

## Beschreibung Webservice „SendData“

4.1 vom 14.04.2007

Mit der Schnittstelle „SendData“ stellt der LTS eine neue Datenbank-Schnittstelle zur Verfügung, über die es möglich ist, alle touristisch relevanten Daten ein- und auszulesen. Die Schnittstelle kann über einen XML-Webservice angesprochen werden. Die Zugangsdaten könne beim LTS angefordert werden.

Der Aufruf erfolgt über: <http://tourist.bz.it/data/CDBData.asmx>

Über die Funktion "SendData" werden sämtliche Daten mittels XML-Strings übergeben. Dieser XML-String wird ausgewertet und dann ein XML-String als Antwort zurückgegeben.

Ein typischer Aufruf in VB.NET sieht folglich so aus:

```
Dim oLTS As New CDB.CDBData() // CDB entspricht der Proxy-Classe
string sX = "<?xml version='1.0' encoding='UTF-8' ?>
    <SendData UserID='8970B4AF0A15483BBD4DDC71E06EB96C'
    SessionID='7650B4AF0A15483BBD4DDC71E06EB96C'
    Function='DeleteHotel' Version='1.0'
    AORID='9990B4AF0A15483BBD4DDC71E06EB96C' />"
Me.TextBox1.Text = oLTS.SendData(sX).ToString()
```

## Hinweise

- Set-Funktionen führen ein „Delete“ auf der Datenbank aus und löschen alle Daten eines bestimmten Betriebes, Gruppe, Angebot, Extra, Sicht oder Sicht-Angebot. Darauf folgt ein „Insert“. Es müssen also immer alle Daten übergeben werden. Eine Ausnahme sind Funktionen, bei denen es eine eigene „Delete“- Funktion gibt. Diese arbeiten mit Update.
- Farben werden immer im Hex-Format übergeben (z.B. "#ffffff")
- Alle Datumswerte werden im Format YYYY-MM-DDTHH:MM:SS übergeben bzw. zurückgegeben. Es ist aber auch möglich YYYY/MM/DD HH:MM:SS zu übergeben.
- Alle Zeitwerte werden im Format YYYY-MM-DDTHH:MM:SS übergeben bzw. zurückgegeben. Es ist aber auch möglich HH:MM:SS oder HH:MM zu übergeben
- Alle Entfernungen werden immer in Metern angegeben
- Encoding UTF-8 / XML- Version 1.0
- Alle Aufrufe sind durch eine Funktions-Version gekennzeichnet
- Die Nutzung jeder einzelnen Funktion muss genehmigt werden da diese sonst ohne Vorwarnung abgeändert werden kann.
- Null-Werte sind nicht zugelassen, es muss immer zumindest ein Defaultwert übergeben werden
- Das Schreiben bzw. Löschen der Daten wird immer mit 0 (nicht korrekt), 1 oder größer (korrekt) oder mit Error quittiert.

Beispiel: `<RespData><Data Value="1" /></RespData>`

- Beim Aufruf muss die UserID und die SessionID übergeben werden (diese werden vom LTS fix oder dynamisch vergeben)
- Beim Aufruf wird als Version die Versionsnummer der Funktion übergeben
- Im Feld LngID steht immer das / die Kürzel der Sprache/n ("DE" oder "DE;IT;EN; : : :")
- Alle Angaben von kompletten Links werden mit "http://" angegeben (z.B. http://tourist.bz.it)
- Die Fehlermeldung "<Error>Invalid login</Error>" bedeutet, dass Sie keine Berechtigung haben auf die Schnittstelle zuzugreifen
- Die Fehlermeldung " <Error>Access denied</Error>" bedeutet, dass Sie nicht berechtigt sind diese Funktion auszuführen. Entweder wurde die Funktion für Sie nicht freigegeben, oder Sie sind nicht berechtigt diese Funktion für die angegebenen RID auszuführen.
- Bitte beachten Sie, dass mit einem Ticket nur die Daten des jeweiligen Betriebs verarbeitet werden.
- Um eine zusätzliche Interaktion mit dem Benutzer zu ermöglichen, liefern einige Funktionen bei der Antwort im Element „Data“ bzw. „Error“ ein zusätzliches Attribut mit Namen „Msg“. In diesem Attribut findet man für den Benutzer (verständlich) aufbereitete Informationen. Achtung: das Attribut ist optional und ist nicht zwingend vorhanden.
- Sollte der Webservice nicht mittels Proxyklassen von VisualStudio angesprochen werden, muss auf die Maskierung der Sonderzeichen Acht gegeben werden.
- Für den Aufruf der Funktionen „SendData“, „SendMYS“ und „BackOff“ gibt es die erweiterten Funktionen „...Xml“ und „...XmlOut“. „...Xml“ erwartet den Input als Xml und nicht als String. Sowohl „...Xml“ als auch „...XmlOut“ geben die Daten als Xml aus.
- Bitte beachten Sie dass der bis zum 08.02.2007 eingesetzte Webservice „http://data.tourist.info/...“ nicht mehr verwendet werden darf und mit 30.06.2007 abgeschaltet wird. Verwenden Sie statt dessen den neuen Webservice „http://tourist.bz.it/...“

## Beispiel einer Anfrage / Antwort:

### Abfrage:

SOAPAction: "http://tourist.bz.it/data/GetTicket"

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<GetTicket xmlns="http://tourist.bz.it/data">
<sInput>&lt;SendData User='...' Pwd='...' Function='GetTicket' Version='1.0'
PrefUserID='...' /&gt;</sInput>
</GetTicket>
</soap:Body>
</soap:Envelope>
```

### Antwort:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Body>
<GetTicketResponse xmlns="http://tourist.bz.it/data">
<GetTicketResult>&lt;?xml version='1.0' encoding='UTF-
8'?&gt;&lt;RespData&gt;&lt;Ticket A0RID='...' RID='...' SessionID='...' Time='6000'
UserID='...' LngID='DE' Type='POR' /&gt;&lt;/RespData&gt;
</GetTicketResult>
</GetTicketResponse>
</soap:Body>
</soap:Envelope>
```

### Anmerkung am Rande:

*Entities und Entity-Referenzen*

*Enthält ein einfacher Text bereits Passagen, die als XML-Markup missverstanden werden könnten, muss die öffnende spitze Klammer innerhalb der Textdaten eines XML-Dokuments maskiert werden. XML bietet dazu so genannte Entities.*

*Die am häufigsten verwendeten analysierten Entities wirken wie ein einfacher Textersetzungsmechanismus. So bedeutet die Notation `&lt;`, dass an dieser Stelle der Inhalt (der Ersetzungstext) des Entity `lt` eingesetzt werden soll; in diesem Fall das Zeichen `<` (less than). Eine Entity-Referenz besteht immer aus dem Ampersand (`&`), dem Namen des Entity und einem Semikolon. Das Ampersand selbst muss nun seinerseits maskiert werden, soll es nur für sich selbst stehen und nicht den Beginn einer Entity-Referenz anzeigen. XML stellt dazu das Entity `amp` zur Verfügung. Weitere vordefinierte Entities sind `gt` (`>`), `apos` (`'`) und `quot` (`"`).*

## Beispiel für eine Fehlermeldung / Antwort:

```
<RespData><Error>Error on process SendData</Error></RespData>
```

## **Unterschiede zwischen den Aufrufen „SendData“, „SendDataXmlOut“ und „SendDataXml“**

SendData: die Ein- und Ausgabe der Daten erfolgt als String, verpackt in ein XML

SendDataXml: die Ein- und Ausgabe der Daten erfolgt als XML

SendDataXmlOut: die Eingabe der Daten erfolgt als String, die Ausgabe erfolgt als XML

## Funktionen der Betriebe

Anmerkung: Von einigen Funktionen gibt es mehrere Versionen (gekennzeichnet durch „S1“, „S2“, . . .). Für die Auswahl der richtigen Funktion beachten Sie bitte die Beschreibung rechts. Standardmäßig stehen nur die Get-Funktionen zur Verfügung. Set- und Delete- Funktionen sind gesperrt.

### HotelData

(HotelDataS1)

(HotelDataS2 nur GET)

(HotelDataS3 nur GET)

Gesamte Betriebsdaten

Ges. Betriebsdaten eingeschränkt

Ges. Betriebsdaten eingeschränkt **für Websides**

Ges. Betriebsdaten eingeschränkt **für Statistik**

### HotelBaseData

(HotelBaseDataS1)

(HotelBaseDataS2 nur GET)

(HotelBaseDataS2 nur GET)

Betrieb Basisdaten

Betrieb Basisdaten eingeschränkt

Betrieb Basisdaten eingeschränkt **für Websides**

Betrieb Basisdaten eingeschränkt **für Statistik**

HotelDefinition

(HotelDefinitionS1)

(HotelDefinitionS2 nur SET)

Betrieb Definition

Betrieb Definition eingeschränkt **für Websides**

Betrieb Definition eingeschränkt **nur für LTS**

HotelAddress

Betrieb Adresse

HotelWebData

Betrieb Webdaten

HotelPublicity

Betrieb Werbung

HotelFoto

Betrieb Fotos

HotelPosition

Betrieb Position / Distanzen

HotelOverview

Betriebsübersicht

HotelTin

Betrieb TIN-Kodes

HotelTVPay

Betrieb Zahlungsart an Muttersicht

**(in HotelBaseDataS2 nicht verfügbar)**

HotelBankData

Betrieb Bankdaten

**(in HotelBaseDataS2 nicht verfügbar)**

HotelVG

Betrieb Muttersicht (wird nicht von übergeordneten Funktionen aufgerufen)

HotelConference

Tagungseinrichtungen

HotelTypeOfAddress

Zuweisung Adresse zu Adresstypen

### GroupBaseData

GroupDefinition

Gruppe alle Basisdaten (Zimmergruppen)

GroupName

Gruppe Definition

GroupRData

Gruppe Namen

GroupRData\_Times

Gruppe Freie Zimmer

GroupTin

Gruppe Freie Zimmer (Format für Hotelprog.)

GroupPublicity

Gruppe TIN-Kodes

GroupFoto

Gruppe Werbung

Gruppe Fotos

### OfferBaseData

OfferDefinition

Angebot alle Basisdaten

OfferName

Angebot Definition

OfferPublicity

Angebot Name

OfferTin

Angebot Werbung

OfferSeason

Angebot TIN-Kodes

OfferPrice

Angebot Saisonsen

Angebot Preise

OfferExChargeType  
OfferExChargeTime  
OfferExChargeName  
OfferExCharge

Angebot Zu / Abschlag Definition  
Angebot Einheiten des Zuschlags  
Angebot Bezeichnung  
Angebot Zu / Abschlag

**ExtraBaseData**

ExtraDefinition  
ExtraName  
ExtraPrice

Betrieb Extras Definition  
Betrieb Extras Name  
Betrieb Extras Preis

**SetFN1**

Übergabe der FN1-Meldungen

## Funktionen der Sichten (z.B. Tourismusverein)

<b>VGroupData</b>	Gesamten Daten einer Sicht
<b>VGroupBaseData</b>	Sicht Basisdaten
VGroupDefinition	Sicht Definition
VGroupName	Sicht Name
VGroupPosition	Sicht Position
VGroupRoomVal	Sicht Zimmer Gültigkeiten
VGroupRoles	Sichten zugewiesene Hotels über Rollen
VGroupHierarchy	Sichten Hierarchische Anordnung
VGroupA0RID	Sichten fix zugewiesene Hotels
VGroupAddress	Sicht Adresse
VGroupWebData	Sicht Webdaten
VGroupPublicity	Sicht Werbung
VGroupFoto	Sicht Fotos
VGroupTin	Sicht Tin-Kodes
VGroupTypeOfAddress	Sicht Adresstypen
<b>VGOfferBaseData</b>	Sicht Angebot Basisdaten
VGOfferDefinition	Sicht Angebot Definition
VGOfferName	Sicht Angebot Name
VGOfferPublicity	Sicht Angebot Werbung
VGOfferTin	Sicht Angebot Tin-Kodes
VGOfferA0RID	Sicht Angebot zugewiesene Betriebe

Anmerkung: „VGOfferBaseData“ enthält aktuell keine Daten

## Funktionen für Zimmersuche

<b>SeekEngine</b>	Betriebssuche
-------------------	---------------

## Allgemeine Funktionen

<b>Tin</b>	Ausgabe der TIN-Kodexe
<b>Bank</b>	Abruf der in der CDB verfügbaren Banken
<b>Fraktion</b>	Abfrage der Fraktionen Südtirols
<b>FormOfAddress</b>	Abfrage der Anrede-Typen
<b>AstatCompanys</b>	Abfrage der Astat-Betriebstypen
<b>Nation</b>	Abfrage der Nationen
<b>SeekEngine</b>	Zimmersuche
<b>PHotelOverView</b>	Schnellübersicht über mehrere Betriebe

## Login / Anfrage eines Tickets für die DB-Schnittstelle

Mit der Funktion „GetTicket“ kann ein gültiges Ticket angefordert werden. Dabei sind die **WEB-Zugangsdaten** des Betriebs zu übergeben.

Als Antwort wird ein Ticket generiert, das die eindeutige Kennung A0RID des Betriebs, sowie UserID, SessionID und die Gültigkeitsdauer des Tickets in Sekunden angibt.

Es wird empfohlen, nach Ablauf der Hälfte der Zeit ein neues Ticket anzufordern.

Der Aufruf der Funktion „GetTicket“ erfolgt über: **https://tourist.bz.it/WSecure/WSecure.aspx;**

Bitte beachten Sie hierbei das **HTTPS.**

Für das Login muss die Webserverfunktion „GetTicket“ benutzt werden.

Finden in einem kurzen Abstand mehrere fehlerhafte Logins statt, wird das Login für einige Minuten gesperrt, und öffnet sich danach automatisch wieder.

Beachten Sie, dass mit einem Ticket nur die Daten des jeweiligen Betriebs verarbeitet werden.

### GetTicket

User	Web-UserID des Betriebs (beim TV erhältlich)
Pwd	Web-Passwort des Betriebs (beim TV erhältlich)
PrefUserID	Bevorzugte Rechtegruppe (beim LTS erhältlich)
LngID	Sprache
RID	RID des Objekts
Type	Objekttype („PAR“=Betrieb; „POR“=Sicht; „TCC“=Büro)
Time	Gültigkeit des Tickets in Sekunden

### Abfrage:

```
z.B.  <SendData User='1234567895AE47F8AF808C64C7631470'
      Pwd='123456789AFE476F95264AA06F3C2F47' Function='GetTicket' Version='1.0'
      PrefUserID='12345686E4CD478CAFB562E7CE937B9B' />
```

### Antwortdaten:

```
z.B.  <RespData >
      <Ticket
        A0RID='123456074E2C11D289320060979ACE62'
        RID='123456074E2C11D289320060979ACE62'
        SessionID='12345686E4CD478CAFB562E7CE937B9B'
        UserID='1234567999B045F994D2DCF458605B70'
        LngID='DE'
        Time='60'
        Type='PAR'
      />
    </RespData>
```

\*\*\* Hinweis: A0RID wird nicht mehr unterstützt. Es muss stattdessen das Attribut RID benutzt werden. Geändert am 2006-03-28.



## Check der Gültigkeit eines Tickets

Mit der Funktion „CheckTicket“ kann ein Ticket auf seine Gültigkeit überprüft werden

Der Aufruf der Funktion „GetTicket“ erfolgt über: <https://tourist.bz.it/WSecure/WSTicket.aspx>  
Bitte beachten Sie hierbei das **HTTPS**.

### CheckTicket

#### Abfrage:

```
z.B.  <SendData UserID='...'
      SessionID='...'
      Function='CheckTicket' Version='1.0' >
      <Head UserID='...' SessionID='...' IP='...'
      RID='F998BD86E4CD478CAFB562E7CE937B9B' />
    </SendData>
```

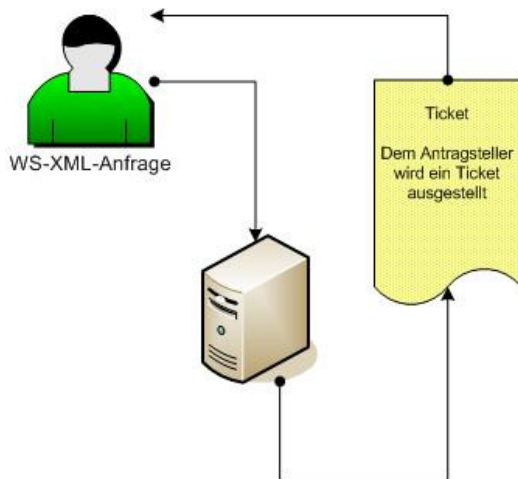
#### Antwortdaten:

```
z.B.  <RespData >
      <Ticket Value='1' />
    </RespData>
```

Value: 1 = Ticket gültig, 0 = Ticket ungültig

SendData.UserID	= Eigene UserID
SendData.SessionID	= Eigene SessionID wenn vorhanden
Head.UserID	= Zu prüfende UserID wenn vorhanden (bleibt beim Prüfen eines Tickets normalerweise leer)
Head.SessionID	= Zu prüfende SessionID wenn vorhanden (bleibt leer wenn mit UserID gearbeitet wird, ansonsten SessionID aus Ticket)
Head.IP	= Zu prüfende SessionID wenn vorhanden (bleibt beim Prüfen eines Tickets normalerweise leer)
Head.RID	= RID des Besitzers

## 1. Erlangen eines Tickets



## 2. Funktion ausführen

