[enter]`;  
debugger; // <- Browser's debugger will stop here  
ps.sendKeys(hostCommand);  
-----
```

♦ `console.log()`, `alert()`

Diese beiden Funktionen werden häufig für die Fehlersuche in JavaScript verwendet. Sie sind nicht so flexibel wie eine ‘debugger;’-Anweisung, bieten aber die Möglichkeit, Debug-Informationen schnell auszugeben. Diese Funktionen geben die Informationen an die JavaScript-Registerkarte „Konsole“ in den Entwicklungswerkzeugen des Browsers aus.

Example 7-2 `console.log(), alert()`

```
-----  
var hostCommand = menuSelection + `[enter]`;  
console.log('Command:' + hostCommand); // <- Will output the string to  
"Console" tab  
alert('Command:' + hostCommand); // Will pop up a small window containing  
the data  
ps.sendKeys(hostCommand);  
-----
```

◆ `ui.message()`

Die Reflection ZFE-Makro-API bietet eine `ui.message()`-Funktion, die der JavaScript-Funktion `alert()` sehr ähnelt. Sie können `ui.message()` auch für die Ausgabe von Debug-Informationen verwenden.

Example 7-3 `ui.message()`

```
-----  
var hostCommand = menuSelection + `[enter]`;  
ui.message('Command:' + hostCommand); // <- Will pop up a ZFE message  
window  
ps.sendKeys(hostCommand);  
-----
```

Hinweise für die Erstellung von Makros

◆ Schrittweise Ausführung und „yields“

Die `yield`-Anweisungen tragen zwar zum besseren Verständnis von Makros bei, können jedoch die schrittweise Ausführung des Codes mit dem Debugger erschweren. Daher sollten Sie entweder mehrere `debugger`-Anweisungen oder sorgfältig platzierte `debugger`-Anweisungen von `console.log()`-Aufrufen verwenden, um die richtigen Debug-Informationen auszugeben.

◆ Internet Explorer

Die Fehlersuche in Internet Explorer beinhaltet umgewandelten Code und kann herausfordernder sein als die Fehlersuche in anderen Browsern.

Verwandte Themen

- ◆ [Fehlersuche für eigene Makros](#)
- ◆ [Erstellen von Makros](#)

8

Verwenden der Makro-API

Makros werden in Reflection ZFE mithilfe von JavaScript aufgezeichnet und geschrieben. JavaScript ist eine beliebte und gängige Programmiersprache. Für die Sprache steht Ihnen eine Vielzahl an Lernressourcen und Werkzeugen zur Verfügung.

Die Reflection ZFE Makro-API setzt sich aus einer Reihe von Objekten zusammen, über die Sie mit dem Host interagieren, auf Bildschirmstatus warten und mit dem Benutzer interagieren können.

Zu berücksichtigende Punkte

Da JavaScript in einem einzelnen Thread läuft und für die Verwaltung der Ausführung ‚callback functions‘ und ‚promises‘ verwendet, ist der Code möglicherweise schwer nachvollziehbar. Reflection ZFE verbindet das Konzept von ‚promises‘ mit dem ‚yield‘-Schlüsselwort, damit Makrocode linearer geordnet werden kann.

- ◆ **Promises**

Promises sind Muster zur Vereinfachung von Funktionen, die an einem in der Zukunft liegenden Punkt Ergebnisse asynchron zurückgeben. Alle Funktionen der Typen ‚wait‘ und ‚ui‘ in der Reflection ZFE-Makro-API geben promise-Objekte zurück.

- ◆ **Yield**

Reflection ZFE-Makros verwenden das ‚yield‘-Schlüsselwort, um die Ausführung des Makros zu blockieren, bis ein ‚promise‘-Objekt aufgelöst oder ausgeführt wurde. Wenn also ‚yield‘ vor einer beliebigen ‚wait‘- oder ‚ui‘-Funktion gesetzt wird, wird die Ausführung des Makros angehalten, bis die Ausführung dieser Funktion abgeschlossen ist. Sie können das ‚yield‘-Schlüsselwort vor jeder Funktion platzieren, die ‚promise‘ zurückgibt, also auch für Ihre benutzerdefinierten Funktionen.

HINWEIS: Die Funktion zum Blockieren der Makroausführung durch die Kombination von ‚yield‘ und ‚promises‘ wird in der Funktion `createMacro()` aktiviert.

Fehlermeldungen

Fehler werden in Makros mithilfe der ‚try / catch‘-Anweisung behandelt. Einige der API-Funktionen können Fehler ausgeben, wenn beispielsweise Bedingungen nicht erfüllt werden können oder eine Zeitüberschreitung eintritt. Der ausgegebene Fehler wird in die ‚catch‘-Anweisung aufgenommen. Sie können kleinere Codeblöcke in einer ‚try / catch‘-Anweisung umbrechen, um Fehler detaillierter zu behandeln. Makro-Entwickler können Fehler auch über `throw new Error('Helpful error message');` ausgeben.

- ◆ [„Makro-API-Objekte“, auf Seite 65](#)
- ◆ [„Beispielmakros“, auf Seite 91](#)

Makro-API-Objekte

Sie können Makros mithilfe der Makro-API erstellen. Für die standardmäßige Verwendung in Makros stehen vier primäre Objekte zur Verfügung:

- ◆ [Session](#)

Bei 'Session' handelt es sich um den Hauptzugriffspunkt zum Host. Mit dem 'Session'-Objekt stellen Sie eine Verbindung zum PresentationSpace-Objekt her, trennen die Verbindung zu diesem Objekt und gewähren Zugriff darauf.

- ◆ [PresentationSpace](#)

Das PresentationSpace-Objekt stellt den Bildschirm dar und bietet zahlreiche allgemeine Funktionen wie das Abrufen und Einrichten der Cursorposition, das Senden von Daten an den Host und das Lesen auf dem Bildschirm. Der Abruf dieses Objekts erfolgt über

```
session.getPresentationSpace().
```

- ◆ [Wait](#)

Dieses Objekt bietet eine einfache Möglichkeit, auf das Auftreten der verschiedenen Hoststatus zu warten, bevor weitere Daten gesendet oder auf dem Bildschirm gelesen werden. Sie können beispielsweise darauf warten, dass sich der Cursor an einer bestimmten Position befindet, dass Text an einer bestimmten Position des Bildschirms angezeigt wird oder einfach für eine festgelegte Zeit. Alle Aufrufe der Funktion 'Wait' erfordern das 'yield'-Schlüsselwort, das weiter unten beschrieben wird.

- ◆ [User Interface](#)

Das UI-Objekt ist in Ihrem Makro automatisch als 'ui'-Variable verfügbar. Es stellt die grundlegenden Funktionen der Benutzeroberfläche bereit. Mit diesem Objekt werden dem Benutzer Daten angezeigt oder Informationen abgefragt. Alle Aufrufe der 'UI'-Funktion erfordern das [yield](#)-Schlüsselwort.

Weitere verfügbare Objekte

- ◆ [Attribute](#)
- ◆ [AttributeSet](#)
- ◆ [AutoSignon](#)
- ◆ [Color](#)
- ◆ [ControlKey](#)
- ◆ [DataCell](#)
- ◆ [Dimension](#)
- ◆ [Field](#)
- ◆ [FieldList](#)
- ◆ [OIA](#)
- ◆ [OIAStatus](#)
- ◆ [Position](#)
- ◆ [SessionType](#)
- ◆ [StatusSet](#)

Attribute

Verwenden Sie das Attribute-Objekt gemeinsam mit dem AttributeSet-Objekt zum Decodieren der Formatierungsinformationen in der Datenzelle.

Tabelle 8-1 Attribute

Attribut	Beschreibung
PROTECTED	Zeigt eine geschützte Datenzelle an.
MODIFIED	Zeigt eine geänderte Datenzelle an.
NUMERIC_ONLY	Zeigt den Anfang einer ausschließlich numerischen Datenzelle an.
ALPHA_NUMERIC	Zeigt eine alphanumerische Datenzelle an.
HIGH_INTENSITY	Zeigt an, ob die Datenzelle Text mit hoher Intensität enthält.
HIDDEN	Zeigt an, ob die Datenzelle ausgeblendeten Text enthält.
PEN_DETECTABLE	Gibt an, ob die Datenzelle von Stiften erkannt wird.
ALPHA_ONLY	Zeigt eine ausschließlich alphanumerische Datenzelle an.
NUMERIC_SHIFT	Zeigt den Anfang eines numerischen Wechsels an. Feld
NUMERIC_SPECIAL	Zeigt an, dass die Datenzelle den Anfang eines numerischen Sonderfelds kennzeichnet.
KATAKANA_SHIFT	Zeigt einen Abschnitt mit Katakana-Text an.
MAGNETIC_STRIPE	Zeigt an, dass die Datenzelle den Anfang eines Magnetstreifenfelds kennzeichnet.
SIGNED_NUMERIC_ONLY	Zeigt an, dass es sich bei der Datenzelle um ein Feld für numerische Daten handelt.
TRANSMIT_ONLY	Zeigt an, dass es sich bei der Datenzelle um ein Feld nur für die Übertragung handelt.
FIELD_END_MARKER	Zeigt an, dass die Datenzelle das Ende eines geänderten Felds kennzeichnet.
FIELD_START_MARKER	Zeigt an, dass die Datenzelle den Anfang eines geänderten Felds kennzeichnet.
SPECIAL_EMPHASIS_PROTECTED	Zeigt ein geschütztes Feld mit besonderer Hervorhebung an.
TAB_STOP	Zeigt an, dass die Datenzelle einen TAB-Stopp enthält.
REVERSE	Zeigt an, dass die Datenzelle in umgekehrter Darstellung angezeigt wird.
BLINKING	Zeigt an, dass die Datenzelle blinkenden Text enthält.
RIGHT_JUSTIFIED	Zeigt an, dass die Datenzelle den Anfang eines rechts ausgerichteten Felds kennzeichnet.
LEFT_JUSTIFIED	Zeigt an, dass die Datenzelle den Anfang eines links ausgerichteten Felds kennzeichnet.
LOW_INTENSITY	Zeigt an, dass die Datenzelle Text mit niedriger Intensität enthält.
UNDERLINE	Zeigt an, dass die Datenzelle unterstrichenen Text enthält.
DOUBLE_BYTE	Zeigt an, dass die Datenzelle Doppelbyte-Text enthält.
COLUMN_SEPARATOR	Zeigt an, dass die Datenzelle ein Spaltentrennzeichen enthält.
BOLD	Zeigt an, dass die Datenzelle fett formatierten Text enthält.

Attribut	Beschreibung
DOUBLE_WIDTH	Zeigt an, dass die Datenzelle ein Feld mit doppelter Breite kennzeichnet.
DOUBLE_HEIGHT_TOP	Zeigt eine Datenzelle mit doppelter oberer Höhe an.
DOUBLE_HEIGHT_BOTTOM	Zeigt eine Datenzelle mit doppelter unterer Höhe an.
CONTROL_PAGE_DATA	Zeigt an, dass die Datenzelle Steuerungsseitendaten enthält.
RIGHT_COLUMN_SEPARATOR	Zeigt an, dass die Datenzelle ein rechtes Spaltentrennzeichen enthält.
LEFT_COLUMN_SEPARATOR	Zeigt an, dass eine Datenzelle ein linkes Spaltentrennzeichen enthält.
UPPERSCORE	Zeigt an, dass die Datenzelle einen Überstrich enthält.
STRIKE_THROUGH	Zeigt an, dass die Datenzelle durchgestrichenen Text enthält.

AttributeSet

Mit dem AttributeSet-Objekt können Benutzer die in der Datenzelle enthaltenen Attribute decodieren. Das AttributeSet-Objekt gibt die im [Attribute](#)-Objekt definierten Werte zurück. Wenn sie gemeinsam verwendet werden, können Sie die Formatierungsinformationen aus der Datenzelle abrufen.

Tabelle 8-2 AttributeSet

METHODEN

<code>contains(attribute)</code>	Gibt an, ob der Satz das festgelegte Attribute enthält.
	Parameter
	{Number} zu prüfendes Attribut.
	Ausgabe
	{Boolean} True, wenn das Attribut im Satz enthalten ist.
<code>isEmpty()</code>	Gibt an, ob der Attributsatz leer ist.
	Ausgabe
	{Boolean} True, wenn der Satz leer ist.
<code>size()</code>	Gibt die Anzahl der in einem Satz enthaltenen Attribute an.
	Ausgabe
	{Number} Die Attributanzahl.
<code>toArray()</code>	Konvertiert den internen Attributsatz in ein Array.
	Ausgabe
	{Number[]} Array von Attributwerten in dem Satz.
<code>toString()</code>	Konvertiert den internen Attributsatz in eine Zeichenfolge.
	Ausgabe
	{String} Durch Leerzeichen eingeschlossene Namen von im Satz enthaltenen Attributen.

METHODEN

`forEach(callback, thisArg)` Funktion zum Durchlaufen der einzelnen Elemente im Attributsatz.

Parameter

{forEachCallback} Rückruffunktion zum Ausführen eines bestimmten Vorgangs. Wird gemeinsam mit dem Namen des jeweiligen Attributs im Satz aufgerufen.

{Object} 'thisArg' Optionaler Verweis auf ein Kontext-Objekt.

`forEachCallback(string, object)` Eine durch Benutzer bereitgestellte Rückruffunktion, mit der Sie das Verhalten bereitstellen. Wird als Rückrufparameter für 'forEach' verwendet.

Parameter

{String} String Name eines Attributs in dem Attributsatz.

{Object} thisArg Optionaler Verweis auf ein Kontext-Objekt.

Farbe

Farbkonstanten zur Verwendung für die Vordergrund- und Hintergrundfarben von DataCell-Objekten.

Tabelle 8-3 Farbkonstanten

Farbe	Beschreibung	Numerischer Wert
BLANK_UNSPECIFIED	Keine Farbe angegeben	0
BLUE	Blau	1
GREEN	Grün	2
CYAN	Cyan	3
RED	Rot	4
MAGENTA	Magenta	5
YELLOW	Gelb	6
WHITE_NORMAL_INTENSITY	Weiß mit normaler Intensität	7
GRAY	Grau	8
LIGHT_BLUE	Hellblau	9
LIGHT_GREEN	Hellgrün	10
LIGHT_CYAN	Cyan (hell)	11
LIGHT_RED	Hellrot	12
LIGHT_MAGENTA	Magenta (hell)	13
BLACK	Schwarz	14
WHITE_HIGH_INTENSITY	Weiß mit hoher Intensität	15
BROWN	Braun	16

Farbe	Beschreibung	Numerischer Wert
PINK	Rosa	17
TURQUOISE	Türkis	18

ControlKey

Das ControlKey-Objekt definiert Konstanten zum Senden von Cursor-Steuertasten und Hostbefehlen mithilfe der sendKeys-Methode. Konstanten sind für die folgenden Hosttypen verfügbar:

- ♦ [IBM 3270](#)
- ♦ [IBM 5250](#)
- ♦ [VT](#)
- ♦ [UTS](#)

IBM 3270

Tabelle 8-4 IBM 3270

Schlüsselwort	Beschreibung
ALTVIEW	Alternative Ansicht
ATTN	Abruf
BACKSPACE	Rückschritt
BACKTAB	Rücktabulator
CLEAR	Löschen oder Bildschirminhalt löschen
CURSOR_SELECT	Cursorauswahl
DELETE_CHAR	Löschen, Zeichen löschen
DELETE_WORD	Wort löschen
DEST_BACK	Rückschritt mit Löschen
DEV_CANCEL	Geräteabbruch
DOWN	Cursor nach unten
DSPSOSI	SO/SI anzeigen
DUP	Feld duplizieren
END_FILE	Feldende
ENTER	Eingabe
ERASE_EOF	Feldende löschen
ERASE_FIELD	Feld löschen
ERASE_INPUT	Eingabe löschen
FIELD_MARK	Feldmarkierung
HOME	Cursor Pos1

Schlüsselwort	Beschreibung
IDENT	Ident
INSERT	Einfügen
LEFT_ARROW	Cursor nach links
LEFT2	Cursor um zwei Positionen nach links
NEW_LINE	Neue Zeile
PA1 – PA3	PA1 – PA3
PF1 – PF24	PF1 – PF24
PAGE_DOWN	Bild nach unten
PAGE_UP	Bild nach oben
RESET	Zurücksetzen, Terminal zurücksetzen
RIGHT2	Cursor um zwei Positionen nach rechts
RIGHT_ARROW	Cursor rechts, rechts
SYSTEM_REQUEST	Systemanforderung
TAB	Tabulatortaste
UP	Cursor nach oben

IBM 5250

Tabelle 8-5 IBM 5250

Schlüsselwort	Beschreibung
ALTVIEW	Alternative Ansicht
ATTN	Abruf
AU1 – AU16	AU1 – AU16
BACKSPACE	Rückschritt
BACKTAB	Rücktabulator
BEGIN_FIELD	Feld beginnen
CLEAR	Löschen
DELETE_CHAR	Löschen, Zeichen löschen
DEST_BACK	Rückschritt mit Löschen
DOWN	Cursor nach unten
DSPSOSI	SO/SI anzeigen
DUP	Feld duplizieren
END_FILE	Feldende
ENTER	Eingabe

Schlüsselwort	Beschreibung
ERASE_EOF	Feldende löschen
ERASE_FIELD	Feld löschen
ERASE_INPUT	Eingabe löschen
FIELD_EXT	Feldende
FIELD_MINUS	Feld Minus
FIELD_PLUS	Feld Plus
FIELD_MARK	Feldmarkierung
HELP	Hilfeanforderung
HEXMODE	Hexadezimalmodus
HOME	Cursor Pos1
INSERT	Einfügen
LEFT_ARROW	Cursor nach links
NEW_LINE	Neue Zeile
PA1 – PA3	PA1 – PA3
[PF1 – PF24	PF1 – PF24
[print]	Drucken
RESET	Zurücksetzen, Terminal zurücksetzen
RIGHT_ARROW	Cursor rechts, rechts
PAGE_UP	Bild nach oben
PAGE_DOWN	Bild nach unten
SYSTEM_REQUEST	Systemanforderung
TAB	Tabulator
UP	Cursor nach oben
VT	

Tabelle 8-6 VT

Schlüsselwörter	Beschreibung
BACKSPACE	Rückschritt
BREAK	Unterbrechungstaste
CLEAR	Löschen oder Bildschirminhalt löschen
CURSOR_SELECT	Cursorauswahl
DELETE_CHAR	Löschen, Zeichen löschen
DOWN	Cursor nach unten

Schlüsselwörter	Beschreibung
EK_FIND	Nt Suche bearbeiten
EK_INSERT	Nt Einfügen bearbeiten
EK_NEXT	Nt Nächster bearbeiten
EK_PREV	Nt Vorheriger bearbeiten
EK_REMOVE	Nt Löschen bearbeiten
EK_SELECT	Nt Auswählen bearbeiten
ENTER	Eingabe
END_FILE	Feldende
F1 - F24	F1 - F24
HOLD	Halten
HOME	Pos1
INSERT	Einfügen
KEYPAD_COMMA	NtKomma
KEYPAD_DOT	NtDezimal
KEYPAD_MINUS	NtMinus
KEYPAD_ENTER	NtEingabe
KEYPAD0 - KEYPAD9	Nt0 - Nt9
LEFT_ARROW:	Cursor nach links
PF1 - PF20	PF1 - PF20
PAGE_DOWN	Bild nach unten
PAGE_UP	Bild nach oben
RESET	Zurücksetzen, Terminal zurücksetzen
RETURN	Zurück, Wagenrücklauf
RIGHT_ARROW	Cursor rechts, rechts
TAB	Tabulatortaste
UDK16 – UDK20	Benutzertaste 6 – Benutzertaste 20
UP	Cursor nach oben

UTS

Tabelle 8-7 UTS

Schlüsselwort	Beschreibung
BACKSPACE	Bewegt den Cursor in die vorherige Tabulatorposition auf dem Bildschirm.

Schlüsselwort	Beschreibung
BACKTAB	Rücktabulator <Umschalt> <Tabulator>
CHAR_ERASE	Löscht Zeichen an der Cursorposition und erweitert den Cursor.
CLEAR_DISPLAY	Bildschirminhalt löschen
CLEAR_EOD	Bis zum Anzeigende löschen
CLEAR_EOF	Bis zum Feldende löschen
CLEAR_EOL	Bis zum Zeilenende löschen
CLEAR_FCC	Feldsteuerungszeichen löschen
CLEAR_HOME	Bildschirminhalt löschen und Cursor Pos1
CONTROL_PAGE	Blendet die Steuerungsseite ein oder aus
DELETE_LINE	Löscht die Zeile, in der sich der Cursor befindet, und verschiebt die restlichen Zeilen um eine Zeile nach oben.
DOWN	Verschiebt den Cursor um eine Zeile nach unten. Umbruch erfolgt am unteren Ende.
DELIN_LINE	Löscht das Zeichen unterhalb des Cursors und verschiebt die restlichen Zeichen auf der Zeile nach links.
DELIN_PAGE	Löscht das Zeichen unterhalb des Cursors und verschiebt die restlichen Zeichen auf der Seite nach links.
DUP_LINE	Erstellt eine Kopie der aktuellen Zeile und überschreibt die nächste Zeile mit dem Duplikat.
EURO	Fügt das Euro-Zeichen ein.
END_FIELD	Bewegt den Cursor an das Ende des aktuellen Felds.
END_PAGE	Bewegt den Cursor an das Ende der aktuellen Seite.
F1 - F22	Funktionstasten F1-F22
HOME	Bewegt den Cursor an den Anfang der aktuellen Seite (Zeile 1, Spalte 1)
INSERT	Wechselt zwischen dem Einfügemodus und dem Überschreibmodus.
INSERT_IN_LINE	Fügt ein Leerzeichen an der Cursorposition ein und verschiebt die restlichen Zeichen auf der Zeile nach rechts. Das Zeichen in der äußeren rechten Spalte der Zeile wird verworfen.
INSERT_IN_PAGE	Fügt ein Leerzeichen an der Cursorposition ein und verschiebt die restlichen Zeichen auf der Seite nach rechts. Das Zeichen in der äußeren rechten Spalte der jeweiligen Zeile wird verworfen.
INSERT_LINE	Fügt eine neue Zeile an der Cursorzeile ein und verschiebt die restlichen Zeilen nach unten. Die letzte Zeile auf der Seite wird verworfen.
LEFT_ARROW	Bewegt den Cursor um eine Position nach links und fügt ggf. einen Umbruch ein.
LOCATE_FCC	Sucht nach dem nächsten Feldsteuerungszeichen auf dem Bildschirm.

Schlüsselwort	Beschreibung
MSG_WAIT	Ruft Meldungen in der Warteschleife des Terminals ab.
RETURN	Wagenrücklauf
RIGHT_ARROW	Bewegt den Cursor um eine Position nach rechts und fügt ggf. einen Umbruch ein.
SOE	Fügt das Zeichen für den Anfang des Eintrags ein.
START_OF_FIELD	Bewegt den Cursor an den Feldanfang.
START_OF_LINE	Bewegt den Cursor zur ersten Spalte der aktuellen Zeile.
TAB	Bewegt den Cursor in die nächste Tabulatorposition auf dem Bildschirm.
TOGGLE_COL_SEP	Ändert das Spaltentrennzeichen-Attribut.
TOGGLE_STRIKE_THRU	Ändert das Durchstreichungs-Attribut in der aktuellen Datenzeile.
TOGGLE_UNDERLINE	Ändert das Unterstreichungs-Attribut in der aktuellen Datenzeile.
TRANSMIT	Überträgt die geänderten Felddaten an den Host.
UNLOCK	Sendet die UNLOCK-Taste an den Host.
UP	Bewegt den Cursor eine Zeile nach oben und fügt ggf. einen Umbruch ein.

DataCell

Das DataCell-Objekt stellt Informationen zu einer bestimmten Position auf einem Terminalbildschirm bereit.

Tabelle 8-8 DataCell

METHODEN

<code>getPosition()</code>	Gibt die Position dieser Datenzeile auf dem Bildschirm zurück. Ausgabe {Position} Die Position dieser Datenzeile auf dem Bildschirm
<code>getChar()</code>	Ruft das mit der Zelle verknüpfte Zeichen ab. Ausgabe {String} Das mit der Zelle verknüpfte Zeichen.
<code>getAttributes()</code>	Gibt die für diese Datenzeileninstanz angegebene Attributgruppe zurück. Weitere Informationen hierzu finden Sie in AttributeSet . Ausgabe {AttributeSet} Von Attributen für diese Datenzeileninstanz.

METHODEN

<code>getForegroundColor()</code>	Gibt die Vordergrundfarbe für diese Datenzelle gemäß der Definition im <code>Color</code> -Objekt zurück. Ausgabe {Number} Vordergrundfarbe für diese Datenzelle. Die Farbe wird im <code>Farbe</code> -Objekt definiert.
<code>getBackgroundColor()</code>	Gibt die Hintergrundfarbe für diese Datenzelle gemäß der Definition im <code>Color</code> -Objekt zurück. Ausgabe {Number} Hintergrundfarbe für diese Datenzelle. Die Farbe wird im <code>Farbe</code> -Objekt definiert.
<code>toString</code>	Konvertiert die interne Datenzelle in eine Zeichenfolge. Ausgabe {String} Die Zeichenfolgendarstellung einer Datenzelle.
<code>isFieldDelimiter()</code>	Testet, ob die Zelle ein Feldtrennzeichen darstellt. Ausgabe {Boolean} 'True', wenn diese Zelle ein Feldtrennzeichen darstellt, ansonsten 'False'.

Dimension

Stellt die Größe des Bildschirms oder des Bildschirmbereichs dar.

Tabelle 8-9 Dimension

Methode	
<code>Dimension(rows, cols)</code>	Erstellt eine neue Dimension-Instanz. Parameter {Number} rows Größe der Bildschirmzeilen {Number} cols Größe der Bildschirmspalten

Field

Verwenden Sie das `Field`-Objekt gemeinsam mit `FieldList`, um die auf dem Bildschirm in einem Feld enthaltenen Informationen abzurufen.

Tabelle 8-10 Feld

Methode	
<code>getAttributes()</code>	<p>Gibt die für diese Feldinstanz angegebene Attributgruppe zurück. Weitere Informationen hierzu finden Sie in AttributeSet.</p> <p>Ausgabe</p> <p>{AttributeSet} Die Attributgruppe für dieses Feld</p>
<code>getForegroundColor()</code>	<p>Gibt die Vordergrundfarbe des Felds zurück.</p> <p>Ausgabe</p> <p>{Number} Die Vordergrundfarbe für dieses Feld. Diese Werte werden im Color-Objekt definiert.</p>
<code>getBackgroundColor()</code>	<p>Gibt die Hintergrundfarbe des Felds zurück.</p> <p>Ausgabe</p> <p>{Number} Die Hintergrundfarbe für dieses Feld. Diese Werte werden im Color-Objekt definiert.</p>
<code>getStart()</code>	<p>Gibt die Anfangsposition des Felds zurück. Die Anfangsposition ist die Position des ersten Zeichens im Feld. Einige Hosttypen verwenden eine Zeichenposition zum Speichern von Feldebeneattributen. In diesem Fall wird die Attributposition nicht als Anfangsposition betrachtet.</p> <p>Ausgabe</p> <p>{Position} Anfangsposition des Felds.</p> <p>Ergebnis</p> <p>{RangeError} Für Felder ohne Länge.</p>
<code>getEnd()</code>	<p>Gibt die Endposition des Felds zurück. Die Endposition ist die Position im Darstellungsbereich mit dem letzten Zeichen des Felds.</p> <p>Ausgabe</p> <p>{Position} Endposition des Felds.</p> <p>Ergebnis</p> <p>{RangeError} Für Felder ohne Länge.</p>
<code>getLength()</code>	<p>Gibt die Länge des Felds zurück. Bei Hosttypen, die zum Speichern der Feldattribute eine Zeichenposition verwenden, ist die Feldattributposition nicht in der Feldlänge enthalten.</p> <p>Ausgabe</p> <p>{Number} Länge des Felds.</p>
<code>getDataCells()</code>	<p>Ruft die Datenzellen ab, die dieses Feld bilden. Weitere Informationen hierzu finden Sie in DataCell.</p> <p>Ausgabe</p> <p>{DataCell []} Datenzellen, die dieses Feld bilden.</p>

Methode

<code>getText()</code>	Ruft den Text aus dem Feld ab. Ausgabe {String} Feldtext.
<code>setText()</code>	Legt den Feldtext fest. Bei bestimmten Hosttypen wie VT wird der Text sofort an den Host übertragen. Bei anderen Hosttypen wiederum wird der Text erst dann an den Host übertragen, wenn die AID-Taste aufgerufen wird. Wenn der Text kürzer als das Feld ist, wird er im Hostfeld platziert, und der Rest des Felds wird gelöscht. Wenn der Text länger als das Hostfeld ist, wird so viel Text wie möglich im Feld platziert. Parameter {String} Text, der im Feld gesetzt wird. Ergebnis {Error} Wenn das Feld geschützt ist.
<code>clearField()</code>	Löscht das aktuelle Feld emulationsspezifisch. Ergebnis {Error} Wenn das Feld geschützt oder Löschen nicht unterstützt wird.
<code>getPresentationSpace()</code>	Ruft den Darstellungsbereich ab, der dieses Feld erstellt hat. Ausgabe {PresentationSpace} Übergeordnetes Element dieser Feldinstanz.
<code>toString()</code>	Erstellt eine benutzerfreundliche Beschreibung des Felds. Ausgabe {String} Von Benutzern lesbare Darstellung des Felds.

FieldList

Verwenden Sie das FieldList-Objekt gemeinsam mit dem Field-Objekt zum Abrufen von Feldlisteninformationen.

Tabelle 8-11 FieldList

Methode

<code>getPresentationSpace()</code>	Ruft den Darstellungsbereich ab, der diese Feldliste erstellt hat. Ausgabe {PresentationSpace} Übergeordnetes Element dieser Feldliste.
-------------------------------------	--

Methode

`findField(position, text, direction)`

Gibt das Feld mit dem angegebenen Text zurück. Der Suchvorgang beginnt an der angegebenen Position und verläuft vorwärts oder rückwärts. Wenn sich die Zeichenfolge über mehrere Felder erstreckt, wird das Feld zurückgegeben, das die Startposition enthält. Wenn vorwärts gesucht wird, findet kein Umbruch bis zum oberen Ende des Bildschirms statt. Wenn rückwärts gesucht wird, findet kein Umbruch bis zum unteren Ende des Bildschirms statt.

Parameter

{Position} Position, an der die Suche beginnen soll. Weitere Informationen finden Sie unter [Position](#)-Objekt.

{String} Der Text, nach dem gesucht werden soll (optional). Wenn nicht vorhanden, wird das nächste Feld rechts von oder unterhalb der angegebenen Position zurückgegeben.

{Number} Richtung der Suche (optional). Verwenden Sie für diesen Parameter [PresentationSpace.SearchDirection](#)-Konstanten. Zum Beispiel `PresentationSpace.SearchDirection.FORWARD` oder `PresentationSpace.SearchDirection.BACKWARD`. Wenn nicht vorhanden, wird vorwärts gesucht.

Ausgabe

{Field} enthält die Zeichenfolge oder Null, wenn ein Feld, das die vorgegebenen Kriterien erfüllt, nicht gefunden wird.

Ergebnis

{RangeError} Wenn die Position außerhalb des Wertebereichs liegt.

`get(index)`

Ruft das Feld am vorgegebenen Index ab.

Parameter

{Number} Index in die Feldliste.

Ausgabe

{Field} befindet sich am angegebenen Index.

Ergebnis

{RangeError} Wenn der Index außerhalb des Wertebereichs liegt.

`isEmpty()`

Gibt an, ob die Feldliste leer ist.

Ausgabe

{Boolean} 'True', wenn die Liste leer ist.

`size()`

Gibt die Anzahl der in der Liste enthaltenen Felder an.

Ausgabe

{Number} Die Feldanzahl

`toString()`

Erstellt eine benutzerfreundliche Beschreibung der Feldliste.

Ausgabe

{String} Von Benutzern lesbare Darstellung der Feldliste.

OIA

Operatorinformationsfeld-Oberfläche (OIA, Operator Information Area). Das OIA-Objekt gibt Werte zurück, die im [OIAStatus](#)-Objekt definiert sind.

Tabelle 8-12 OIA

Methode	
<code>getStatus ()</code>	Gibt die Gruppe der aktivierten Statuskennzeichen zurück. Weitere Informationen hierzu finden Sie in StatusSet . Parameter Ausgabe {StatusSet} Enthält die Statusinformationen.
<code>getCommErrorCode ()</code>	Gibt den Fehlercode zur aktuellen Kommunikation zurück. Ausgabe {Number} der Fehlercode der aktuellen Kommunikation. Wenn kein Fehlercode vorhanden ist, lautet der Wert 0.
<code>getProgErrorCode ()</code>	Gibt den Fehlercode zum aktuellen Programm zurück. Ausgabe {Number} der Fehlercode des aktuellen Programms. Wenn kein Fehlercode vorhanden ist, lautet der Wert 0.

OIAStatus

Tabelle 8-13 OIAStatus

OIAStatus	Beschreibung
CONTROLLER_READY	Controller bereit
A_LINE	Online mit einer Nicht-SNA-Verbindung
MY_JOB	Verbunden mit einer Hostanwendung
OP_SYS	Verbunden mit einer SSCP-Verbindung (SNA)
UNOWNED	Nicht angeschlossen
TIME	Tastatur gesperrt
SYS_LOCK	Systemsperrung nach AID-Taste
COMM_CHECK	Kommunikationsprüfung
PROG_CHECK	Programmprüfung
ELSEWHERE	Tastenfolge an Cursorposition ungültig
FN_MINUS	Funktion nicht verfügbar
WHAT_KEY	Tastenfolge ungültig

OIAStatus	Beschreibung
MORE_THAN	Zu viele Zeichen wurden im Feld eingegeben
SYM_MINUS	Eingegebenes Symbol nicht verfügbar
INPUT_ERROR	Operator-Eingabefehler (nur 5250)
DO_NOT_ENTER	Nicht eingeben
INSERT	Cursor im Einfügemodus
GR_CURSOR	Cursor im Grafikmodus
COMM_ERR_REM	Erinnerung Kommunikationsfehler
MSG_WAITING	Anzeige für Nachricht vorhanden
ENCRYPT	Sitzung ist verschlüsselt
NUM_FIELD	Ungültiges Zeichen in einem Feld für numerische Daten

AutoSignon

Einige Mainframe-Hosts verfügen über einen Digital Certificate Access Server (DCAS). Sie können für die Anmeldung bei einer Hostanwendung ein temporäres Einmal-Weiterleitungsticket vom DCAS verwenden. Mit diesem Objekt können Sie ein Makro schreiben und konfigurieren, das beim Sitzungsstart ausgeführt werden soll und Sie automatisch mit den Anmeldeinformationen des aktuell angemeldeten Reflection ZFE-Benutzers anmeldet.

Tabelle 8-14 AutoSignon

Methode	
<code>getPassTicket()</code>	<p>Ruft ein Weiterleitungsticket ab, das für die Anmeldung bei einer Mainframe-Anwendung verwendet werden soll. Unter Verwendung verschiedener Anwendungskennungen können mehrere Weiterleitungstickets angefordert werden.</p> <p>Parameter</p> <p>{String} Anwendungskennung, die dem Host mitteilt, zu welcher Anwendung die Anmeldung gehört.</p> <p>Ausgabe</p> <p>{Promise} wird mit der Taste für das Weiterleitungsticket ausgeführt oder abgelehnt, wenn der Vorgang fehlschlägt. Das vom DCAS abgerufene Weiterleitungsticket kann nur mit der aktuellen Hostsitzung verwendet werden und ist für zehn Minuten gültig.</p>

Methode

`sendUserName()`

Wendet den im Weiterleitungsticket enthaltenen Benutzernamen auf das Feld an der aktuellen Cursorposition auf dem aktuellen Hostbildschirm an. Der Benutzername muss vor dem Kennwort gesendet werden. Wenn Sie das Kennwort zuerst senden, wird das Weiterleitungsticket ungültig, und Sie müssen ein neues Ticket anfordern.

Parameter

{String} 'passTicketKey' abgerufen aus 'getPassTicket'

Ausgabe

{Promise} wird ausgeführt, wenn der Benutzername erfolgreich gesendet wurde. Wird abgelehnt, wenn der Vorgang fehlschlägt.

`sendPassword()`

Wendet das im Weiterleitungsticket enthaltene Kennwort auf das Feld an der aktuellen Cursorposition auf dem aktuellen Hostbildschirm an. Der Benutzername muss vor dem Kennwort gesendet werden. Wenn Sie das Kennwort zuerst senden, wird das Weiterleitungsticket ungültig, und Sie müssen ein neues Ticket anfordern.

Parameter

{String} 'passTicketKey' abgerufen aus 'getPassTicket'

Ausgabe

{Promise} wird ausgeführt, wenn das Kennwort erfolgreich gesendet wurde. Wird abgelehnt, wenn der Vorgang fehlschlägt.

Position

Stellt eine Zeile und eine Spalte auf dem Bildschirm dar.

Tabelle 8-15 Position

Methode

`Position(row,col)`

Erstellt eine neue Position-Instanz.

Parameter

{Number} row Koordinaten der Bildschirmzeile

{Number} col Koordinaten der Bildschirmspalte

PresentationSpace

Verwenden Sie das PresentationSpace-Objekt zur Interaktion mit dem Terminalbildschirm. Zu den verfügbaren Interaktionen zählen das Einrichten und Abrufen der Cursorposition, das Senden von Tasten und das Lesen von Text.

Tabelle 8-16 PresentationSpace

METHODEN

<code>getCursorPosition()</code>	<p>Gibt eine Position-Instanz zurück, die die aktuelle Cursorposition darstellt. Eine Sitzung ohne Verbindung weist die Cursorposition 0,0 auf.</p> <p>Ausgabe</p> <p>{Position} aktuelle Cursorposition</p>
<code>setCursorPosition(position)</code>	<p>Bewegt den Hostcursor an die angegebene Zeilen- und Spaltenposition. Bei einigen Hosts wie VT werden die Cursorbewegungen durch den Host eingeschränkt.</p> <p>Parameter</p> <p>{Position} Position neue Cursorposition.</p> <p>Ausgabe</p> <p>Keine</p> <p>Ergebnis</p> <p>{RangeError} Wenn die Position auf dem aktuellen Bildschirm ungültig ist.</p>
<code>isCursorVisible()</code>	<p>Testet, ob der Cursor aktuell im Präsentationsbereich angezeigt wird. Wenn die Sitzung nicht verbunden ist, wird der Cursor als nicht sichtbar betrachtet.</p> <p>Ausgabe</p> <p>{Boolean} 'True', wenn der Cursor angezeigt wird. 'False', wenn der Cursor nicht angezeigt wird.</p>
<code>sendKeys (keys)</code>	<p>Überträgt eine Textzeichenfolge oder ControlKeys an der aktuellen Cursorposition im Präsentationsbereich an den Host. Wenn sich der Cursor nicht an der gewünschten Position befindet, verwenden Sie zunächst die <code>setCursorPosition</code>-Funktion.</p> <p>Die Textzeichenfolge kann eine beliebige Anzahl an Zeichen und ControlKeys enthalten.</p> <p>Ein Beispiel: <code>"myname" + ControlKey.TAB + "mypass" + ControlKey.ENTER</code> überträgt eine Benutzerkennung, tabuliert zum nächsten Feld und überträgt zunächst ein Kennwort und dann die Eingabetaste.</p> <p>Um eine eckige Klammer zu übertragen, geben Sie die Klammern doppelt ein (<code>[[</code> oder <code>]]</code>).</p> <p>Parameter</p> <p>{String} Tastentext und/oder Steuertasten, der/die übertragen werden soll(en).</p>

METHODEN

<code>getText (position, length)</code>	<p>Gibt eine Zeichenfolge zurück, die einen linearen Bereich des Präsentationsbereichs darstellt. Wenn Zeilengrenzen auftreten, werden keine Zeichen für neue Zeilen eingefügt.</p> <p>Parameter</p> <p><code>{Position}</code> Startposition, aus der Text abgerufen wird.</p> <p><code>{Number}</code> Länge der maximalen Anzahl an Zeichen, die zurückgegeben werden sollen. Wenn durch den Längenparameter die letzte Position des Präsentationsbereichs überschritten wird, werden nur die Zeichen bis zur letzten Position zurückgegeben.</p> <p>Ausgabe</p> <p><code>{String}</code> stellt einen linearen Bereich des Präsentationsbereichs dar, der leer sein kann, wenn die Sitzung nicht verbunden ist.</p> <p>Ergebnis</p> <p><code>{RangeError}</code> Wenn die Position oder Länge auf dem aktuellen Bildschirm ungültig ist.</p>
<code>getSize ()</code>	<p>Ruft die Abmessungen des Bildschirms als Dimension-Objekt ab.</p> <p>Ausgabe</p> <p><code>{Dimension}</code> Enthält die Anzahl an Zeilen und Spalten. Die Bildschirmgröße beträgt <code>[row:0, col:0]</code>, wenn die Sitzung nicht verbunden ist.</p>
<code>getDataCells (start, length)</code>	<p>Gibt <code>DataCell</code>-Instanzen zurück, bei denen das erste Mitglied für die durch den Startparameter angegebene Position gedacht ist. Die maximale Anzahl an <code>DataCell</code>-Instanzen in der Liste wird durch den Längenparameter angegeben.</p> <p>Parameter</p> <p><code>{Position}</code> Start der ersten Position auf dem Hostbildschirm, auf dem <code>DataCell</code>-Instanzen abgerufen werden sollen. Weitere Informationen hierzu finden Sie in Position.</p> <p><code>{Number}</code> Länge der maximalen Anzahl an <code>DataCell</code>-Instanzen, die abgerufen werden sollen. Wenn die Länge nicht angegeben ist, werden <code>DataCell</code>-Instanzen von der Startposition bis zum Ende des Bildschirms zurückgegeben.</p> <p>Ausgabe</p> <p><code>{DataCell []}</code>-Instanzen, die leer sein können, wenn die Sitzung nicht verbunden ist. Wenn die Position nicht angegeben ist, werden alle <code>DataCell</code>-Instanzen zurückgegeben. Wenn die Länge nicht angegeben ist, werden <code>DataCell</code>-Instanzen von der Startposition bis zum Ende des Bildschirms zurückgegeben.</p> <p>Ergebnis</p> <p><code>{RangeError}</code> Wenn Start oder Länge außerhalb des Wertebereichs liegt.</p>

METHODEN

`getFields()`

Gibt eine Liste der Felder im Präsentationsbereich zurück. Wenn der Hosttyp keine Felder unterstützt oder der aktuelle Bildschirm nicht formatiert wurde, ist der Rückgabewert immer eine leere Liste. Weitere Informationen hierzu finden Sie in [FieldList](#).

Ausgabe

{`FieldList`} von durch den Host definierten Feldern im Präsentationsbereich.

Session

Bei dem Session-Objekt handelt es sich um den Hauptzugriffspunkt für die Interaktion mit dem Host. Es enthält Funktionen zum Herstellen und Trennen einer Verbindung und zum Abrufen des `PresentationSpace`-Objekts.

Table 8-17 Funktionen des Session-Objekts

METHODEN

`connect()`

Stellt die Verbindung zum konfigurierten Host her. Verwenden Sie ggf. `wait.forConnect()`, um die Makroausführung zu blockieren, bis die Verbindung hergestellt wurde.

Ausgabe

Keine

`disconnect()`

Trennt die Sitzung zum konfigurierten Host. Verwenden Sie ggf. `wait.forDisconnect()`, um die Makroausführung zu blockieren, bis die Verbindung hergestellt wurde.

Ausgabe

Keine

`isConnected()`

Gibt an, ob eine Verbindung zum Host besteht.

Ausgabe

{`Boolean`} 'True', wenn die Hostverbindung hergestellt wurde, 'False', wenn keine Verbindung besteht.

`getPresentationSpace()`

Gewährt Zugriff auf die [PresentationSpace](#)-Instanz für diese Sitzung.

Ausgabe

{`PresentationSpace`} Instanz, die dieser Sitzung zugeordnet ist.

METHODEN

<code>getDeviceName()</code>	<p>Gibt den Namen des verbundenen verfügbaren Geräts, des konfigurierten Geräts oder Null zurück, wenn kein Gerätenamen konfiguriert ist.</p> <p>Der Name des verbundenen Geräts ist der Name, der im Rahmen der Verbindungsverhandlung zwischen dem Host und dem Terminal vereinbart wurde. Es kann sich dabei um den angegebenen oder um einen anderen Namen handeln, beispielsweise wenn ein Pool für Gerätenamen angegeben wurde.</p> <p>Ausgabe</p> <p>{String} Der Name des verbundenen Geräts, der Name des konfigurierten Geräts oder Null.</p>
<code>getType()</code>	<p>Gibt den Typ der Hostsitzung zurück. Weitere Informationen finden Sie unter SessionType.</p> <p>Ausgabe</p> <p>{String} Der Typ der Hostsitzung.</p>
<code>setDeviceName()</code>	<p>Bietet eine Möglichkeit, in einer Sitzungsinstanz den Gerätenamen zu ändern.</p> <p>Parameter</p> <p>{String} name Gerätenamen, der bei der Herstellung einer Verbindung zu einem Host verwendet werden soll.</p> <p>Ergebnis</p> <p>{Error} Wenn während einer hergestellten Sitzung versucht wurde, den Gerätenamen einzurichten.</p>
<code>getOIA()</code>	<p>Bietet Zugriff auf die Operator Information Area (OIA)-Instanz für diese Sitzung.</p> <p>Ausgabe</p> <p>{OIA} Ist dieser Sitzung zugeordnet.</p>

SessionType

Konstanten zum Identifizieren des Hosttyps, zu dem die Verbindung hergestellt wird. Weitere Informationen finden Sie unter [Session-Objekt](#).

Tabelle 8-18 SessionType

Hosttyp	Beschreibung
IBM_3270	Zeigt eine IBM 3270-Terminalsitzung an.
IBM_5250	Zeigt eine IBM 5250-Terminalsitzung an.
VT	Zeigt eine VT-Sitzung an.

StatusSet

Mit dem StatusSet-Objekt können Sie den Status des OIA-Objekts decodieren. Das StatusSet-Objekt gibt die im [OIAStatus](#)-Objekt definierten Werte zurück. Wenn sie gemeinsam verwendet werden, können Sie die Statusinformationen aus dem OIA-Objekt abrufen.

Tabelle 8-19 StatusSet

Methode	
<code>contains(statusFlag)</code>	<p>Legt fest, ob der Satz das angegebene Statuskennzeichen aus OIAStatus-Konstanten enthält.</p> <p>Parameter</p> <p>{Number} statusFlag Zu prüfender Status.</p> <p>Ausgabe</p> <p>{Boolean} 'True', wenn das Statuskennzeichen im Satz enthalten ist.</p>
<code>isEmpty()</code>	<p>Gibt an, ob der Statussatz leer ist.</p> <p>Ausgabe</p> <p>{Boolean} 'True', wenn der Satz leer ist.</p>
<code>size()</code>	<p>Gibt die Anzahl der in dem Satz enthaltenen Statuskennzeichen an.</p> <p>Ausgabe</p> <p>{Number} Die Statusanzahl</p>
<code>toArray()</code>	<p>Konvertiert den internen Statussatz in ein Array.</p> <p>Ausgabe</p> <p>{Object []} Array von Statuskennzeichen in dem Satz.</p>
<code>toString()</code>	<p>Konvertiert den internen Statussatz in eine Zeichenfolge.</p> <p>Ausgabe</p> <p>{String} Durch Leerzeichen eingeschlossene Namen der im Satz enthaltenen Statuskennzeichen.</p>
<code>forEach(callback, thisArg)</code>	<p>Funktion zum Durchlaufen der einzelnen Elemente im Statussatz.</p> <p>Parameter</p> <p>{forEachCallback} Rückruffunktion zum Ausführen eines bestimmten Vorgangs. Wird gemeinsam mit dem Namen des jeweiligen Status im Satz aufgerufen.</p> <p>{Object} 'thisArg' Optionaler Verweis auf ein Kontext-Objekt.</p>

Methode

`forEachCallback(string, thisArg)` Eine durch Benutzer bereitgestellte Rückruffunktion, mit der Sie das Verhalten bereitstellen. Wird als Rückrufparameter für 'forEach' verwendet.

Parameter

`{String} String` Der Name eines Status in dem Statussatz.

`{Object} thisArg` Optionaler Verweis auf ein Kontext-Objekt.

User Interface

Das UI-Objekt stellt Funktionen zur Interaktion mit dem Benutzer sowie zur Abfrage und Anzeige von grundlegenden Informationen bereit. Das UI-Objekt ist in Ihrem Makro automatisch als `ui-Variable` verfügbar.

HINWEIS: Wichtig! Allen UI-Funktionen muss das `yield`-Schlüsselwort voranstehen. Dadurch kann das Makro die Ausführung blockieren, bis die Bedingungen der UI-Funktion erfüllt wurden.

[parameter] Kennzeichnet einen optionalen Parameter.

Tabelle 8-20 Benutzerinteraktion

METHODEN

`prompt (message, [defaultAnswer], [mask])`

Fragt von Benutzern Informationen auf der Benutzeroberfläche ab.

Parameter

`{String}` Meldungstitel, der Benutzern angezeigt werden soll. Standard: Leere Zeichenfolge.

`{String}` Standardantwort, wenn der Benutzer die Zeichenfolge leer lässt. Standard: Leere Zeichenfolge.

`{Boolean}` Maske, die anzeigt, ob die Aufforderung ausgeblendet werden soll (wie bei einem Kennwort).

Ausgabe

`{Promise}` Wird ausgeführt, wenn der Benutzer das Dialogfenster schließt. Bei „OK“ wird eine Benutzereingabe zurückgegeben, „Abbrechen“ ergibt Null.

`message ([message])`

Zeigt eine Meldung auf der Benutzeroberfläche an.

Parameter

`{String}` Meldung, die dem Benutzer angezeigt werden soll. Standard: Leere Zeichenfolge.

Ausgabe

`{Promise}` Wird ausgeführt, wenn der Benutzer das Meldungsfenster schließt.

Wait

Verwenden Sie das wait-Objekt, um auf einen bestimmten Sitzungs- oder Bildschirmstatus zu warten. Sie können beispielsweise darauf warten, dass der Cursor an einer bestimmten Position gefunden wird oder Text an einer bestimmten Position vorhanden ist, bevor Sie mit dem Ausführen des Makros fortfahren.

Wait-Funktionen werden häufig zusammen mit asynchronen Funktionen wie connect() oder sendKeys() verwendet.

HINWEIS: Alle Funktionen verwenden Wartezeiten als optionale Parameter und verfügen über einen Wartezeitwert von 10 Sekunden (10000 ms).

Wichtig: Allen wait-Funktionen muss das yield-Schlüsselwort voranstehen. Dadurch kann das Makro die Ausführung blockieren, bis die Bedingungen der wait-Funktion erfüllt wurden.

[parameter] Kennzeichnet einen optionalen Parameter.

Table 8-21 Auf den Host warten

METHODEN

<code>setDefaultTimeout (timeout)</code>	Legt die Standardwartezeit für alle Funktionen fest.
	Parameter
	{Number} Standardwartezeit für alle wait-Funktionen in Millisekunden.
	Ausgabe
	Keine
	Ergebnis
	{RangeError} Wenn die angegebene Wartezeit kleiner als null ist.
<code>forConnect ([timeout])</code>	Wartet auf das Abschließen einer Verbindungsanforderung.
	Parameter
	{Number} in Millisekunden.
	Ausgabe
	{Promise} Wird ausgeführt, wenn die Sitzung bereits verbunden wurde oder die Verbindung zustande kommt. Wird abgelehnt, wenn die Wartezeit überschritten wurde.
<code>forDisconnect ([timeout])</code>	Wartet auf das Abschließen einer Anforderung zur Trennung einer Verbindung.
	Parameter
	{Number} Wartezeit in Millisekunden.
	Ausgabe
	{Promise} Wird ausgeführt, wenn die Verbindung der Sitzung bereits getrennt wurde oder endgültig getrennt wird. Wird abgelehnt, wenn die Wartezeit überschritten wurde.

METHODEN

<code>forFixedTime([timeout])</code>	<p>Wartet ohne Bedingungen für eine festgelegte Zeit. Die Zeit wird in Millisekunden (ms) angegeben.</p> <p>Parameter</p> <p>{Number} Wartezeit in Millisekunden.</p> <p>Ausgabe</p> <p>{Promise} Wird nach dem Verstreichen der Zeit ausgeführt.</p>
<code>forCursor(position, [timeout])</code>	<p>Wartet darauf, dass der Cursor die angegebene Position erreicht.</p> <p>Parameter</p> <p>{Position} Die Position, die die Zeile und die Spalte angibt,</p> <p>{Number} Wartezeit in Millisekunden.</p> <p>Ausgabe</p> <p>{Promise} Wird ausgeführt, wenn der Cursor die Position bereits erreicht hat oder sie endgültig erreicht. Wird abgelehnt, wenn die Wartezeit überschritten wurde.</p>
<code>forText(string, position, [timeout])</code>	<p>Wartet darauf, dass sich der Text an einer bestimmten Position auf dem Bildschirm befindet.</p> <p>Parameter</p> <p>{String} Der zu erwartende Text.</p> <p>{Position} Die Position, die die Zeile und die Spalte angibt.</p> <p>{Number} Wartezeit in Millisekunden.</p> <p>Ausgabe</p> <p>{Promise} Wird ausgeführt, wenn sich der Text bereits an der angegebenen Position befand oder auf der Position angezeigt wird. Wird abgelehnt, wenn die Wartezeit überschritten wurde.</p> <p>Ergebnis</p> <p>{rangeError} Wenn die Position ungültig ist.</p>

METHODEN

```
forHostPrompt (string,  
column, [timeout])
```

Wartet auf eine Eingabeaufforderung in einer bestimmten Spalte auf dem Bildschirm.

Parameter

{String} Die zu erwartende Texteingabeaufforderung.

{Number} Die Spalte, an der der Cursor erwartet wird.

{Number} Wartezeit in Millisekunden.

Ausgabe

{Promise} Wird ausgeführt, wenn die Bedingungen bereits erfüllt wurden oder endgültig erfüllt werden. Wird abgelehnt, wenn die Wartezeit überschritten wurde.

Ergebnis

{rangeError} Wenn die Spalte außerhalb des Wertebereichs liegt.

Beispielmakros

Die nachstehenden Beispiele eignen sich als Ausgangspunkt für die Erstellung erfolgreicher Makros, in denen die Funktionen des Makroeditors und Reflection ZFE ideal genutzt werden.

- ♦ „Grundlegende Hostinteraktion“, auf Seite 91
- ♦ „Benutzerinteraktion“, auf Seite 93
- ♦ „Durchlaufen von Daten“, auf Seite 93
- ♦ „Aufrufen eines Webdienstes“, auf Seite 94
- ♦ „Arbeiten mit Datenzellen und Attributen“, auf Seite 95
- ♦ „Verwenden von Feldern und Feldlisten“, auf Seite 96
- ♦ „Makro für die automatische Anmeldung für Mainframes“, auf Seite 97

Grundlegende Hostinteraktion

In diesem Beispiel wird die grundlegende Hostinteraktion dargestellt. Dazu zählen die folgenden Interaktionen:

- ♦ Daten an den Host senden
 - ♦ Auf die Anzeige von Bildschirmen warten
 - ♦ Das `yield`-Schlüsselwort verwenden, um auf asynchrone Funktionen zu warten
 - ♦ Text auf dem Bildschirm lesen
 - ♦ Dem Benutzer grundlegende Informationen anzeigen
 - ♦ Fehlergrundlagen behandeln
1. **Session** – Hauptzugriffspunkt zum Host. Kann Verbindungen herstellen und trennen und bietet Zugriff auf den Präsentationsbereich.

Das aus der Sitzung abgerufene PresentationSpace-Objekt stellt den Bildschirm dar und bietet zahlreiche allgemeine Funktionen wie das Abrufen und Einrichten der Cursorposition, das Senden von Daten an den Host und das Lesen auf dem Bildschirm.

2. **wait** – Dieses Objekt bietet eine einfache Möglichkeit, auf das Auftreten der verschiedenen Hoststatus zu warten, bevor weitere Daten gesendet oder auf dem Bildschirm gelesen werden.
3. **UI** – Stellt die grundlegenden Funktionen der Benutzeroberfläche bereit. Zeigt Benutzern Daten an oder fragt Informationen von Benutzern ab.

```
// Neue Makrofunktion erstellen var macro = createMacro(function*(){ 'use strict';
// Für alle Makros sind standardmäßig die folgenden Objekte verfügbar: // 1.
session - Hauptzugriffspunkt zum Host. Kann Verbindungen herstellen und trennen und
bietet Zugriff auf das PresentationSpace-Objekt. // Das aus der Sitzung abgerufene
PresentationSpace-Objekt stellt den Bildschirm dar und bietet zahlreiche allgemeine
Funktionen wie das Abrufen und Einrichten der // Cursorposition, das Senden von
Daten an den Host und das Lesen auf dem Bildschirm. // 2. wait - Dieses Objekt
bietet eine einfache Möglichkeit, auf das Auftreten der verschiedenen Hoststatus zu
warten, bevor weitere Daten gesendet oder auf dem Bildschirm gelesen werden. // 3.
ui - Stellt die grundlegenden Funktionen der Benutzeroberfläche bereit. Benutzern
Daten anzeigen oder Informationen von Benutzern abfragen. // Eine Variable zum
Lesen und Anzeigen von Bildschirmdaten deklarieren. // Es wird empfohlen, alle
Variablen im oberen Bereich einer Funktion zu deklarieren. var numberOfAccounts =
0; // Mit dem Abrufen des PresentationSpace-Objekts beginnen, das zahlreiche
gängige Bildschirmoperationen bereitstellt. var ps =
session.getPresentationSpace(); try { // Kann die Cursorposition einrichten und
abrufen ps.setCursorPosition(new Position(24, 2)); // Die sendKeys-Funktion zum
Senden von Zeichen an den Host verwenden ps.sendKeys('cics'); // SendKeys wird auch
zum Senden von Hosttasten wie PA- und PF-Tasten verwendet. // Siehe "Steuertasten"
in der Dokumentation für alle verfügbaren Optionen ps.sendKeys(ControlKey.ENTER); /
/ Darauf warten, dass der Cursor die korrekte Position erreicht. // Das wait-Objekt
bietet verschiedene Funktionen zum Warten auf das Eintreten bestimmter Status, //
sodass Sie weitere Tasten senden oder Daten auf dem Bildschirm lesen können. yield
wait.forCursor(new Position(24, 2)); // Sie können Zeichen und Steuerungstasten
gemeinsam in einem sendKeys-Aufruf verwenden. ps.sendKeys('data' + ControlKey.TAB +
ControlKey.TAB + 'more data' + ControlKey.ENTER); // Das "yield"-Schlüsselwort muss
allen "wait"- und "ui"-Funktionsaufrufen voranstehen. // Es weist den Browser an,
die Ausführung des Makros anzuhalten, bis die // (asynchrone) wait-Funktion
zurückgegeben wird. Informationen darüber, welche Funktionen das yield-Kennwort //
erfordern, finden Sie in der Dokumentation. yield wait.forCursor(new Position(10,
26)); ps.sendKeys('accounts' + ControlKey.ENTER); // Kann auch darauf warten, dass
in bestimmten Bereichen des Bildschirms Text angezeigt wird yield
wait.forText('ACCOUNTS', new Position(3, 36)) ; ps.sendKeys('1' +
ControlKey.ENTER); // Alle wait-Funktionen werden unterbrochen, wenn die Kriterien
nicht innerhalb einer bestimmten Zeitspanne erfüllt werden. // Kann mit einem
optionalen Parameter in den wait-Funktionen die Wartezeit erhöhen (in
Millisekunden) // Alle Wartezeiten werden in Millisekunden angegeben. Der
Standardwert ist 10 Sekunden (10000 ms). yield wait.forCursor(new Position(1, 1),
15000); ps.sendKeys('A' + ControlKey.ENTER); // PS bietet die getText-Funktion zum
Lesen von Text auf dem Bildschirm numberOfAccounts = ps.getText(new Position(12,
3), 5); // Das ui-Objekt verwenden, um Daten auf dem Bildschirm anzuzeigen
ui.message('Number of active accounts: ' + numberOfAccounts); // Mit try / catch
werden alle Fehler erfasst und an einer zentralen Position gemeldet } catch (error)
{ // Wir verwenden wieder das ui-Objekt zum Anzeigen einer Fehlermeldung yield
ui.message('Error: ' + error.message); } //Ende generiertes Makro}); // Das Makro
ausführen und die Ergebnisse an Macro Runner zurückgeben // Die Rückgabe-Anweisung
ist erforderlich, da sie von der ZFE-Anwendung verwendet wird, um zu wissen, // ob
das Makro erfolgreich ausgeführt und wann es beendet wurde return macro();
```

Benutzerinteraktion

Dieses Beispiel zeigt, wie Benutzer mithilfe der bereitgestellten API-Methoden zur Eingabe aufgefordert oder über eine Meldung benachrichtigt werden.

```
var macro = createMacro(function*(){ 'use strict'; // Das "ui"-Objekt stellt
Funktionen zur Abfrage und Anzeige von Informationen bereit // Variablen für eine
spätere Verwendung deklarieren var username; var password; var flavor; var scoops;
//Start generiertes Makro var ps = session.getPresentationSpace(); try { //
Benutzer zur Eingabe ihres Namens auffordern und den Namen in einer Variable
speichern. // Das 'yield'-Schlüsselwort ist zum Blockieren der Ausführung
erforderlich, während auf die Benutzereingabe gewartet wird. username = yield
ui.prompt('Geben Sie Ihr Kennwort ein'); // Benutzer mit bereitgestelltem Standard
zur Eingabe eines Werts auffordern. flavor = yield ui.prompt('What is your favorite
flavor of ice cream?', 'Chocolate'); // Benutzer über die 'mask'-Option zur Eingabe
persönlicher Informationen auffordern. Das Eingabefeld wird bei der Eingabe
maskiert. // Wenn ein Parameter nicht verwendet wird, kann mit 'null' angegeben
werden, dass er nicht verwendet werden soll. // Hier zeigen wir durch die Angabe,
dass wir keinen Standardwert zeigen müssen. password = yield ui.prompt('Please
enter your password', null, true); // Die Aufforderung gibt null zurück, wenn der
Benutzer nicht auf die Schaltfläche 'OK' klickt, sondern auf 'Abbrechen'. // Eine
Möglichkeit zum Behandeln dieses Falls ist das Umbrechen des Aufrufs in einem try/
catch-Block. scoops = yield ui.prompt('How many scoops would you like?'); if
(scoops === null) { // Dadurch wird das Makro beendet. return; // Alternativ könnte
ein Fehler ausgegeben und im nachstehenden "Catch" erfasst werden } // Die
gesammelten Werte verwenden, um das Eis zu bestellen ps.sendKeys(username +
ControlKey.TAB + password + ControlKey.ENTER); yield wait.forCursor(new Position(5,
1)); ps.sendKeys(flavor + ControlKey.TAB + scoops + ControlKey.ENTER); // Dem
Benutzer eine Meldung anzeigen. Durch die Verwendung des 'yield'-Schlüsselworts vor
dem Aufruf wird die // weitere Ausführung des Makros blockiert, bis der Benutzer
auf die Schaltfläche 'OK' klickt. yield ui.message('Order successful. Enjoy your '
+ scoops + ' scoops of ' + flavor + ' ice cream ' + username + '!'); } catch (error)
{ // Hier verwenden wir das ui-Objekt zum Anzeigen einer Fehlermeldung yield
ui.message(error.message); } //Ende generiertes Makro}); return macro();
```

Durchlaufen von Daten

Dieses Beispiel zeigt, wie eine beliebige Anzahl an Bildschirmen durchlaufen wird und die Daten auf den jeweiligen Bildschirmen verarbeitet werden.

```
// Neue Makrofunktion erstellen. var macro = createMacro(function*(){ 'use strict'; //
Variable(n) für eine spätere Verwendung erstellen var password; var accountNumber; var
transactionCount = 0; var row = 0; // Eine Referenz zum PresentationSpace-Objekt abrufen. var
ps = session.getPresentationSpace(); try { // Für die Anmeldung bei der Anwendung
Benutzername und Kennwort eingeben. yield wait.forCursor(new Position(19, 48));
ps.sendKeys('bjones' + ControlKey.TAB); yield wait.forCursor(new Position(20, 48)); password
= yield ui.prompt('Password:', null, true); ps.sendKeys(password);
ps.sendKeys(ControlKey.ENTER); // Anwendungsbefehl eingeben. yield wait.forCursor(new
Position(20, 38)); ps.sendKeys('4'); ps.sendKeys(ControlKey.ENTER); // Transaktionen für ein
Konto werden aufgelistet. yield wait.forCursor(new Position(13, 25)); ps.sendKeys('2'); //
Kontonummer eingeben. Hier zur Erleichterung hartcodiert. yield wait.forCursor(new
Position(15, 25)); accountNumber = yield ui.prompt('Account Number:', '167439459');
ps.sendKeys(accountNumber); ps.sendKeys(ControlKey.ENTER); // Warten, bis auf dem
Kontoprofilbildschirm vorhanden yield wait.forText('ACCOUNT PROFILE', new Position(3, 33));
// Nach Text suchen, der anzeigt, dass die letzte Seite des Datensatzes erreicht wurde while
(ps.getText(new Position(22, 12), 9) !== 'LAST PAGE') { // Während die letzte Seite mit
Datensätzen nicht erreicht wurde, zur nächsten Seite mit Datensätzen wechseln.
ps.sendKeys(ControlKey.PF2); yield wait.forCursor(new Position(1, 1)); // Wenn die
Cursorposition nicht zwischen Datensatzbildschirmen wechselt und der Bildschirm keinen Text
enthält, // können Sie prüfen, ob ein Bildschirm aktualisiert wurde. Sie können eine //
festgelegte Zeitspanne warten, bis eine aid-Taste für den Aufbau des Bildschirms gesendet
wird. // Zum Beispiel: // yield wait.forFixedTime(1000); // Für alle Zeilen die Zählvariable
erhöhen, wenn sie Daten enthalten. for (row = 5; row <= 21; row++) { // Es befinden sich zwei
```

```

Spalten auf dem Bildschirm. Daten in Spalte 1 prüfen. // In diesem Beispiel wissen wir: Wenn
sich an einer bestimmten // Position ein Leerzeichen befindet, liegt eine Transaktion vor. if
(ps.getText(new Position(row, 8), 1) !== ' ') { transactionCount++; } // Daten in Spalte 2
prüfen. if (ps.getText(new Position(row, 49), 1) !== ' ') { transactionCount++; } } //
Daten in Spalte 2 prüfen. if (ps.getText(new Position(row, 49), 1) !== ' ') {
transactionCount++; } } // Nach dem Durchlaufen aller Datensatzseiten die Anzahl der
Datensätze in einem Meldefenster anzeigen. yield ui.message('There are ' + transactionCount
+ ' records found for account ' + accountNumber + '.'); // Bei der Anwendung abmelden
ps.sendKeys(ControlKey.PF13); ps.sendKeys(ControlKey.PF12); // Mit try / catch werden alle
Fehler erfasst und an einer zentralen Position gemeldet } catch (error) { // Hier verwenden
wir das ui-Objekt zum Anzeigen einer Fehlermeldung yield ui.message(error.message); } }); //
Hier führen wir das Makro aus und geben die Ergebnisse an Macro Runner zurück // Die
Rückgabe-Anweisung ist erforderlich, da sie von der ZFE-Anwendung verwendet wird, um zu
wissen, // ob das Makro erfolgreich ausgeführt wurde return macro();

```

Aufrufen eines Webdienstes

Dieses Beispiel zeigt, wie direkt über ein Makro ein AJAX / REST-Aufruf an einen Webdienst ausgeführt wird. Sie können Daten über Ihre Hostanwendung in den Aufruf des Webdienstes oder umgekehrt über den Webdienst in Ihre Hostanwendung integrieren.

In diesem Beispiel wird der 'Verastream Host Integrator (VHI) CICSAcctsDemo REST'-Dienst aufgerufen. Sie können den Code natürlich auch problemlos anpassen und einen anderen Webdienst aufrufen. Sie sind nicht an den VHI-Dienst gebunden.

In diesem Beispiel wird der Aufruf über einen im Sitzungsserver konfigurierten Proxy durchgeführt (wie unten beschrieben), um „Same-Origin-Policy“-Komplikationen zu vermeiden. Wenn Sie einen Webdienst verwenden, der die [Cross-origin Resource Sharing \(CORS, Ressourcenfreigabe zwischen verschiedenen Ursprüngen\)](#) unterstützt, und wenn Sie einen modernen Browser verwenden, ist der Proxy nicht erforderlich.

Die jQuery-Bibliothek ist in Makros verfügbar, sie können zum Aufrufen von REST-Diensten also direkt die \$.post()-Funktion verwenden.

In dem Beispiel wird außerdem beschrieben, wie ein jQuery REST-Aufruf in einem neuen Promise-Muster umbrochen wird. Das über die nachstehende benutzerdefinierte Funktion zurückgegebene promise-Muster ermöglicht die Verwendung von „yield“ im Hauptmakrocode. Dadurch kann die Hauptmakroausführung warten, bis der Dienstaufruf abgeschlossen ist, bevor sie fortgesetzt wird.

```

var macro = createMacro(function*() { 'use strict'; // Einige Variablen für eine spätere
Verwendung erstellen; var username; var password; var accountNumber; var accountDetails; //
Eine Funktion erstellen, die einen AJAX / REST-Aufruf an einen VHI-Webdienst ausführt. //
Kann für den Aufruf eines beliebigen Webdienstes (nicht nur VHI) angepasst werden. // Wenn
nicht CORS verwendet wird, muss die Anforderung wahrscheinlich einen // Proxy auf dem
Sitzungsserver durchlaufen. Siehe Beispielhinweise für weitere Informationen. /** *
Hartcodierte Unterstützungsfunktion zum Einschließen von AJAX / REST-Parametern, zum
Aktivieren des * REST-Dienstes und zum Zurückgeben der Ergebnisse in einem Promise-Objekt. *
@param {Number} Kontonummer zum Senden an die REST-Abfrage. * @param {String} Benutzername
für den Zugriff auf den REST-Dienst. * @param {String} Kennwort für den Zugriff auf den REST-
Dienst. * @return {Promise} enthält $.post()-Ergebnisse, die mit yield kompatibel sind. */
var getAccountDetails = function (acctNum, username, password) { var url = "proxy1/model/
CICSAcctsDemo/GetAccountDetail"; var args = {"filters": {"AcctNum": acctNum}, "envVars":
{"Username": username, "Password": password}}; // jQuery AJAX / HTTP POST-Aufruf in einem
neuen Promise-Objekt umbrechen. // Das hier zurückgegebene Promise-Objekt ermöglicht dem
Makro über yield / wait // auf seinen Abschluss zu warten. return Promise.resolve($.post(url,
JSON.stringify(args))).catch(function (error) { // Fehler zuordnen, die im jQuery-Aufruf an
unser Promise-Objekt auftreten. throw new Error('REST API Error: ' + error.statusText); });
}; // Start generiertes Makro var ps = session.getPresentationSpace(); try { // Konnte hier

```

```

mit dem Host interagieren, sich bei einer Hostanwendung anmelden usw. // Benutzername und
Kennwort erfassen username = yield ui.prompt('Username:'); password = yield
ui.prompt('Password:', null, true); accountNumber = yield ui.prompt('Account Number:'); if
(!username || !password || !accountNumber) { throw new Error('Username or password not
specified'); } // Externen REST-Dienst aktivieren, und yields / wartet auf den Abschluss des
Aufrufs. accountDetails = yield getAccountDetails(accountNumber, username, password); //
Jetzt haben wir die Daten von unserem externen Dienst. // Können die Daten in unsere lokale
Hostanwendung integrieren oder die Daten einfach dem Benutzer anzeigen. // In diesem Beispiel
zeigen wir einfach die sich ergebenden Kontodetails an. if (accountDetails.result &&
accountDetails.result.length > 0) { yield ui.message(accountDetails.result[0].FirstName + '
$' + accountDetails.result[0].AcctBalance); } else { yield ui.message('No records found for
account: ' + accountNumber); } } catch (error) { // Wenn während des AJAX / REST-Aufrufs //
oder beim Erfassen von Benutzername/Kennwort ein Fehler auftritt, befinden wir uns hier.
yield ui.message(error.message); } }); // Unser Makro ausführen return macro();

```

Proxy-Unterstützung für die Skripterstellung aus verschiedenen Ursprüngen

Wenn Ihre Webdienste CORS nicht unterstützen, schlagen AJAX/REST-Aufrufe fehl, wenn sie versuchen, auf einen nicht aus der ZFE-Anwendung stammenden Server zuzugreifen. Dabei handelt es sich um eine Sicherheitsfunktion für Browser.

Der Reflection ZFE-Server bietet die Möglichkeit, explizit zu vertrauenswürdigen Remoteservern weiterzuleiten.

- ♦ Öffnen Sie in Ihrer ZFE-Umgebung die Datei
`\ReflectionZFE\sessionserver\webapps\zfe\WEB-INF\web.xml`.
- ♦ Ändern Sie die Datei wie folgt:

```

<!-- Beispiel für die Weiterleitung von Fremddiensten, die CORS nicht unterstützen. Mit
der folgenden Konfiguration können Sie VHI REST-Remotedienste über lokale URLs
aufgerufen werden (Beispiel: Ein HTTP POST-Aufruf an "proxyl/model/CICSAcctsDemo/
GetAccountDetail" würde weitergeleitet an "http://remote-vhi-server:9680/vhi-rs/model/
CICSAcctsDemo/GetAccountDetail"). --> <servlet> <servlet-name>vhi-rs-proxyl</servlet-
name> <servlet-class>org.eclipse.jetty.proxy.ProxyServlet$Transparent</servlet-class>
<init-param> <param-name>proxyTo</param-name> <param-value>http://remote-vhi-
server:9680/vhi-rs/</param-value> </init-param> <init-param> <param-name>prefix</param-
name> <param-value>/proxyl</param-value> </init-param> <async-supported>true</async-
supported> </servlet> <servlet-mapping> <servlet-name>vhi-rs-proxyl</servlet-name>
<url-pattern>/proxyl/*</url-pattern> </servlet-mapping>

```

- ♦ Kommentieren Sie die Bereiche 'servlet' und 'servlet-mappings' aus.
- ♦ Ändern Sie `http://remote-vhi-server:9680/vhi-rs` in den tatsächlichen URL des gewünschten REST-Servers. Sie können auch das URL-Muster umbenennen.
- ♦ Bitte beachten Sie: Ein REST-Server unterstützt zwar CORS-Titel, ältere Browser tun dies jedoch nicht. Daher kann dieses Beispiel weiterhin relevant sein.

TIPP: Ihre angepasste `web.xml`-Datei kann jederzeit ersetzt werden, wenn Sie Reflection ZFE neu bereitstellen. Denken Sie daran, immer Ihre Dateien zu sichern.

Arbeiten mit Datenzellen und Attributen

Dieses Makro veranschaulicht, wie Sie Datenzellen und Attributsätze zum Überprüfen einer bestimmten Zeile/Spalte auf dem Bildschirm für Text und Attribute verwenden können. In diesem Beispiel sehen Sie Folgendes:

- ♦ Wie Sie eine Sammlung von DataCells-Objekten für eine bestimmte Position und Länge abrufen.
- ♦ Wie Sie DataCells-Objekte zum Erstellen einer Textzeichenfolge durchlaufen.

- ♦ Wie Sie zum Vergleich in ähnlicher Weise auch `getText()` verwenden können.
- ♦ Wie Sie mit Attributen arbeiten, eine Auflistung mit Zeichenfolgen abrufen oder feststellen, ob bestimmte Attribute an einer festgelegten Bildschirmposition angegeben sind.

```
var macro = createMacro(function*() { 'use strict'; // Präsentationsbereich für die
Interaktion mit dem Host abrufen var ps = session.getPresentationSpace(); // Variablen für
eine spätere Verwendung deklarieren var cells; var text; var attrs; // Standardwartezeit für
"wait"-Funktionen festlegen wait.setDefaultTimeout(10000); // Beispielmakro für das Arbeiten
mit Datenzellen und Attributen try { yield wait.forCursor(new Position(24, 2)); //
Datenzellen aus dem Präsentationsbereich abrufen // Zeile 19, Spalte 3 ist die
Eingabeaufforderung 35 Zeichen lang // "Wählen Sie einen der folgenden Befehle:" cells =
ps.getDataCells({row:19, col:3}, 35); text = ''; // Sie können Text mithilfe von 'getText'
anzeigen yield ui.message("Screen text: " + ps.getText({row:19, col:3}, 35)); // Oder den
Text aus den Datenzellen an den einzelnen Positionen bilden for(var index = 0; index <
cells.length; index++) { text = text.concat(cells[index].getChar()); } // Und den Text
anzeigen yield ui.message("Cells text: " + text); // Attribute für die erste Datenzelle
abrufen (cell[0]) attrs = cells[0].getAttributes(); // Anzeigen, ob Attribute für die
Datenzelle vorhanden sind yield ui.message("Attribute set is empty: " + attrs.isEmpty()); //
Anzeigen, wie viele Attribute angegeben sind yield ui.message("Number of attributes: " +
attrs.size()); // Anzeigen, welche Attribute angegeben sind yield ui.message("Attributes: "
+ attrs.toString()); // Anzeigen, ob das Attribut 'HIGH_INTENSITY' angegeben ist yield
ui.message("Is high intensity: " + attrs.contains(Attribute.HIGH_INTENSITY)); // Anzeigen,
ob das Attribut 'UNDERLINE' angegeben ist yield ui.message("Is underline: " +
attrs.contains(Attribute.UNDERLINE)); // Anzeigen, ob die Attribute 'ALPHA_NUMERIC',
'HIGH_INTENSITY' und 'PEN_DETECTABLE' angegeben sind yield ui.message("Is alphanumeric,
intensified and pen-detectable: " + attrs.containsAll([Attribute.ALPHA_NUMERIC,
Attribute.HIGH_INTENSITY, Attribute.PEN_DETECTABLE])); // Anzeigen, ob die Attribute
'UNDERLINE', 'HIGH_INTENSITY' und 'PEN_DETECTABLE' angegeben sind yield ui.message("Is
underline, intensified and pen-detectable: " + attrs.containsAll([Attribute.UNDERLINE,
Attribute.HIGH_INTENSITY, Attribute.PEN_DETECTABLE])); } catch (error) { yield
ui.message(error); } //Generiertes Makro beenden }); // Zurückgegebenes Makro ausführen
macro();
```

Verwenden von Feldern und Feldlisten

Dieses Makrobeispiel veranschaulicht die Verwendung allgemeiner Funktionen für die Interaktion mit den Feldern in der Makro-API. Es wird beispielsweise dargestellt, wie Feldtext abgerufen wird, Feldinformationen angezeigt werden und wie `field.setText` als Alternative zu `sendKeys` für die Interaktion mit dem Host verwendet werden kann.

HINWEIS: Aufgrund bestimmter Browseraspekte reduziert `ui.message` mehrere aufeinander folgende Leerzeichen zu einem einzelnen Leerzeichen. Die Leerzeichen werden im JavaScript-Code beibehalten.

```

var macro = createMacro(function*() { 'use strict'; // Präsentationsbereich für die
Interaktion mit dem Host abrufen var ps = session.getPresentationSpace(); // Variablen für
eine spätere Verwendung deklarieren var fields; var field; var searchString = 'z/vM'; //
Standardwartezeit für "wait"-Funktionen festlegen wait.setDefaultTimeout(10000); //
Beispielmakro für das Arbeiten mit Feldlisten und Feldern try { yield wait.forCursor(new
Position(24, 2)); // Feldliste abrufen. fields = ps.getFields(); // Die gesamte Feldliste
durchlaufen und Feldinfo anzeigen. for(var index = 0; index < fields.size(); index++) { field
= fields.get(index); yield ui.message("Field " + index + " info: " + field.toString()); }
yield ui.message("Now, find a field containing the text '" + searchString + "'"); field =
fields.findField(new Position(1, 1), searchString); if(field !== null) { yield
ui.message("Found field info: " + field.toString()); yield ui.message("Found field
foreground is green? " + (Color.GREEN === field.getForegroundColor())); yield
ui.message("Found field background is default? " + (Color.BLANK_UNSPECIFIED ===
field.getBackgroundColor())); } // Jetzt nach einem Befehlsfeld suchen und es ändern. field
= fields.findField(new Position(23, 80)); if(field !== null) { field.setText("cics"); }
yield ui.message("Click to send 'cics' to host."); ps.sendKeys(ControlKey.ENTER); // Auf
neuen Bildschirm warten; neue Felder abrufen. yield wait.forCursor(new Position(10, 26));
fields = ps.getFields(); // Benutzerfeld suchen und festlegen. field = fields.findField(new
Position(10, 24)); if(field !== null) { field.setText("bvtst01"); } // Kennwortfeld suchen
und festlegen. field = fields.findField(new Position(11, 24)); if(field !== null) {
field.setText("milk"); } yield ui.message("Click to send login to host.");
ps.sendKeys(ControlKey.ENTER); // Auf neuen Bildschirm warten; neue Felder abrufen. yield
wait.forCursor(new Position(1, 1)); fields = ps.getFields(); // Befehlsfeld suchen und
'logoff'-Befehl festlegen. field = fields.findField(new Position(24, 45)); if(field !==
null) { field.setText("cesf logoff"); } yield ui.message("Click to send logoff to host.");
ps.sendKeys(ControlKey.ENTER); } catch (error) { yield ui.message(error); } //Ende
generiertes Makro}); // Makro ausführen return macro();

```

Makro für die automatische Anmeldung für Mainframes

In diesem Beispiel wird mit dem AutoSignon-Objekt ein Makro erstellt, das die einem Benutzer zugeordneten Berechtigungsnachweise verwendet, um ein Weiterleitungsticket vom Digital Certificate Access Server (DCAS) abzurufen.

```

var macro = createMacro(function*() { 'use strict'; // Präsentationsbereich für die
Interaktion mit dem Host abrufen var ps = session.getPresentationSpace(); // Variable für
Weiterleitungsticket für die Anmeldung var passTicket; // Anwendungs-ID für die Anmeldung
var appId = 'CICSV41A'; // Standardwartezeit für "wait"-Funktionen festlegen
wait.setDefaultTimeout(10000); // Start generiertes Makro try { yield wait.forCursor(new
Position(24, 2)); // Weiterleitungsticket von DCAS abrufen. passTicket = yield
autoSignon.getPassTicket(appId); ps.sendKeys('cics'); ps.sendKeys(ControlKey.ENTER); yield
wait.forCursor(new Position(10, 26)); // Generierten Benutzernamen mit
'sendUserName(passTicket) ...' ersetzen yield autoSignon.sendUserName(passTicket); //
ps.sendKeys('bvtst01' + ControlKey.TAB + ControlKey.TAB); ps.sendKeys(ControlKey.TAB +
ControlKey.TAB); yield wait.forCursor(new Position(11, 26)); // Generiertes Kennwort mit
'sendPassword(passTicket) ...' ersetzen yield autoSignon.sendPassword(passTicket); // var
userInput3 = yield ui.prompt('Password:', '', true); // if (userInput3 === null) { // throw
new Error('Password not provided'); // } // ps.sendKeys(userInput3);
ps.sendKeys(ControlKey.ENTER); yield wait.forCursor(new Position(1, 1)); yield
ui.message('Logged in. Log me off. '); ps.sendKeys('cesf logoff');
ps.sendKeys(ControlKey.ENTER); } catch (error) { yield ui.message(error); } //Ende
generiertes Makro }); // Makro ausführen return macro();

```

9 Datei- und Datenübertragung

Bei der Mainframe-Dateiübertragung werden mithilfe des IND\$FILE-Hostprogramms Dateien zwischen Ihrem Computer und einem 3270-Mainframe kopiert. Als Administrator können Sie angeben, ob Dateien über ein TSO- oder ein CMS-Hostdateisystem übertragen werden.

Mainframe-Dateiübertragung

Bevor Sie Dateien übertragen können, müssen Sie (als Administrator) die Übertragungsoption für die aktuelle Sitzung aktivieren. Dies ist im Bereich „Verbindung“ möglich.


Wählen Sie in der Dropdownliste **Hostdateisystem** aus, welche IBM 3270-Betriebsumgebung der Host ausführt. ZFE unterstützt TSO (Time Sharing Option) und CMS (Conversational Monitor System). Die Standardeinstellung ist „Keine“.

Es werden Übertragungen im ASCII- oder Binär-Modus unterstützt.

Übertragen von Dateien

Sie müssen mit dem Host verbunden sein, um Dateien für die aktuelle 3270-Sitzung übertragen zu können.

- 1 Stellen Sie sicher, dass sich der Host in einem betriebsbereiten Status befindet, um den IND\$FILE-Befehl akzeptieren zu können.

- 2 Klicken Sie im linken Bereich auf das Symbol **IND\$FILE** .

- 3 Das Dialogfeld „Dateiübertragung“ wird angezeigt, das eine Liste von Hostdateien und -verzeichnissen enthält, die übertragen werden können. Verzeichnisse und Dateien werden bei der Auswahl der Datei durch ein entsprechendes Symbol gekennzeichnet.

- 4 Wählen Sie die Übertragungsmethode aus: Die Optionen sind:

- ♦ Binär

Für Programmdateien und andere Dateitypen, die nicht konvertiert werden sollen. Dies gilt beispielsweise für Dateien mit anwendungsspezifischer Formatierung oder für Dateien, die bereits für einen bestimmten Druckertyp formatiert wurden. Binärdateien enthalten nicht druckbare Zeichen. Wenn Sie diese Übertragungsart wählen, werden die Dateien während der Übertragung nicht konvertiert (umgewandelt).

- ♦ ASCII

Zur Übertragung von Textdateien ohne besondere Formatierung. ASCII-Dateien auf dem PC werden in den EBCDIC-Zeichensatz auf dem Host übersetzt, und Hosttextdateien werden beim Herunterladen von EBCDIC in ASCII konvertiert.

Sie können die Dateiliste jederzeit aktualisieren, indem Sie im Dialogfeld **Dateiübertragung** rechts oben auf das Aktualisieren-Symbol klicken.

Herunterladen von Dateien

- 1 Wählen Sie aus der Liste die Datei oder das Verzeichnis aus, um die Übertragung zu starten. Sie können die Datei in dem in Schritt 3 ausgewählten Format speichern oder öffnen.
- 2 Falls nötig, können Sie die Übertragung im Bereich mit dem Fortschritt der Übertragung abbrechen.

Hochladen von Dateien

HINWEIS: IBM Mainframe-Computersysteme legen bestimmte Namenskonventionen für Dateien fest. Detaillierte Informationen zu den Anforderungen bei der Namensgebung für Dateien finden Sie in der [IBM Dokumentation](#).

Es stehen zwei Möglichkeiten für das Hochladen von Dateien zur Verfügung:

- 1 Klicken Sie im Dialogfeld „Dateiübertragung“ auf **Hochladen**.
- 2 Wählen Sie die hochzuladende Datei aus dem Fenster „Durchsuchen“ aus. Die hochgeladene Datei wird in alphabetischer Reihenfolge in der Liste angezeigt.

Oder:

- 1 Ziehen Sie die hochzuladende Datei von ihrer Speicherposition in das Dialogfeld „Dateiübertragung“.
- 2 Klicken Sie auf **Aktualisieren**, um sicherzustellen, dass die Datei erfolgreich hochgeladen wurde.

Wenn Sie den Hochladeprozess abbrechen, bevor eine Datei vollständig übertragen wurde, verbleibt eine unvollständige Datei auf dem Host.

Fehlerbehebung für Dateiübertragungen

In einigen Fällen kann es bei einer Dateiübertragung zu Fehlern kommen. Bei diesen Fehlern kann es sich um Mainframe-Probleme handeln, oder (da Sie Dateien mithilfe eines Browsers übertragen) auch Browsereinstellungen können eine Quelle unerwarteter Fehler sein.

Beispielsweise kann ein Browser Sie trotz einer fehlgeschlagenen Dateiübertragung zu einer Aktion, beispielsweise zum Speichern, auffordern. Dieses Problem kann ganz einfach durch das Ändern der Browsereinstellungen geändert werden.

Informationen zu hostspezifischen Fehlern finden Sie im Internet in der Veröffentlichung zu [IBM Fehlermeldungen bei der Dateiübertragung](#).

10 Protokollierung

Reflection ZFE verwendet Log4J 1.2 für die Implementierung der Protokollierung. Log4J verfügt über eine eigene Konfigurationsdatei und Dokumentation. Die Konfigurationsdatei, die unter `ReflectionZFE/sessionserver/conf/log4j.xml` gespeichert ist, weist eine Reihe von für die Ausgabe konfigurierten Protokollierungsstufen auf und enthält die Art von Informationen, die Sie durch Ändern der Protokollierungsstufe erfassen können.

Weitere Informationen finden Sie in der [Log4J-Dokumentation \(http://logging.apache.org/log4j/1.2\)](http://logging.apache.org/log4j/1.2).

Die folgenden Standard-Protokollierungskonfigurationen (log4j) sind verfügbar:

- ♦ Die Protokolldateiausgabe wird unter `logs/server.log` gespeichert.
- ♦ Zusätzlich zur Protokollierung der Datei `server.log` wird die gesamte Konsolenausgabe vom Reflection ZFE-Sitzungsserver erfasst und in einer Datei auf dem Datenträger gespeichert.
- ♦ Die Konfiguration, wie die Konsolenausgabe in Form einer Datei in `ReflectionZFE/sessionserver/conf/container.conf` gespeichert ist.

Zu den Konfigurationseigenschaften für die Dateispeicherung zählen unter anderem die folgenden Eigenschaften (in der Datei `container.conf` finden Sie Kommentare mit weiterführenden Informationen):

- ♦ `wrapper.logfile` – Der Speicherort der erfassten Protokolldatei (standardmäßig `.../logs/server.log`)
- ♦ `wrapper.logfile.rollmode` – Die Art und Weise, wie die vorhandene Protokolldatei als Backup gespeichert und eine neue Datei erstellt wird (standardmäßig wird, wenn die Protokolldatei eine bestimmte Größe erreicht, eine neue Protokolldatei angelegt, deren Name um eine inkrementierte Ziffer erweitert wird)
- ♦ `wrapper.logfile.maxsize` – Die maximale Größe, die die Protokolldatei erreichen kann, bevor eine neue Protokolldatei erstellt wird (Standardwert ist 10 MB)
- ♦ `wrapper.logfile.maxfiles` – Die maximale Anzahl der Protokolldateien, für die eine neue Protokolldatei erstellt wurde, die auf dem Datenträger beibehalten werden sollen (Standardwert ist 10)
- ♦ Es gibt verschiedene Typen von Protokollierungsstufen, mit denen unterschiedliche Arten von Informationen generiert werden können. Log4j unterstützt die folgenden Stufen (diese Definitionen stammen aus der Dokumentation zu Log4j, in der Sie weitere Informationen hierzu finden):
 - ♦ TRACE – Diese Stufe gibt detailliertere informative Ereignisse an als DEBUG.
 - ♦ DEBUG – Diese Stufe gibt informative Ereignisse auf Detailebene an, die sich besonders für die Fehlersuche bei einer Anwendung eignen.
 - ♦ INFO – Diese Stufe gibt Informationsmeldungen an, die den Fortschritt der Anwendung auf allgemeiner Ebene hervorheben.
 - ♦ WARN – Diese Stufe gibt potenziell gefährliche Situationen an.
 - ♦ ERROR – Diese Stufe gibt Fehlerereignisse an, bei denen die Anwendung trotzdem weiterhin ausgeführt werden könnte.
 - ♦ FATAL – Diese Stufe gibt sehr schwerwiegende Fehlerereignisse an, die sehr wahrscheinlich zu einer Beendigung der Anwendung führen.

11 Fehlerbehebung

In diesem Abschnitt finden Sie Informationen zu verschiedenen bekannten Problemen und Tipps zur Umgehung dieser Probleme für Reflection ZFE. Ein detailliertere Liste bekannter Probleme in Reflection ZFE finden Sie im [technischen Hinweis 2853](#) (in englischer Sprache).

Verbindungsprobleme bei der Installation

Genauere Informationen zur Zertifikatauthentifizierung bzw. zu weiteren Verbindungsproblemen finden Sie im [technischen Hinweis 2838](#) (in englischer Sprache).

SSL/TLS-Fehlermeldungen

- ♦ **(ECL1011) Fehler bei der Verbindung zum Host: Die Verbindung zum Host ist fehlgeschlagen.**

Dieser Fehler kann in verschiedenen Situationen angezeigt werden, die nicht nur einfach mit einem Verbindungsfehler in Zusammenhang stehen.

- ♦ Dieser Fehler kann beispielsweise auftreten, wenn eine SSL/TLS-Verbindung aufgrund eines fehlenden vertrauenswürdigen Zertifikats im MSS-Truststore fehlschlägt.
- ♦ Der Fehler wird angezeigt, wenn es bei der Verwendung von SSL/TLS für die Verbindung mit einem Host für Klartext zu einem Fehler beim TLS-Quittungsbetrieb kommt.

Anzeigen des Euro-Zeichens

- ♦ Wenn das Euro-Zeichen nicht ordnungsgemäß auf dem Terminalbildschirm angezeigt wird, wenden Sie sich an Ihren Systemadministrator, um sicherzustellen, dass der Hostzeichensatz für die Sitzung korrekt eingerichtet ist. Standardmäßig verwendet Reflection ZFE einen Zeichensatz, der das Euro-Zeichen (€) nicht unterstützt. Um das Euro-Zeichen anzuzeigen, ändern Sie den Zeichensatz in einen Satz, der das Euro-Zeichen unterstützt.

