

A secure data transmitter component for the SwissDec
ELM/SalaryDeclaration V3 system

SwissDecTX 3.09

Our first customer has passed all certification tests for his salary application on June 26, 2009 and obtained the official SwissDec certification on September 7, 2009. Since then, ten more applications passed the certification tests across all regions of the country (total 2 in the Swiss-German part, 7 in Romandie and 2 in Ticino). Our last two customers have now passed their SwissDec v3 certification.

The SwissDecTX transmitter satisfies all SwissDec technical transmission requirements including data validation, digital signature, encryption, interoperability tests, transmission logging, as well as rich response formatting to HTML! SwissDecTX is 100% ready for production with SwissDec ELM v3.

Goals

- ◆ Makes salary data transmission (*much*) easier
- ◆ Implement all SwissDec security requirements (digital signatures, encryption, error handling, result formatting...)
- ◆ Easily accessible (any language or environment)
- ◆ Fully “SwissDec certifiable” – guaranteed!
- ◆ Easy integration with existing applications: DLL, COM, .NET, scripting, command-line..., Delphi, PROGRESS, VB6... we’ve done all this already!

What SwissDecTX is not

- ◆ Not a payroll application
- ◆ Not an end-user program
- ◆ Not an XML salary data generator



You create the data (e.g. ICHAGCompany.xml)
SwissDecTX sends it to the Web Service,
using SwissDec protocols and strong security.

What is SwissDecTX?

- ◆ Secure SwissDec salary data transmitter for...
 - ◆ .NET managed code (C#, VB.NET,...)
 - ◆ COM (interoperability with VB6, Delphi, PROGRESS...)
 - ◆ Classic DLL binding (C, C++, WINDEV,...)
 - ◆ Command-line interface (any dev. environment)
- ◆ Takes care of XML validation (XSD schemas)
- ◆ Takes care of all transmission technical issues
- ◆ Optional disk-based transmission log
- ◆ Point and click GUI configuration tool
- ◆ Command-line utility, XML Helpers, XSLT formatting

Communication

- ◆ SOAP + WS-Security
- ◆ Supports HTTP and HTTPS (with authentication)
- ◆ Supports HTTP Proxy (with proxy authentication)
- ◆ X509 client and server certificates
- ◆ Message signing and time stamping
- ◆ Message encryption
- ◆ Verification of replies (signatures, timestamps)
- ◆ “EIV” file generation, raw SOAP messages & replies

Configuration

- ◆ GUI configuration tool
 - ◆ Certificate installer / selector
 - ◆ Web Service URL configuration, quick interop test (suitable for the certification), transmission tests, optional log
 - ◆ Optional encryption option
- ◆ Settings saved to registry
 - ◆ Opportunity to configure at setup time
 - ◆ Most settings available programmatically
- ◆ Command-line tool supports certificate installation

Configuration & Test Tool

The screenshot shows a Windows-style application window titled "SwissDecTX Transmitter 1.13 - Configuration & Test Tool". It features a tabbed interface with the following tabs: Configuration, Interoperability, Transmission, Async. Transmission, EIV Generator, Transmission Log, and About. The "Configuration" tab is currently selected. The configuration fields are as follows:

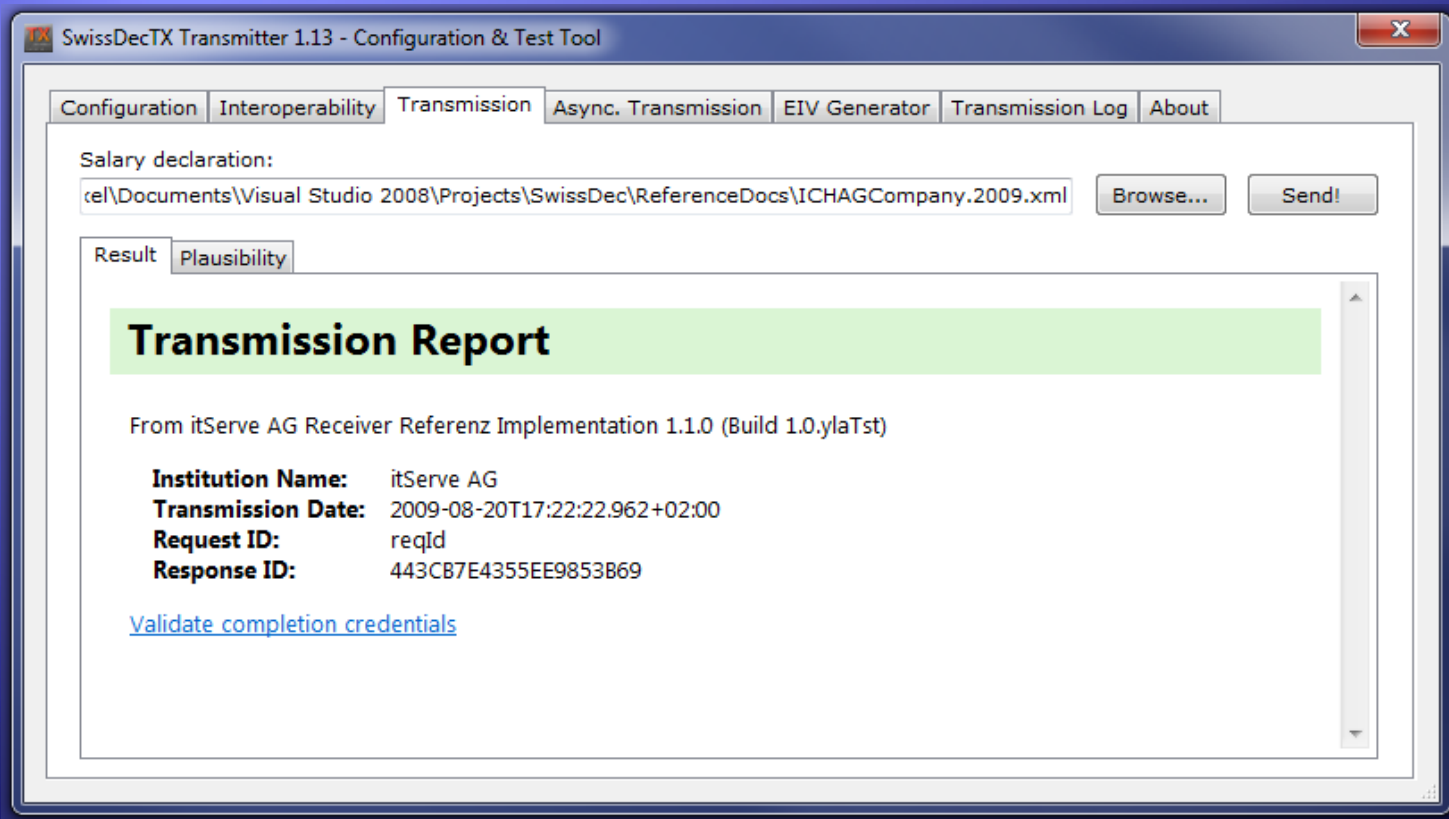
- Web Service URL:** A text box containing the URL `https://tst.itserve.ch/itserve/lohnstandard/testing/refapps2/receiver-webservice/20051002-0.0/SalaryDeclarationService`.
- Monitoring ID:** A text box containing the value `tester`.
- SwissDec Root CA Certificate (obtained from SwissDec):** A dropdown menu showing `CN=Swissdec, C=CH`.
- Client Certificate (Transmitter) obtained from SwissDec:** A dropdown menu showing `SERIALNUMBER=3, CN=Test, O=Swissdec, C=CH`.
- Server Certificate (Receiver) obtained from SwissDec:** A dropdown menu showing `SERIALNUMBER=4, CN=RefApp Receiver, O=Swissdec, C=CH`.
- SwissDecTX Transmitter License:** A dropdown menu showing the file path `C:\Users\Axel\Documents\Visual Studio 2008\Projects\SwissDec\Issued Licenses\SwissDecTX.Demo.lic`.
- Encrypt outgoing communications:** An unchecked checkbox.
- Quick Test:** A button located at the bottom left of the configuration area.

Test: Interoperability

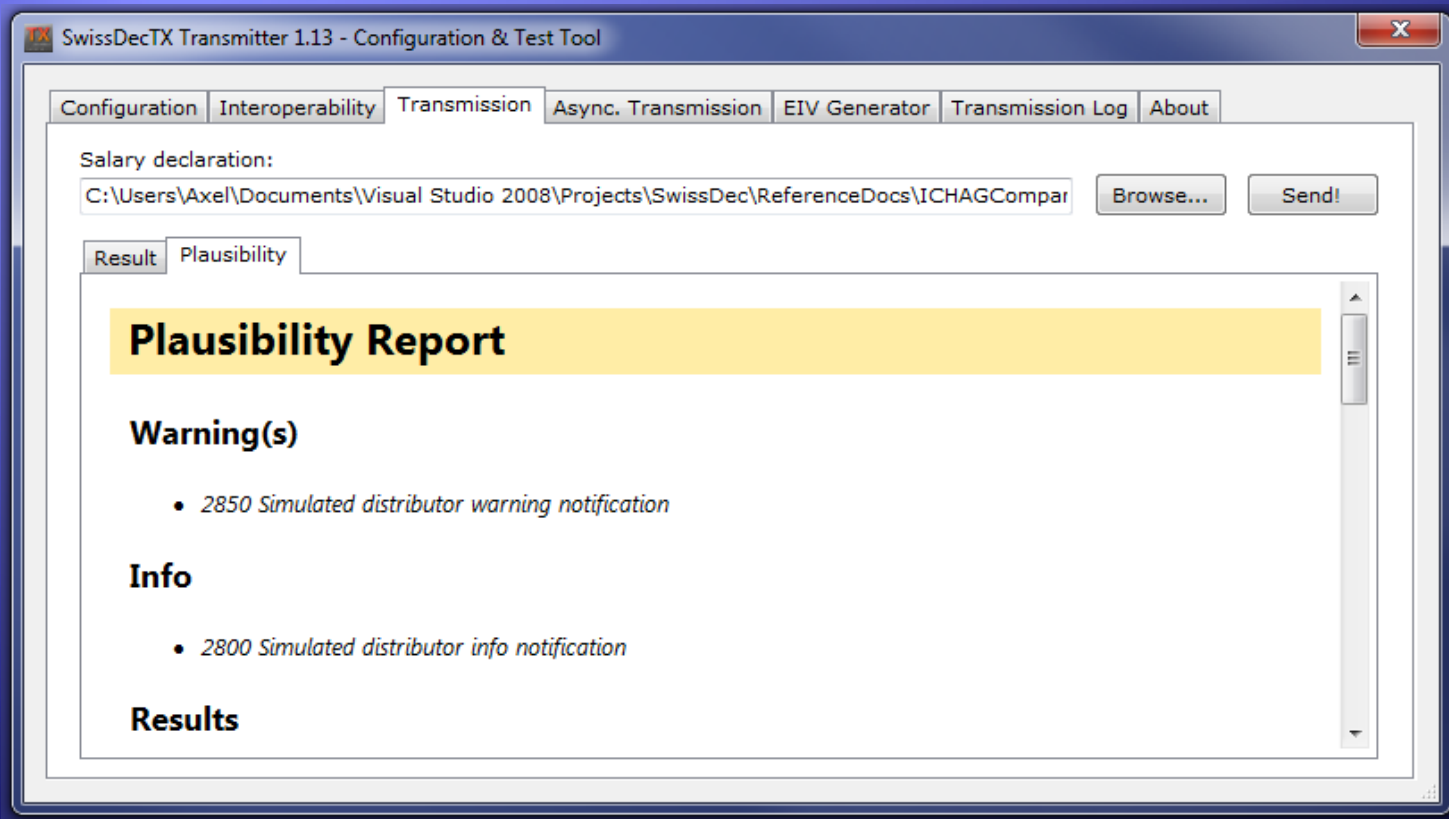
The screenshot shows a software window titled "SwissDecTX Transmitter 1.13 - Configuration & Test Tool". It has a tabbed interface with the following tabs: Configuration, Interoperability (selected), Transmission, Async. Transmission, EIV Generator, Transmission Log, and About. The Interoperability tab contains the following fields and controls:

Input Fields	Buttons	Output Fields	Result
First operand: 999000000000	Test!	Addition result: 999000000001.00	Result The test succeeded and all return values checked OK. The server's date/time is 20.08.2009 17:12:55 and is 0'8" away from the local computer's time.
Second operand: 1.00		Subtraction result: 998999999999.00	
Umlaut string: ÄËÖÜÄËÖÜÄËÖÜÄËÖÜ		Returned string: äëöüäëöüäëöüäëöü	

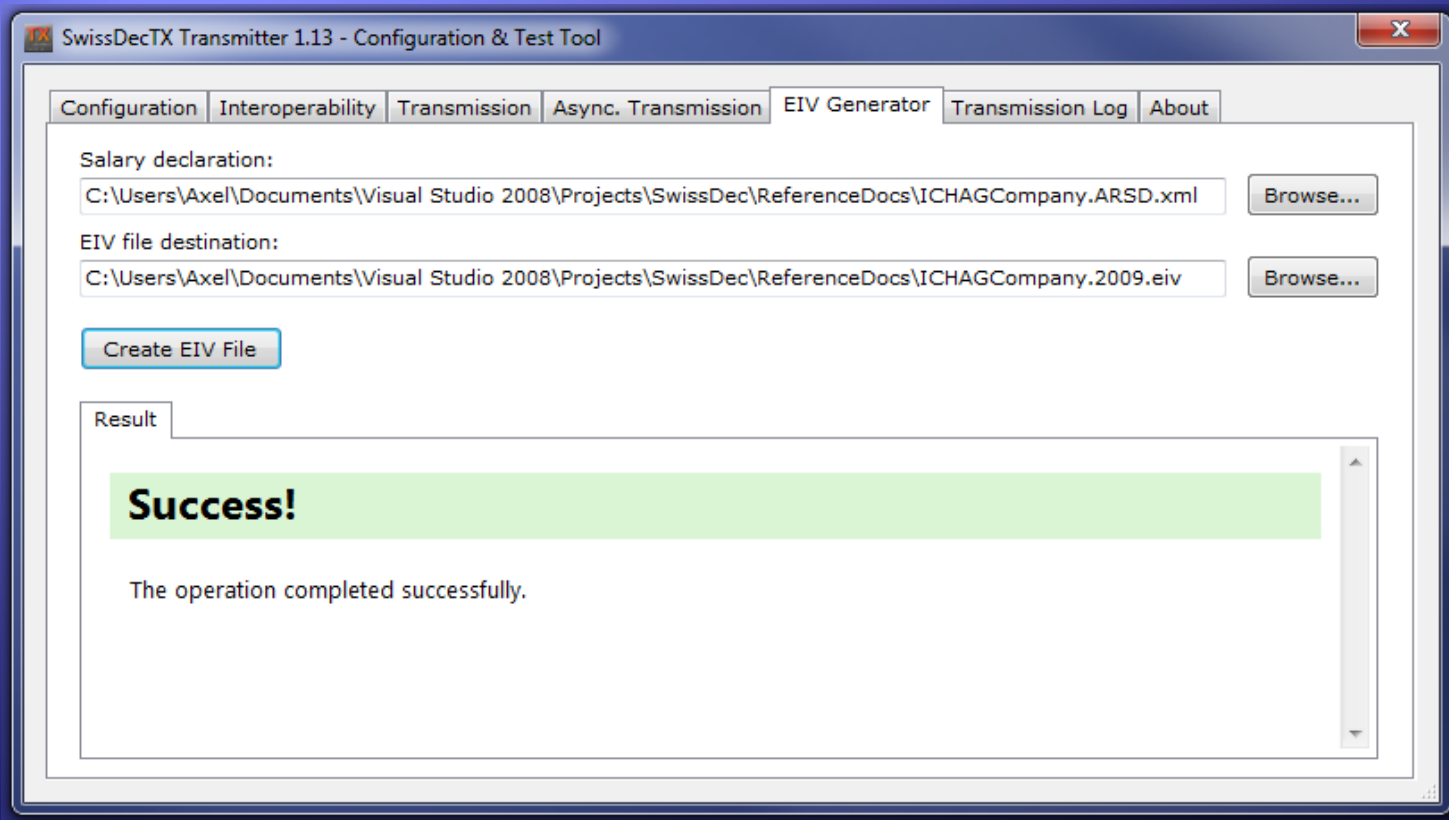
Test: Transmission



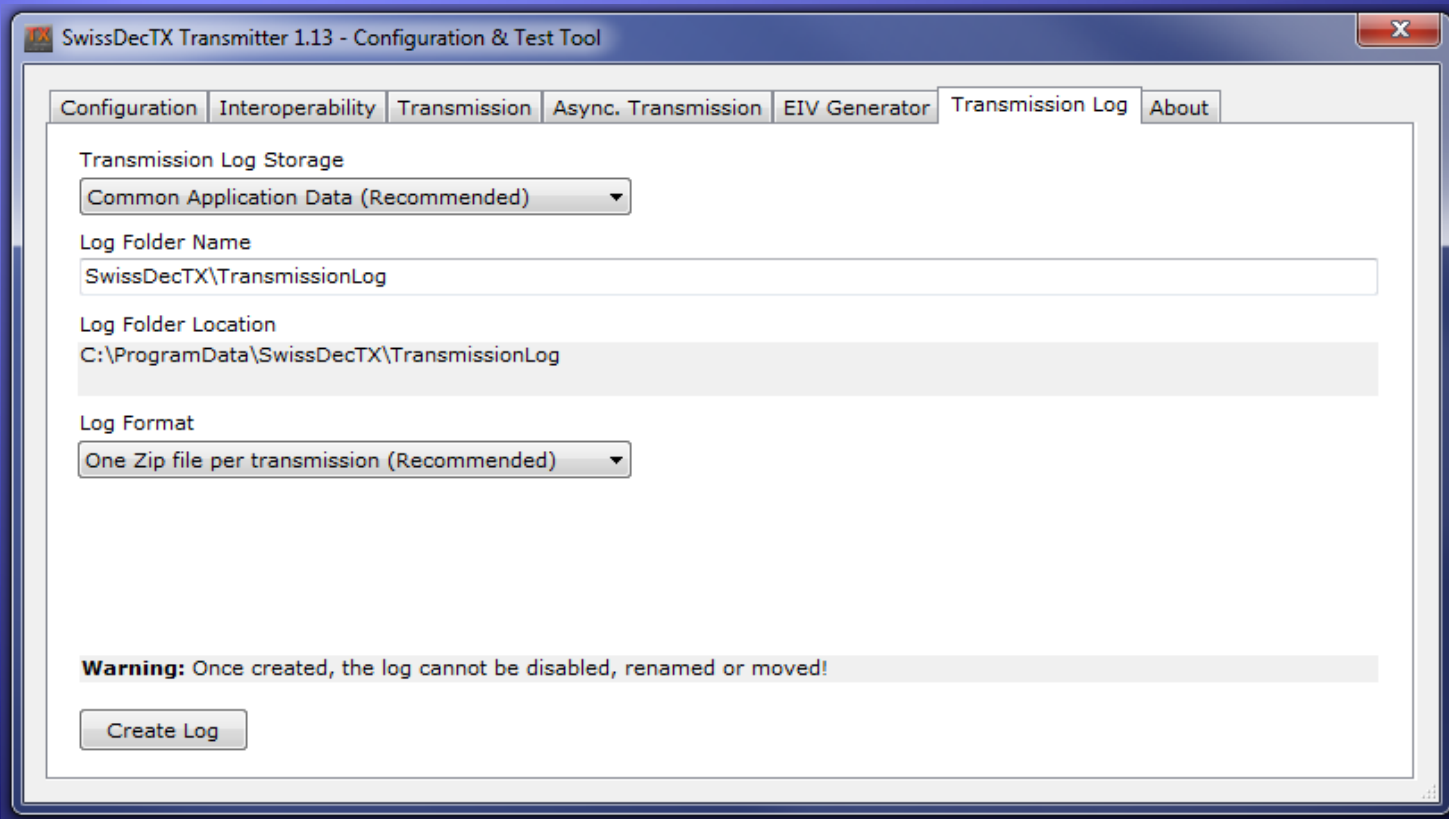
Test: Plausibility Report



Test: EIV File Creation



Transmission Log Config



SwissDecTX Transmitter 1.13 - Configuration & Test Tool

Configuration Interoperability Transmission Async. Transmission EIV Generator **Transmission Log** About

Transmission Log Storage
Common Application Data (Recommended) ▼

Log Folder Name
SwissDecTX\TransmissionLog

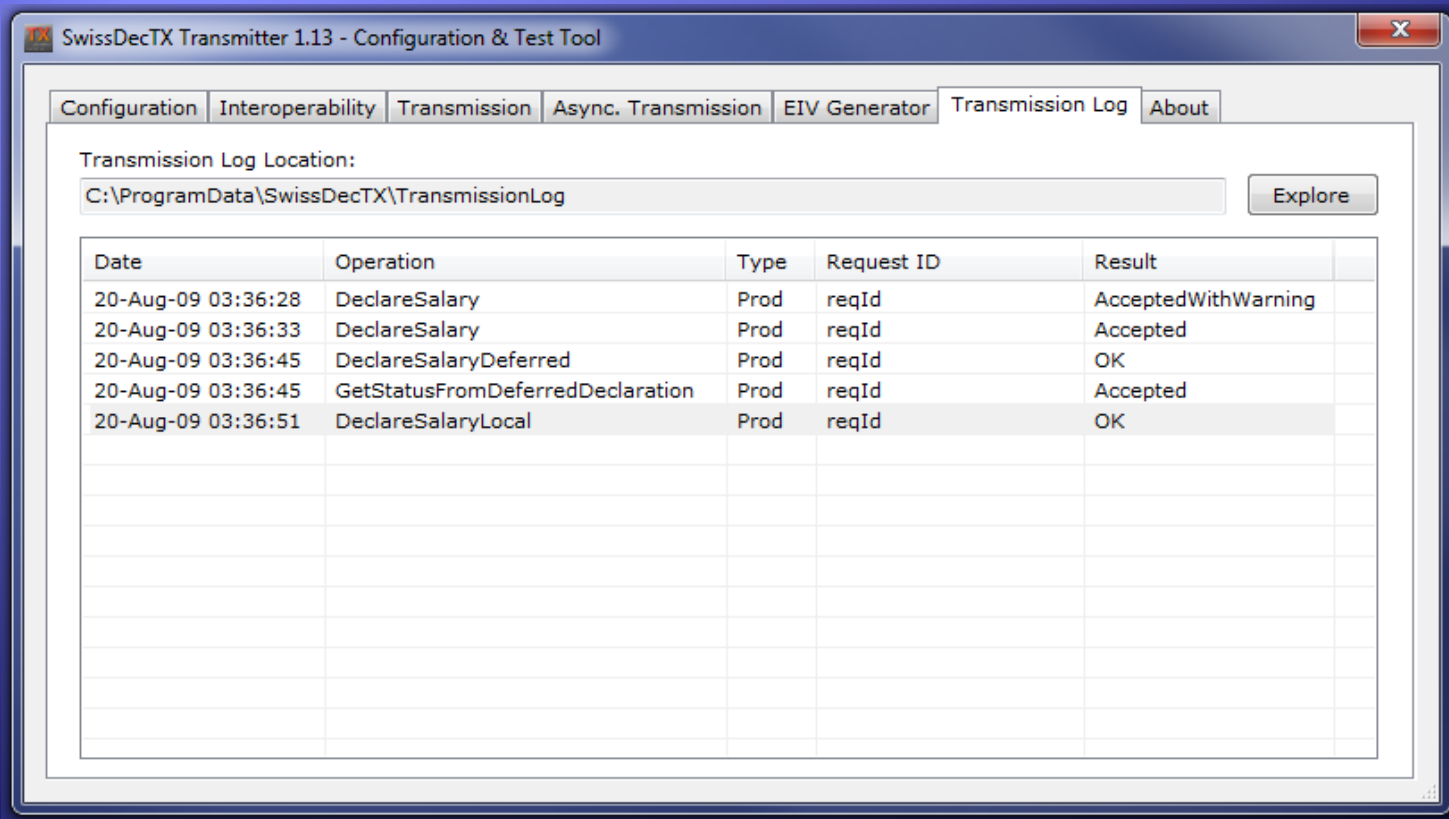
Log Folder Location
C:\ProgramData\SwissDecTX\TransmissionLog

Log Format
One Zip file per transmission (Recommended) ▼

Warning: Once created, the log cannot be disabled, renamed or moved!

Create Log

Transmission Log



Installation

- ◆ Standalone silent installer (.exe)
- ◆ Pulls pre-requisites from web (DotNetFX, WSE)
- ◆ Transmitter assemblies goes to GAC
- ◆ DLL wrapper, command-line tool and configuration tool goes to \Program Files\SwissDecTX
- ◆ Installation location is fixed
- ◆ Tiny package (~ 1.3MB) to bundle with your product and run once at install time
- ◆ Depends only on .NET 2.0, works on .NET 3.x / 4.x

System Requirements

- ◆ Same as .NET 2.0 SP1
- ◆ Baseline: Windows 2000 SP4 or later
- ◆ Windows XP, Vista, Windows 7, Server versions supported, 64-bit edition available upon request
- ◆ Only depends on .NET 2.0 SP1, no need to push .NET 3.x /4.x down to XP/W2K/Vista clients
- ◆ Clients can be *anything* (native, COM, .NET, ...)

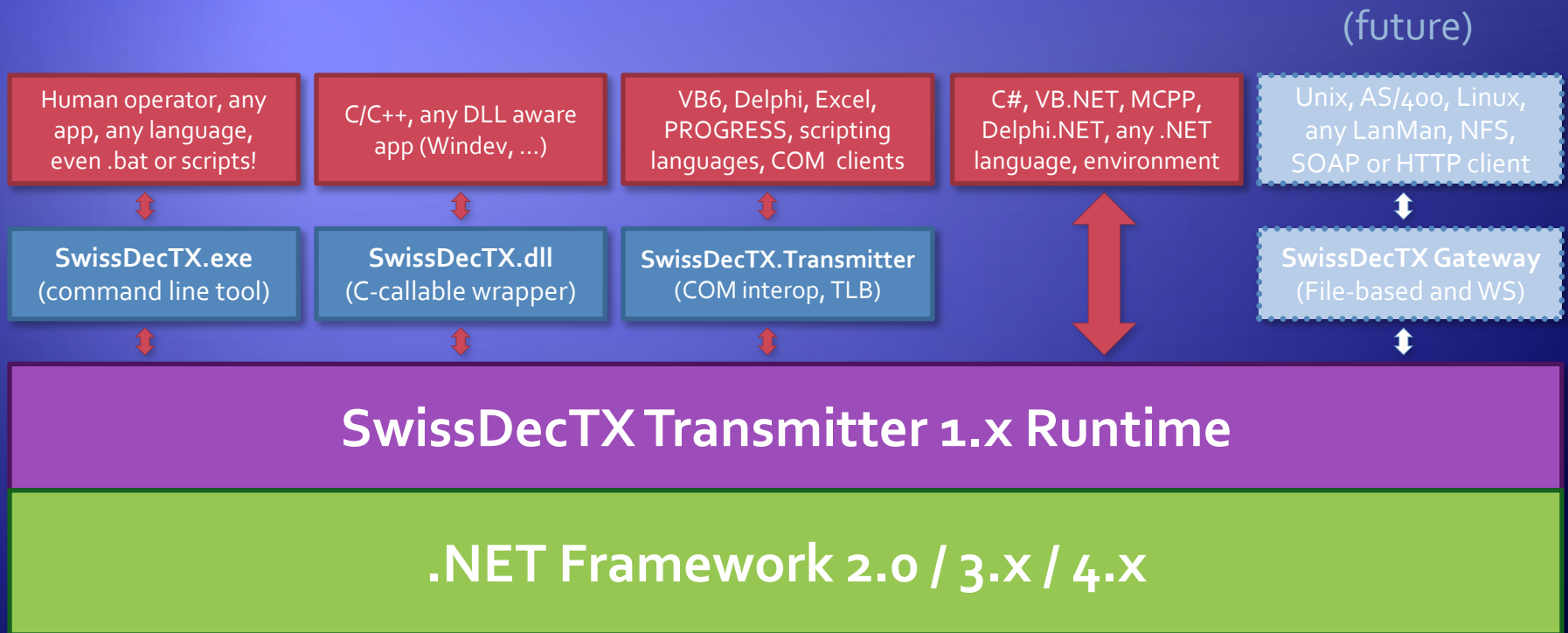
How to use it?

- ◆ Application creates XML declaration data
- ◆ Application invokes transmitter
- ◆ Transmitter signs/encrypts/transmits data
- ◆ Transmitter decrypts/checks/saves replies
- ◆ Application reads and displays results and server replies, maintains/display transmission journal
- ◆ Just a few lines of code to transmit/receive!

Transmit a Salary Declaration

```
try {  
  
    var tx = new Transmitter();  
  
    // Declare variables, create/load declaration data..  
  
    tx.DeclareSalary(xmlDoc, out result, out messageSent, out  
        messageReceived, out plausibility, out jobFinished,  
        out transmissionDateUtc);  
  
    // save / interpret results  
  
} catch (Exception ex) {  
    // Handle all errors here  
}
```

Diagram



Development kit

- ◆ Transmitter API is fully documented (CHM file)
- ◆ Small code samples (C++, C# and VBS)
- ◆ **Integrate and start transmitting in a day or less!**
- ◆ Current state (07.2012)
 - ◆ Stable transmitter, in production, fully tested over more than 12'000 secure transactions on the test servers
 - ◆ Eight customer applications **certified** since Sept 7, 2009
 - ◆ Available now (**get the demo and transmit today!**)
 - ◆ SDK, language bindings, samples and documentation still evolving, optional “gateway” in development, XML Helpers

Licensing, maintenance

- ◆ Licensed “per application” regardless of number of developers or application copies sold
- ◆ **No redistribution royalties** for runtime components: flat fee pricing model.
- ◆ Maintenance and support plan available
- ◆ **Free fully functional demo** (locked to test servers)
- ◆ (Upcoming optional SwissDecTX Gateway will be licensed “per installation”, more details TBA)

Future

- ◆ SwissDecTX v3.03, compatible with SwissDec v3 specifications, **is already available** for you to prepare for SwissDec v3 certification.
- ◆ We'll track future SwissDec specification updates and provide timely support for them.

About Us

- ◆ Extensive development experience, ranging from device drivers to internet-scale distributed systems
- ◆ Experience in creating and supporting custom APIs and frameworks for other developer's use
- ◆ Strongest C/C++/COM and Windows platform skills
- ◆ Strong .NET experience (since 1.0 beta1 in 2000!)
- ◆ Experience in working with small and large companies and teams

Call to action: contact us now!

Axel Rietschin Software Developments

36, Rue de Carouge

1205 Genève

(022) 321.70.22 (English and French spoken)

info@softwarecomponents.ch

<http://www.softwarecomponents.ch>