



Einleitung

Die hier beschriebene Schnittstelle basiert auf dem SOAP Protokoll (Simple Object Access Protokoll) und wird als Web Service bereit gestellt. Hierbei werden XML-formatierte Anfragen an den FAX.de Server gesendet um Faxdokumente, SMS oder iLetter (Postbriefe) zu versenden.

Um die Schnittstelle möglichst einfach zu halten, haben wir die Aufrufe stark reduziert. Mit diesen Aufrufen lassen sich alle Services sehr schnell und effizient aus allen möglichen Arten von Office- oder Webanwendungen bedienen. Die Aufrufe 4-12 sind Services, welche Sie nutzen können um die Anwendung komfortabel zu machen.

1. Register (Registrieren von Benutzern/Änderung Benutzerdaten)
2. Send (Senden von FAX, SMS, iLetter oder eMail)
3. Status (Abfrage des Verarbeitungs-Status, Protokoll-Abfrage)
4. KundenName (Kundenname bereitstellen)
5. KundenDaten (Kundendaten bereitstellen)
6. KontoStand (Kontosumme bereitstellen)
7. Journal (Journaldaten bereitstellen)
8. KundenAccount (Kontodaten bereitstellen)
9. SendWelcomeMail (Zugangsdaten per Mail zusenden)
10. SetPassword (Neues Passwort setzen)
11. FaxReceive (Empfangenes Fax abholen)
12. FaxDelete (Empfangsfax auf dem Server löschen)

Nachfolgend wird der Aufbau der einzelnen Abfragetypen und deren Rückantwort beschrieben.

Detaillierte XML Aufrufdokumentationen (Aufbau der WSDL-Strukturen) erhalten Sie unter folgenden Adressen:

Borland Delphi 7/8: <http://ccs.fax.de/xmlws3.exe?intf=IXMLWS>

Microsoft .NET, Microsoft Soap-tools, PHP4:
<http://ccs.fax.de/xmlws3.exe?intf=IXMLWS2>

Vereinfachtes Interface (z.B. Microsoft Foxpro):
<http://ccs.fax.de/xmlws3.exe?intf=IXMLWS3>

Anwendungsbeispiele für Visual-Basic, VB.net, C#.net und PHP4 finden Sie auf www.fax.de / Download



Register

```
Int Register(    string Account,

               string Password ,

               TStringDynArray RegData)
```

Die Benutzerdaten werden in einem String-Array übergeben. Bei einer Neuanmeldung bleiben Account und Password leer. Diese sind nach erfolgreicher Registrierung serverseitig mit der vergebenen Kunden-Nummer und Passwort gefüllt. Bei einer Datenänderung muss AccountNo und Password mit übergeben werden.

Die Registerdaten werden mit Token übergeben, Beispiel „ANREDE=Herr“:

TOKEN	Feldbeschreibung	Format	Hinweis
ACCOUNT=	Kunden-Nummer	999999	Leer übergeben bei Neuregistrierung Mit @<Nummer> übergeben bei Kartenregistrierung
ANREDE=	Anrede	X(15)	z.B. „Herr“ oder „Firma“
NAME1=	Kunden-Name	X(30)	Vorname, Nachname
NAME2=	Zusatz	X(30)	Zusatz Name
STRASSE=	Strasse	X(30)	
PLZORT=	Land/PLZ/Ort	X(40)	
TELEFON=	Telefon-1	X(20)	
ANSPRECH=	Ansprechpartner-1	X(30)	
MOBIL=	Telefon-2	X(20)	
ANSPRECH2=	Ansprechpartner-2	X(30)	
BANK-KK=	Bank oder KK-Inhaber	X(30)	Bankname, wenn Feld 33="J" KK-Inhaber, wenn Feld 33=K1..K4
BLZGUELTIG=	BLZ oder gültig bis	X(8)	BLZ, wenn Feld 33="J" Gültig bis, wenn Feld 33=K1..K4
KTOKARTNR=	Konto-Nr. oder Karten-Nr.	X(20)	Kto-Nr., wenn Feld 33="J" Karten-Nr., wenn Feld 33 = K1..K4
ZAHLART=	Lastschrift/Kreditkarte	X	J = Banklastschrift K1 = Eurocard/Mastercard K2 = Visa N = per Rechnung R3, R6, R12 = PrePaid 3,6,12 Mon.
KARTENPIN=	3-4 Stellige Karten-PIN	X(4)	Die Rückseitig letzten 3-4 Ziffern
FAX=	Fax-Nummer	X(20)	



EMAIL=	Email	X(50)	
MWSTID=	MwSt-IDNummer	X(15)	DE....
SCHALTER=	Schalter	X(20)	„-EETF“ eintragen, um Faxempfang auf eMail-Konto zu aktivieren „AR=<Account>“ Angabe des Zentral-Abrechnungs-Account Mehrere Angaben mit Komma getrennt
SMSFTE=	SMS/Faxeingang	X(20)	Handy-Nr. fuer Benachrichtigung Faxempfang
EMAILFTE=	Email/Faxeingang	X(40)	E-Mail fuer Benachrichtigung Faxempfang
TARIF=	EASY-Tarif	X	P=Privat L=Light B=Business S=Superbusiness X=XML/XETF Schnittstelle T=Signatur-Tarif (Rechnung.de)
AGENTURID=	Agentur-ID	9(10)	FAX.de AgenturNr.
CARDCODE=	Karten-Code	X(6)	KartenCode (Alternativ-Passwort)

Folgende Statusmeldungen sind möglich

0	Kunde gespeichert
1	<i>Fehler Kundendaten leer oder abgelehnt</i>
4	<i>Kunden-Aenderung...Kunde nicht gefunden</i>
5	<i>Sonst. Fehler</i>
6	<i>Datenfehler: Zahlungsart falsch</i>
7	<i>Datenfehler: Tarif falsch/unbekannt</i>
8	<i>Datenfehler: BLZ falsch/unbekannt</i>
9	<i>Datenfehler: Kto-Nummer falsch</i>
10	<i>Datenfehler: eMail fehlerhaft/leer</i>
11	<i>Datenfehler: Adresse leer/fehlerhaft</i>
12	<i>MAC-Adresse gesperrt</i>

Nur wenn Status = 0 ist, sind die Daten serverseitig gespeichert. Die Parameter

Account = *Neue verwendete Kundennummer*
Password = *verwendetes Passwort/generiertes Passwort*

werden dann zurückgegeben. Das Passwort kann auch lokal vorgegeben werden, dieses muss dann beim Aufruf der Registrierung als Parameter mitgeliefert werden.

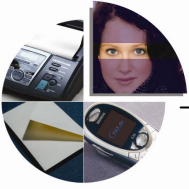


```

Int Send(
    string Account,
    string Password,
    int JobArt,
    string EmpfaengerNr,
    base64Binary EmpfaengerListe,
    string SendText,
    base64Binary SendFile1,
    base64Binary SendFile2,
    base64Binary SendFile3,
    dateTime SendeDatumZeit,
    string Schalter,
    int JobId,
    int AnzahlEmpfaenger,
    int AnzahlSeiten,
    base64Binary CheckDocFile)
  
```

Diese Funktion veranlasst den Versand eines Dokumentes. Mit dem Testflag kann der Versand auch erst einmal getestet und verifiziert werden (von PDF zu Fax-TIFF etc.)

Account	Kundennummer des Versenders
Password	Passwort des Versenders
JobArt	0=Fax 1=SMS 2=eMail 3=iLetter (Schwarz/Weiss) 4=iLetter (Farbe - ab 08/2004)
EmpfaengerNr	Empfaenger-Nummern (mit Komma separiert) bei Einzelversand, Leer bei Massenversand
EmpfaengerListe	CSV-Dateiformat für den Mengenversand, EmpfaengerNr wird ignoriert, Siehe Tabelle Empfaenger-Datei
SendText	Verwende angegebenen Plain-Text für ein Kurzfax/SMS/Brief - in die Standard-FAX.de Vorlage eingesetzt. SendText kann natürlich auch leer übergeben werden
SendFile1 ... SendFile3	Zu versendete Datei im Format TIF, BMP, HTML, DOC Nur bei Jobart=0, 3 oder 4 möglich
SendeDatumZeit	Zeit-Format, wenn leer erfolgt Sofortversand
Schalter	Verschiedene Versandoptionen, siehe Tabelle
JobId	Laufende Nummer zur Identifikation bei der Abfrage Sendestatus (Status), Bei Mengenversand wird eine Job-Nummer vergeben, die zurückgegeben wird.
AnzahlEmpfaenger	Gibt die tatsächliche Anzahl Empfänger zurück, ungültige Empfänger wurden ausgefiltert
AnzahlSeiten	Gibt die Anzahl Seiten zurück, die durch die Konvertierung zum FAX-TIFF Format erstellt wurden
CheckDocFile	Schalter="TEST", JobArt = 0 (Fax) oder 3/4 (iLetter), gibt die Dokumente im GIF-Format zur Sichtkontrolle zurück.



Schalter

Über den Parameter Schalter können verschiedene Sonderoptionen veranlasst werden:

STATIONID=xxx,xxx	Fax-StationsID zur Anzeige beim Empfänger und im Faxkopf, max. 20 Zeichen
ISDNABS=<Nr>	ISDN-Kennung des Absenders
FAXRETRY=	Anzahl Wiederholungen bei Besetzt
FAXHEADER=	Freie Definition des Faxkopfes (obere 20 Pixel), maximal 30 Zeichen
RETRY=	Anzahl Wiederholungen der ganzen Sendung
FLASHSMS	Versand der SMS als Flash
SMSFROM=<Nr>	Versand der SMS mit dieser Absendekennung
STATURL=<URL>	Optionale Meldung des Sendeergebnisses an eine URL*, maximal 128 Zeichen
TEST	Kein Versand, nur Konvertierung zur Testsichtung

Es können mehrere Schalter übergeben werden, jeder Schalter wird mit „;“ getrennt.

Die folgenden Statis werden von der Funktion Send zurückgemeldet:

0	Auftrag angenommen, wird ausgeführt
-1	Passwort ungültig
-2	Kunde nicht bekannt
-3	Dokument nicht vorhanden
-4	Serverfehler
-5	EmpfaengerNr generell gesperrt
-6	DokumentFormat nicht unterstützt
-7	Dokument nicht konvertierbar
-8	Kein Versand, da TEST Aufruf (Schalter=TEST)
-14	Dokument nicht konvertierbar
1	Nicht autorisiert
2	KundenNr ist leer
3	Kunde ist unbekannt (→ -2)
6	Guthaben noch nicht verfügbar (erste Lastschrift + 14 Tage)
7	Guthaben nicht ausreichend (nur iLetter, Jobart=3)
45	Kein Empfänger angegeben
75	Keine Dateien und/oder Text zum Versand übergeben

*URL-Status

Optional kann über den Schalter eine URL angegeben werden. Hierzu muss die Kopfadresse in dem Schalter mit Platzhaltern für die Ergebnisse angegeben werden. Diese wird nach erfolgtem Versand aufgerufen. Folgende Platzhalter sind möglich:

\$JID	JobID dieses Auftrages
\$RCV	Empfaenger-Nummer / Adresse
\$STAT	Sende-Status wie Aufruf „Status()“ mit verkürztem Ergebniss (OK,BUSY,NOANSWER,VOICE,NORING,ISDNERR,NOSERVICE,BLACKLIST)
\$MSGGS	Anzahl Seiten / Messages
\$COST	Sende-Gebühren in Euro netto + MwSt
\$RID	Remote-ID des Empfängers (nur FAX)
\$CID	Kunden-ID des Versenders (wird in [] hinter Empfänger-Nr. gegeben)
\$DATE	Sendedatum im Format DD.MM.YY
\$TIME	Sendezeit (Ende Übertragung) im Format HH:MM:SS

Der URL-Status kann nur bei erfolgreicher Einlieferung gesendet werden (Status Send() = 0).

Beispiel:



Schalter = „STATURL=www.fax.de/status &kdid=2010 &JOBID=\$JID &STAT=\$STAT“
URL-Aufruf von FAX.de: „www.fax.de/status &kdid=2010 &JOBID=1356665 &STAT=OK“



**Aufbau der Empfänger-Datei (CSV-Format),
Komma als Feld-Trenner, CrLf als Zeilentrenner:**

„Empfänger“,	Fax-Nr., Handy-Nr., eMail-Adresse, BriefAdresse(1..30)
„Kunden-Id“,	Zur Identifikation im Sendeprotokoll
„Personalisierung-1“,	Fax: FaxHeader oben links (Nur OHNE Station-ID möglich) Brief: Anrede EMail: Variablenfeld \$1\$
„Personalisierung-2“,	Fax: FaxHeader oben rechts (Nur OHNE Station-ID möglich) Brief: Name-1 Email: Variablenfeld \$2\$
„Personalisierung-3“,	Fax: FaxHeader 2. Zeile links (Nur OHNE Station-ID) Brief: Name-2 EMail: Variablenfeld \$3\$
„Personalisierung-4“,	Fax: Faxheader 2. Zeile rechts (Nur OHNE Station-ID) Brief: Strasse/Hausnummer EMail: Variablenfeld \$4\$
„Personalisierung-5“,	Brief: Land/PLZ/Ort EMail: Variablenfeld \$5\$
„Personalisierung-6“,	Brief: Verwendungszweck/Ordnungsnummer EMail: Variablenfeld \$6\$
„Personalisierung-7-10“,	EMail: Variablenfeld \$7\$... \$10\$
Zeilenende	CrLf oder nur Lf (Unix/Linux)



Int **Status**(string **Account**,
 string **Password**,
 string **JobId**,
 string **ProtokollFilename**,
 base64Binary **Protokoll**)

Mit diesem Aufruf wird eine einzelne Sendung beauskunftet. Dies ist nur möglich, wenn bei der Versendung (Send) eine JobID zur Identifikation mitgegeben wurde oder aufgezeichnet wurde. Zurückgegeben wird der Protokolleintrag, der in Bezug zur JobID steht. Bei Mengenversand können dies n-Protokolleinträge sein, die in Bezug zur Auftragsnummer (in JobID übergeben) stehen.

Die Protokoll-Einträge werden als Kommaseparierte Liste übergeben, die Erste Zeile beinhaltet den Aufbau des Protokolls in Form von TOKEN. Beispiel

Start Liste:

„EMPFAENGER“,„JOBART“,„STATUS“,„ANZSEITEN“,„STARTDATE“,„STARTTIME“,„KUNDENID“,
 „04038045211“,„FAX“,„OK: +49 40 380 45 211“,„2“,„15.02.06“,„14:11:17“,„150788“

Aufbau Protokoll:

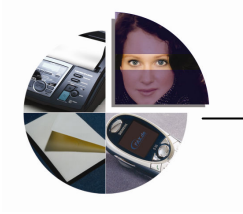
TOKEN	Format	Inhalt	Bemerkung
„EMPFAENGER“	X(16)	Fax-Nr/Handy-Nr/eMail-Empfänger	
„JOBART“	X(8)	JobArt-Text	FAX, SMS, Mail, Brief
„STATUS“	X(15)	Sende-Status-Text	Siehe Fax-Sendestati
„ANZSEITEN“	99	Anzahl Seiten/Messages	
„STARTDATE“	99.99.99	Startdatum (DD.MM.JJ)	
„STARTTIME“	99.99.99	Startzeit (HH:MM:SS)	
„SENDSEC“	9999	Sendezeit in Sekunden	Nur Fax (JobArt=0)
„KUNDENID“	X(20)	Kunden-ID	Kunden-Nr. für Zuordnung

Fehlermeldungen Status:

0	Protokoll-eintrag gefunden, Siehe Protokoll-Variable
1	Account nicht gefunden
2	Passwort stimmt nicht zum Account
3	Konto Testzeit abgelaufen
4	Account gesperrt
5	Kein Protokoll-Eintrag mit dieser JobId gefunden
6	Job ist derzeit in Bearbeitung

Sendestati:

OK	Fax, SMS, Brief wurde gesendet
Busy	Empfänger wurde dreimal angewählt, nach 1, 5 und 10 Minuten. Empfänger war ständig besetzt
No Answer	Gegenseite klingelt, niemand nimmt ab
Voice	Ansage oder ein Mensch hat geantwortet (Kostenpflichtig)
No Service	Kein Service auf dieser Empfängernummer geschaltet
Send-Fail	Erfolgreiche Faxverbindung, Fax nimmt das Dokument nicht an (z.B. Papierstau)
Call failed	Nummer zu kurz/falsch, Kein verwendeter Anschluss, Dienstmerkmal nicht unterstützt
Black-List	Empfängernummer ist auf der Sperrliste und darf nicht angewählt werden





Int **KundenName**(string **Account**,
 string **Password**,
 string **KdName**)

Diese Funktion gibt den Namen zur Anzeige auf der Website eines bestimmten Kunden zurück.

Rueckgabe-Stati:

0	OK, KdName ist mit dem Kunden-Namen gefüllt
1	Account nicht gefunden
2	Password stimmt nicht zum Account
3	Konto Testzeit abgelaufen
4	Account gesperrt

Int **KundenDaten**(string **Account**,
 string **Password**,
 TStringDynArray **KdDaten**)

Diese Funktion gibt die Kundendaten zur Anzeige/Änderung auf der Website eines bestimmten Kunden zurück. Die Daten werden analog zur Funktion Register() zurückgegeben.

Rückgabe-Stati:

0	OK, KdDaten ist mit den Kunden-Daten als String-Array gefüllt, Inhalt siehe Funktion Register
1	Account nicht gefunden
2	Password stimmt nicht zum Account
3	Konto Testzeit abgelaufen
4	Account gesperrt



Int **Kontostand**(string **Account**,
string **Password**,
double **KdKontoStand**)

Diese Funktion gibt das Guthaben zur Anzeige eines bestimmten Kunden zurück.

Rueckgabe-Stati:

0	OK, Guthaben ist in KdKontoStand angegeben
1	Account nicht gefunden
2	Password stimmt nicht zum Account
3	Konto Testzeit abgelaufen
4	Account gesperrt



Int **Journal**(string **Account**,
 string **Password**,
 int **AnzLines**,
 int **OffSet**,
 TStringDynArray **JournalTab**)

Diese Funktion gibt eine Journaltabelle der letzten Versendungen zurück.
AnzLines gibt an, wieviele Einträge zurückgegeben werden sollen (Maximum 100).
OffSet gibt an, ab welchem Eintrag zurück die Tabelle angegeben werden soll. Damit ist ein Blättern im Journal Seitenweise möglich.

Start Liste:

„EMPFAENGER“,„JOBART“,„STATUS“,„ANZSEITEN“,„STARTDATE“,„STARTTIME“,„KUNDENID“,
 „04038045211“,„FAX“,„OK: +49 40 380 45 211“,„2“,„15.02.06“,„14:11:17“,„150788“

.

Aufbau Protokoll:

TOKEN	Format	Inhalt	Bemerkung
„EMPFAENGER“,	X(16)	Fax-Nr/Handy-Nr/eMail-Empfänger	
„JOBART“,	X(8)	JobArt-Text	FAX, SMS, Mail, Brief
„STATUS“,	X(15)	Sende-Status-Text	Siehe Fax-Sendestati
„ANZSEITEN“,	99	Anzahl Seiten/Messages	
„STARTDATE“,	99.99.99	Startdatum (DD.MM.JJ)	
„STARTTIME“,	99.99.99	Startzeit (HH:MM:SS)	
„SENDSEC“,	9999	Sendezeit in Sekunden	Nur Fax (JobArt=0)
„KUNDENID“,	X(20)	Kunden-ID	Kunden-Nr. für Zuordnung

Der Aufbau einer Journalzeile entspricht dem Aufbau bei **Status**.

Sendestati:

OK	Fax, SMS, Brief wurde gesendet
Busy	Empfänger wurde dreimal angewählt, nach 1, 5 und 10 Minuten. Empfänger war ständig besetzt
No Answer	Gegenseite klingelt, niemand nimmt ab
Voice	Ansage oder ein Mensch hat geantwortet (Kostenpflichtig)
No Service	Kein Service auf dieser Empfängernummer geschaltet
Send-Fail	Erfolgreiche Faxverbindung, Fax nimmt das Dokument nicht an (z.B. Papierstau)
Call failed	Nummer zu kurz/falsch, Kein verwendeter Anschluss, Dienstmerkmal nicht unterstützt
Black-List	Empfängernummer ist auf der Sperrliste und darf nicht angewählt werden



Int **KundenAccount**(string **Account**,
string **Password**,
TStringDynArray **MyAccountData**)

Diese Funktion gibt eine genaue Aufschlüsselung der Kontodaten/Abrechnungsdaten zurück. Die KontoDaten werden wiederum als Kommaseparierte Liste zurückgegeben, wobei der erste Eintrag die TOKEN-Liste beinhaltet. Beispiel

„BELEGTEXT“, „MENGE“, „EPREIS“, „RABATT“, „GPREIS“
„Unsere Forderungen per 18.2.04“, „“, „“, „“, „10,06“
„Ab 02.02.“, „“, „“, „“, „“
„SPAR-BRD-N 1 Fax-Seite“, „2“, „0,3200“, „5%“, „0,6080“

TOKEN	Format	Inhalt	Bemerkung
„BELEGTEXT“	X(80)	Belegtext	
„MENGE“	9999999	Menge	
„EPREIS“	9999999,9999	Einzelpreis	4 Nachkommastellen
„RABATT“	X(8)	Rabatt	
„GPREIS“	9999999,9999	Gesamtpreis	4 Nachkommastellen

Int **SendWelcomeMail**(string **Account**)

Diese Funktion veranlasst das FAX.de System, die Registrierungsmail (mit dem Anmelde-Passwort) an den Kunden zu senden.

Rueckgabe-Stati:

0	OK, Welcome-Mail wurde versendet
1	Account nicht gefunden
2	Password stimmt nicht zur Account
3	Konto Testzeit abgelaufen
4	Account gesperrt

Int **SetPassword**(string **Account**,
string **OldPassword**,
string **NewPassword**)



Diese Funktion ermöglicht die Änderung des Passwortes eines bestimmten Accounts.

Rueckgabe-Stat:

0	OK, Passwort wurde geändert
1	Account nicht gefunden
2	Passwort stimmt nicht zur Account
3	Konto Testzeit abgelaufen
4	Account gesperrt



```
Int FaxReceive(    string Account,  
  
                  string Password,  
  
                  string FaxFile,  
  
                  dateTime ReceiveDate,  
  
                  base64Binary FaxImage)
```

Diese Funktion holt empfangene Faxe für einen bestimmten Account vom FAX.de Server ab. Das empfangene Fax wird als MultiPage-TIFF Datei bereitgestellt.

Wichtig:

Nach der erfolgreichen Übertragung der Faxdatei muss das Dokument mit der Funktion „FaxDelete“ gelöscht werden, anderenfalls wird immer wieder das gleiche Fax übergeben.

Rueckgabe-Stati:

0	EmpfangsFax gefunden, Datei wird als FaxImage übergeben
1	Account nicht gefunden
2	Password stimmt nicht zur Account
3	Konto Testzeit abgelaufen
4	Account gesperrt
5	Kein Fax im Fax-Eingang

```
Int FaxDelete(  string Account,  
  
               string Password,  
  
               string FaxFile)
```



Diese Funktion löscht ein geholtes Empfangsfax aus dem Empfangsspeicher bei FAX.de

Rueckgabe-Stati:

0	Fax wurde aus der Inbox gelöscht
1	Account nicht gefunden
2	Passwort stimmt nicht zur Account
3	Konto Testzeit abgelaufen
4	Account gesperrt
5	Faxdatei nicht gefunden



Int **CheckPromoCode**(string **Account**,
string **PromoCode**)

Diese Funktion prüft den vorgegebenen Promotion-Code (Code auf der FAX.de Kundenkarte) auf Gültigkeit.

Rueckgabe-Stati:

0	Promotionsguthaben in voller Höhe verfügbar
1	Account nicht gefunden
2	PromoCode stimmt nicht
3	Konto Testzeit abgelaufen
4	Account gesperrt
5	Promotionguthaben bereits eingelöst



XML-Beispiel Register

GESENDET:

```
<?xml version="1.0"?>
```

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">
```

```
<SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
```

```
<NS1:Register xmlns:NS1="urn:XMLWSIntf-IXMLWS">
```

```
<Account xsi:type="xsd:string">
```

```
</Account>
```

```
<Password xsi:type="xsd:string">
```

```
</Password>
```

```
<RegData xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="xsd:string[3]">
```

```
<item>
```

```
ANREDE=Herr
```

```
</item>
```

```
<item>
```

```
NAME1=Manfred Muster
```

```
</item>
```

```
!!
```

```
!! weitere Tokens mit Kundeninfos lt. Schnittstellenbeschreibung
```

```
!!
```

```
</RegData>
```

```
</NS1:Register>
```

```
</SOAP-ENV:Body>
```



</SOAP-ENV:Envelope>

Empfangen:

<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

<SOAP-ENV:Body SOAP-ENC:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">

<NS1:RegisterResponse xmlns:NS1="urn:XMLWSIntf-IXMLWS">

<return xsi:type="xsd:int">

0

</return>

<Account xsi:type="xsd:string">

158123

</Account>

<Password xsi:type="xsd:string">

secret

</Password>

</NS1:RegisterResponse>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>



XML-Beispiel Send

GESENDET:

--MIME_boundaryB0R9532143182121

Content-Type: text/xml

SOAPAction: "urn:XMLWSIntf-IXMLWS#Send"

Content-ID:

<http://www.borland.com/rootpart.xml>

Content-Location: http://www.borland.com/rootpart.xml

Content-Length: 1075

<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"

xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

<SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

<NS1:Send xmlns:NS1="urn:XMLWSIntf-IXMLWS">

<Account xsi:type="xsd:string">

158123

</Account>

<Password xsi:type="xsd:string">

secret

</Password>

<JobArt xsi:type="xsd:int">

1

</JobArt>



<EmpfaengerNr xsi:type="xsd:string">

04038017851515

</EmpfaengerNr>

<EmpfaengerListe href="cid:AC4AB75F-291E-47FA-B6BF-BC3EB32B1FD5"/>

<SendText xsi:type="xsd:string">

Hallo

</SendText>

<SendFile1 href="cid:4695C5C3-6D8F-4790-B005-EF26FC652AC5"/>

<SendFile2 href="cid:89D1AE2F-AB6D-4A7D-906F-39F4BD1C494D"/>

<SendFile3 xsi:nil="true"/>

<SendeDatumZeit xsi:type="xsd:dateTime">

2004-02-25T12:29:27.812+01:00

</SendeDatumZeit>

<Schalter xsi:type="xsd:string">

</Schalter>

<CheckDocFile xsi:type="xsd:boolean">

false

</CheckDocFile>

</NS1:Send>

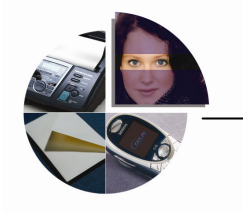
</SOAP-ENV:Body>

</SOAP-ENV:Envelope>

--MIME_boundaryB0R9532143182121

Content-ID:

<AC4AB75F-291E-47FA-B6BF-BC3EB32B1FD5>



Content-Length: 41

Content-Type: application/binary

Content-transfer-encoding: binary



ANMERKUNG: Attachment Daten hier

--MIME_boundaryB0R9532143182121

Content-ID:

<4695C5C3-6D8F-4790-B005-EF26FC652AC5>

Content-Length: 9959

Content-Type: application/pdf

Content-transfer-encoding: binary

ANMERKUNG: Attachment Daten hier

--MIME_boundaryB0R9532143182121

Content-ID:

<89D1AE2F-AB6D-4A7D-906F-39F4BD1C494D>

Content-Length: 44209

Content-Type: image/tif

Content-transfer-encoding: binary

ANMERKUNG: Attachment Daten hier

--MIME_boundaryB0R9532143182121--



EMPFANGEN:

```
<?xml version="1.0"?>
```

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-  
ENC="http://schemas.xmlsoap.org/soap/encoding/">
```

```
<SOAP-ENV:Body SOAP-ENC:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">
```

```
<NS1:SendResponse xmlns:NS1="urn:XMLWSIntf-IXMLWS">
```

```
<return xsi:type="xsd:int">
```

```
0
```

```
</return>
```

```
<JobId xsi:type="xsd:int">
```

```
900123
```

```
</JobId>
```

```
</NS1:SendResponse>
```

```
</SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```



XML-Beispiel Journal

gesendet:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

    <NS1:Journal xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <Account xsi:type="xsd:string">

        158123

      </Account>

      <Password xsi:type="xsd:string">

        secret

      </Password>

      <AnzLines xsi:type="xsd:int">

        100

      </AnzLines>

      <Offset xsi:type="xsd:int">

        0

      </Offset>

    </NS1:Journal>

  </SOAP-ENV:Body>

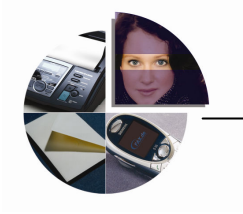
</SOAP-ENV:Envelope>
```

empfangen:

```
<?xml version="1.0"?>
```



```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Body SOAP-ENC:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">
    <NS1:JournalResponse xmlns:NS1="urn:XMLWSIntf-IXMLWS">
      <return xsi:type="xsd:int">
        0
      </return>
      <JournalTab xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="xsd:string[31]">
        <item>
          "EMPFAENGER","JOBART","ANZSEITEN","STARTDATE","STARTTIME","SENDSEC","KUNDENID"
        </item>
        <item>
          "0418579522","FAX","OK:+49 4185 7","1","04.02.04","17:21:00","28","+49 4185 7"
        </item>
        <item>
          "04185795260","FAX","OK:+49 00 000","1","12.01.04","13:15:00","24","+49 00 000"
        </item>
        <item>
          "04185795260","FAX","OK:+49 00 000","1","12.01.04","13:12:00","24","+49 00 000"
        </item>
        <item>
          "04185795260","FAX","OK:+49 00 000","1","12.01.04","13:07:00","24","+49 00 000"
        </item>
      </JournalTab>
    </NS1:JournalResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```





XML-Beispiel Kontostand

Gesendet:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

    <NS1:Kontostand xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <Account xsi:type="xsd:string">

        158123

      </Account>

      <Password xsi:type="xsd:string">

        secret

      </Password>

    </NS1:Kontostand>

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

empfangen:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENC:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">

    <NS1:KontostandResponse xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <return xsi:type="xsd:int">

        0

      </return>

    </NS1:KontostandResponse>

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```



```
<KdKontostand xsi:type="xsd:double">
```

```
47.11
```

```
</KdKontostand>
```

```
</NS1:KontostandResponse>
```

```
</SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```



XML-Beispiel Kundendaten

gesendet:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

    <NS1:Kundendaten xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <Account xsi:type="xsd:string">

        158123

      </Account>

      <Password xsi:type="xsd:string">

        secret

      </Password>

    </NS1:Kundendaten>

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

empfangen:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENC:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">

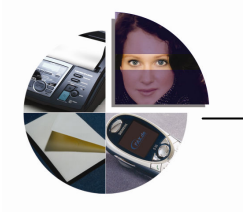
    <NS1:KundendatenResponse xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <return xsi:type="xsd:int">

        0

      </return>

      <KdDaten xsi:type="SOAP-ENC:Array" SOAP-ENC:arrayType="xsd:string[22]">
```



<item>

ACCOUNTNO=158123

</item>

<item>

ANREDE=Herr

</item>

<item>

NAME1=Manfred Muster

</item>

<item>

NAME2=

</item>

<item>

STRASSE=Hauptstrasse 1

</item>

<item>

PLZORT=21111 Testdorf

</item>

<item>

TELEFON=+49 (4185) 79 52 0

</item>

<item>

ANSPRECH=

</item>

<item>

MOBIL=

</item>

<item>



ANSPRECH2=

</item>

<item>

BANK-KK=Harry Hirsch

</item>

<item>

BLZGUELTIG=12/05

</item>

<item>

KTOKARTNR=123456789012345

</item>

<item>

ZAHLART=K1

</item>

<item>

AGENTURID=0

</item>

<item>

KARTENPIN=123

</item>

<item>

FAX=

</item>

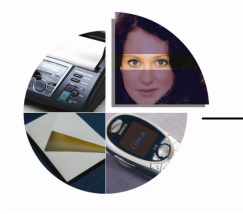
<item>

EMAIL=ManfredMuster@fax.de

</item>

<item>

MWSTID=



</item>

<item>

TARIF=P

</item>

<item>

SCHALTER=

</item>

<item>

SMSFTE=

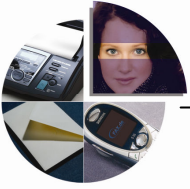
</item>

</KdDaten>

</NS1:KundendatenResponse>

</SOAP-ENV:Body>

</SOAP-ENV:Envelope>



XML-Beispiel SendWelcomeMail

gesendet:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

    <NS1:SendWelcomeMail xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <Account xsi:type="xsd:string">

        158123

      </Account>

    </NS1:SendWelcomeMail>

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

empfangen:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENC:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">

    <NS1:SendWelcomeMailResponse xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <return xsi:type="xsd:int">

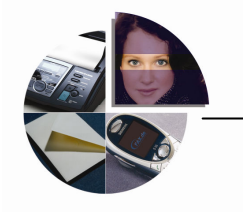
        0

      </return>

    </NS1:SendWelcomeMailResponse>

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```





XML-Beispiel Setpassword

gesendet:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

    <NS1:SetPassword xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <Account xsi:type="xsd:string">

        158123

      </Account>

      <OldPassword xsi:type="xsd:string">

        secret

      </OldPassword>

      <NewPassword xsi:type="xsd:string">

        geheim

      </NewPassword>

    </NS1:SetPassword>

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

Empfangen:

```
<?xml version="1.0"?>
```



```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENC:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">

    <NS1:SetPasswordResponse xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <return xsi:type="xsd:int">

        0

      </return>

    </NS1:SetPasswordResponse>

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```



XML-Beispiel Status

gesendet:

```
<?xml version="1.0"?>

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">

  <SOAP-ENV:Body SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

    <NS1:Status xmlns:NS1="urn:XMLWSIntf-IXMLWS">

      <Account xsi:type="xsd:string">

        158123

      </Account>

      <Password xsi:type="xsd:string">

        secret

      </Password>

      <JobId xsi:type="xsd:int">

        900123

      </JobId>

    </NS1:Status>

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

empfangen:

```
--MIME_boundaryB0R9532143182121

Content-Type: text/xml

Content-ID:

<http://www.borland.com/rootpart.xml>
```



Content-Location: <http://www.borland.com/rootpart.xml>

Content-Length: 550

```
<?xml version="1.0"?>
```

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/">
```

```
<SOAP-ENV:Body SOAP-ENC:encodingStyle="http://schemas.xmlsoap.org/soap/envelope/">
```

```
<NS1:StatusResponse xmlns:NS1="urn:XMLWSIntf-IXMLWS">
```

```
<return xsi:type="xsd:int">
```

```
0
```

```
</return>
```

```
<Protokoll href="cid:B1DB255F-9CF8-4C49-975A-C8E257EDA959"/>
```

```
</NS1:StatusResponse>
```

```
</SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```

--MIME_boundaryB0R9532143182121

Content-ID:

```
<B1DB255F-9CF8-4C49-975A-C8E257EDA959>
```

Content-Length: 8980

Content-Type: text/plain

Content-transfer-encoding: binary

!! Attachment Daten hier

--MIME_boundaryB0R9532143182121--