



ELIS 3.9

WCF Schnittstellenbeschreibung

Einsatzleitinformationssystem



Einsatzleit- & Informationssystem



1 ELIS WCF Struktur

ELISWCF ist als Modul von ELISService ausgeführt. Externe Anwendungen außerhalb von ELIS Service können die ELISWCF Schnittstelle nutzen. Diesen verschiedenen Client-Programmen wird dabei die .NET DLL **ELISWCFClient** zur Verfügung gestellt, welche die Funktionalität zur Verfügung stellt.

ELIS WCF ist eine .NET 3.0 WCF Schnittstelle, welche über das netTcp Protokoll Funktionen zur Verfügung stellt.

Über die ELIS WCF Schnittstelle können alle Daten von ELIS geholt und bearbeitet werden.

2 ELIS WCF verwenden

Um die WCF Funktionen zu nutzen, muss ein ELISWCFClient angelegt werden:

```
using ELIS.ELISWCF;

internal ELISWCFServiceClient WCFServiceClient
{
    get
    {
        ELISWCFClientFunctions cs = new
ELISWCFClientFunctions("URL_ELISServer", "PORT_ELISServer");
        bool isSecured=false;
        serviceClient = cs.getWCFServiceClient(ref isSecured);

        return this.serviceClient;
    }
}
```

3 Authentifizierung

Um ELISWCF Funktionen aufrufen zu können, muss sich der Client zuvor beim ELISServer anmelden. Nach der Anmeldung bekommt der Client ein Ticket, welches bei jedem ELISWCF Funktionsaufruf mitgegeben werden muss.

3.1 Benutzeranmeldung über WCF

Ein neues Ticket kann mit der *CreateTicket* Funktion angefordert werden. Rückgabewert ist ein Ticket oder gegeben falls Fehlermeldungen (NOUSER, USERINACTIVE, ERROR).

```
ELISWCFClientFunctions.getTicket("ELISLogin", "ELISPasswort");

public WCFTicket Ticket
{
    get
    {
        ELISWCFClientFunctions cs = new
ELISWCFClientFunctions("URL_ELISServer", "PORT_ELISServer");
        WCFTicket _ticket = cs.getTicket("ELISLogin", "ELISPasswort");
        return _ticket;
    }
}
```

3.2 Ticket-System

Um eine WCF-Funktion nutzen zu können, muss vorher ein Ticket vom ELISWCFService angefordert werden, welches bei jedem WCF-Funktionsaufruf mitgegeben werden muss (TicketID). So kann jeder



Funktionsaufruf einer Person zugeordnet werden und weiters kann mit Hilfe des Tickets auch kontrolliert werden, ob der Benutzer die Berechtigungen zum Aufruf einer Funktion besitzt.

Gültigkeit eines Tickets

Ein Ticket ist ab der letzten Verwendung genau 30 Minuten lang gültig. Jede Verwendung (=WCF-Funktionsaufruf mit Ticket) erneuert das Ticket automatisch. Ein aktives Ticket kann auch per Funktionsaufruf ungültig gemacht werden.

4 Logging

Alle Aufrufe der WCF Schnittstelle werden protokolliert in der System-Datenbank.

Tabelle WCFLog

- Int ID
- Guid TicketID
- String FunktionsName
- DateTime Zeit
- String Informationen

5 Eigene WCF Bindings

Für .NET WCF Spezialisten ist es auch möglich eigene Bindings zu definieren:

5.1 ELIS Service

Der ELIS Service erstellt automatisch einen vorkonfigurierten Endpoint („netTcp“ Binding) auf Port 9000. Durch manuelle Konfiguration in der ELISService.exe.config können aber beliebig viele zusätzliche WCF Endpoints mit eigenen Bindings definiert werden.

```
<system.serviceModel>
  <behavior>
    <serviceBehaviors>
      <behavior name="ELIS.ELISWCF.ELISWCFServiceBehavior">
        <serviceMetadata httpGetEnabled="true"/>
        <serviceDebug includeExceptionDetailInFaults="true" />
        <serviceCredentials>
          <serviceCertificate findValue="kdope"
            storeLocation="LocalMachine"
            x509FindType="FindBySubjectName" />
        </serviceCredentials>
      </behavior>
    </serviceBehaviors>
  </behavior>
  <services>
    <service name="ELIS.ELISWCF.ELISWCFService" behaviorConfiguration="ELIS.ELISWCF.ELISWCFServiceBehavior">
      <endpoint address="http://127.0.0.1:9001/ELISWCFService/mex"
        binding="mexHttpBinding"
        name="ELISWCFService_MEX"
        contract="IMetadataExchange" />
      <endpoint address="http://127.0.0.1:8000/Design_Time_Addresses/ELISWCFService"
        binding="netTcpBinding"
        name="ELISWCFService"
        contract="ELISWCFServiceContract" />
    </service>
  </services>
  <bindings>
    <netTcpBinding>
      <binding name="netTcpBinding"
        maxReceivedMessageSize="65536"
        maxBufferSize="65536"
        maxConnectionPoolSize="10"
        maxIdleObjectLifetime="00:01:00"
        receiveTimeout="00:00:30"
        sendTimeout="00:00:30"
        security="None" />
    </netTcpBinding>
  </bindings>

```



```

</netTcpBinding>
</wsHttpBinding>
<binding name="mexHttpBinding" transactionFlow="true">
  <reliableSession enabled="false"/>
</binding>
</wsHttpBinding>
</bindings>
</system.serviceModel>

```

Alle konfigurierten Endpoints werden beim Start von ELIS Service ausgegeben:

```

14.08.2013 21:55:55\ELISWCF: ELISWCF Service gestartet!
14.08.2013 21:55:55\ELISWCF: ELISWCF Web Service Definition: http://127.0.0.1:8000/Design\_Time\_Addresses/ELISWCFService?WSDL
14.08.2013 21:55:55\ELISWCF: ELISWCF Service Host: http://127.0.0.1:9001/ELISWCFService/mex
14.08.2013 21:55:55\ELISWCF: ELISWCF Service Host: net.tcp://10.0.0.130:9000/ELISWCFService
14.08.2013 21:55:55\ELISWCF: ELISREST Service gestartet! Host(http://10.0.0.130:9002)

```

5.2 ELIS Clients

Um die zusätzlich definierten Endpoints in den ELIS Clients (Verwaltung/WebPortal/Visualisierung) nutzen zu können, muss auch die jeweilige Client Konfiguration angepasst werden.

Es wird dabei automatisch bei jedem ELIS Client Start nach einer definierten Bindung in der jeweiligen ELIS Client Konfigurationsdatei gesucht mit dem Namen „ELISWCFtcpBindingCustom“. Wird keine eigene Bindungskonfiguration gefunden verwendet der ELIS Client die vorkonfigurierte Bindung.

Beispiel einer Anpassung einer ELIS Verwaltung Konfiguration:

```

<?xml version="1.0"?>
<configuration>
  <configSections>
  </configSections>
  </configSections>
  <appSettings>
    <add key="ELISServerURL" value="127.0.0.1:9000"/>
    <add key="ELISBenutzerName" value=""/>
    <add key="LogPfad" value="C:\Programme\Feuerwehr\ELIS Verwaltung\ELISVerwaltungLog.txt"/>
  </appSettings>
  <bindings>
    <binding name="ELISWCFtcpBindingCustom" type="System.ServiceModel.HttpBinding" >
      <endpoint address="http://127.0.0.1:9001/ELISWCFService/mex" >
        <binding name="mexHttpBinding" transactionFlow="true">
          <reliableSession enabled="false"/>
        </binding>
      </endpoint>
    </binding>
  </bindings>
  <startup>
    <supportedRuntime version="v2.0.50727"/>
  </startup>
</configuration>

```

6 Alle Funktionen im Überblick

Auflistung der WCF Funktionen

Die aktuelle ELIS WCF API ist zu finden unter:

<http://www.steinbichl.com/ELISWCFApi>



7 Impressum

„ELIS 3.9“ ist ein Programm für Freiwillige Feuerwehren in Oberösterreich. Die Software wird über das Downloadportal (www.steinbichl.com) zur Verfügung gestellt.

Autor:



Wolfgang Steinbichl MSc.

FF Puchenua

Mail: Wolfgang.steinbichl@ff-puchenua.at