

Integrating Real-time Predictive Analytics into SAP Applications



Applies to:

For more information, visit the [Analytics homepage](#).

Summary

Demonstrates the necessary steps on how to integrate an external real-time prediction engine into an SAP application exit.

Author: Albrecht Weiss

Company: SAP AG

Created on: 30 December 2009

Table of Contents

Introduction	3
Define & Train a Predictive Model	4
Converting PMML 3.1 to PMML 3.2	4
Deploy the PMML Model	5
Download the WSDL file.....	6
Setting up the Security Configuration	6
Proxy Configuration	7
Creating a Client Proxy based on the WSDL file.....	7
Creating a Logical Port	8
Creating ABAP Code to Call the Client Proxy in the SAP Application Exit	9
Copyright.....	10

Introduction

Supporting our customers in their quest to implement the most efficient business processes is one of the core values of SAP. As more and more businesses recognize the value that predictive analytics delivers, they seek solutions which seamlessly integrate intelligent decisions and leverage complex statistical models as central assets in their operational systems.

The idea for the implementation described in this whitepaper arose when we were building a prototype to demonstrate the feasibility of integrating real-time scoring into Tax and Revenue Compliance management processes as well as for the optimization of collection strategies of tax agencies. In this article, we do not elaborate on the various SAP applications which will be able to benefit from an integration of on-demand predictive analytics, but rather showcase a real world, step-by-step example linking an external real-time scoring service into an SAP application exit.

SAP Business Objects has an advanced data mining tool in its portfolio called SAP Business Objects Predictive Workbench. This workbench provides an extensive feature set to build predictive models, however, it does not offer an out-of-the-box integration with an SAP ABAP-based application exit.

Bridging the gap between data mining and model building on one side and scalable production deployment of predictive models on the other side is the Predictive Model Markup Language (PMML) standard. The SAP BusinessObjects Predictive Workbench product supports the PMML standard which enables us to easily export our models as a PMML file and then leverage other vendors for the integration and real-time execution of the models within operational systems.

After review of the market for predictive analytics deployment solutions which support secure integration via service oriented architecture (SOA), we selected the ADAPA predictive analytics engine by Zementis (<http://www.zementis.com>) for a proof of concept. ADAPA provides a model deployment and execution environment, either as a service on the Amazon Elastic Compute Cloud (Amazon EC2) or in-house.

Another proprietary solution which achieves similar results based on the streams created in the SAP BusinessObjects Predictive Workbench is IBM SPSS Collaborative and Deployment Services (<http://www.spss.com>).

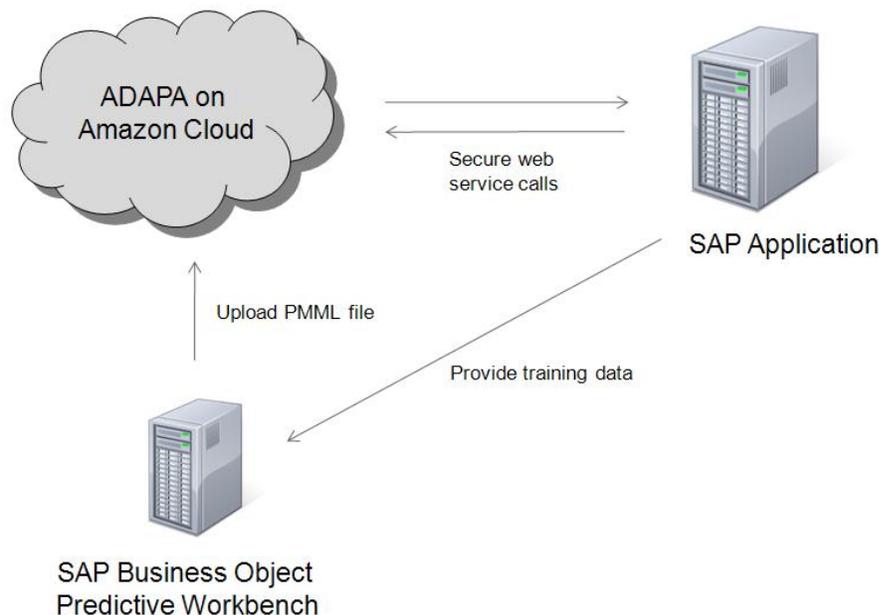


Figure 1: General overview of SAP/ADAPA integration

The following sections will outline a step-by-step approach to develop, deploy, integrate and execute predictive models based on SAP and ADAPA.

Define & Train a Predictive Model

First we need to create and train a predictive model using a data mining tool. In our example, we choose the SAP BusinessObjects Predictive Workbench.

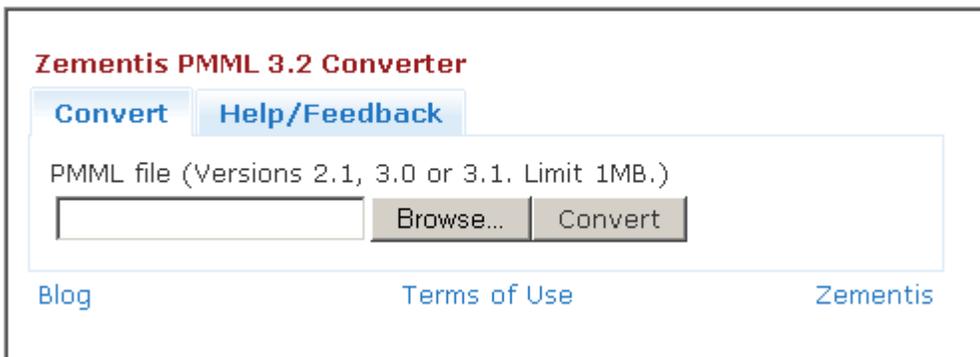
The trained model is exported as a file in the file format PMML. For more information about the Predictive Model Markup Language standard, please visit the Data Mining Group web site at <http://www.dmg.org>. PMML is supported by all major data mining vendors, commercial and open source. Therefore, you may also use various other tools to develop your predictive model.

Converting PMML 3.1 to PMML 3.2

Various vendors support different versions of the PMML standard. In our example, SAP BusinessObjects Predictive Workbench produces PMML 3.1 format. Depending on the version exported by your data mining application, you may need to convert the PMML file to PMML 3.2 before being able to deploy it in the ADAPA engine.

As a free tool to convert, correct and validate PMML models, Zementis provides a PMML Converter on its web site at <http://www.zementis.com/pmml.htm>

PMML Converter



The screenshot shows the Zementis PMML 3.2 Converter web interface. At the top, there are two buttons: "Convert" and "Help/Feedback". Below these is a text input field labeled "PMML file (Versions 2.1, 3.0 or 3.1. Limit 1MB.)". To the right of the input field are two buttons: "Browse..." and "Convert". At the bottom of the interface, there are three links: "Blog", "Terms of Use", and "Zementis".

Figure 2: Zementis PMML Converter converts, validates and corrects various PMML files.

Deploy the PMML Model

Once you have a predictive model in a PMML file, the next step is to deploy this model in the ADAPA engine. In our example, ADAPA is used as a private, secure instance on the Amazon Elastic Compute Cloud (Amazon EC2).

Log into the ADAPA Management Console, select your PMML file and upload the model(s). It instantly becomes available for interactive scoring as well as via web service API.

The screenshot shows the ADAPA Management Console interface. The browser title is "ADAPA Predictive Analytics Edition 2.18 - Microsoft Internet Explorer provided by SAP IT". The address bar shows the URL: <https://ec2-174-129-57-155.compute-1.amazonaws.com/adapaconsole/login.jsf;jsessionid=AA51F41F8DF5B55ACA88992E5514F0DC>. The page header includes the ZEMENTIS logo, the version "ADAPA Predictive Analytics Edition 2.18", and links for "Change Password", "Help", and "Logout". There are three tabs: "Predictive Models" (selected), "Rule Sets", and "Reports".

The main content area is titled "Manage Models" and contains a "New Model Upload" section. It features a text input field labeled "PMML 3.2 File" with a "Browse..." button to its right, and an "Upload Model" button below it.

Below the upload section is the "Available Models" section, which contains a table with the following data:

Name	Actions	Description	Creation Date
2008	 		
TreeModel-2009-12-01-04-06-24-832	 		

At the bottom of the "Available Models" section, there is a link for "RPC Web Service Description (WSDL)" with a globe icon.

Figure 3: ADAPA Management Console to deploy predictive models.

Download the WSDL file

After uploading the model, the ADAPA Console provides a WSDL file (see Figure above) which can be downloaded containing all the input and output parameters of the predictive model. In our example, we use the RPC style web service for single decision scoring. As outlined in the Help Section of the Console, ADAPA also offers additional web service options, e.g., for bulk scoring, scoring against multiple models, as well as automated deployment of models.

Setting up the Security Configuration

As the ADAPA cloud web service runs on the SSL stack, you first need to tell the SAP system to trust the ADAPA server on the cloud.

This is achieved by downloading the security certificate of your ADAPA cloud service and add this into the certificate store of SAP (transaction: sstrust2)

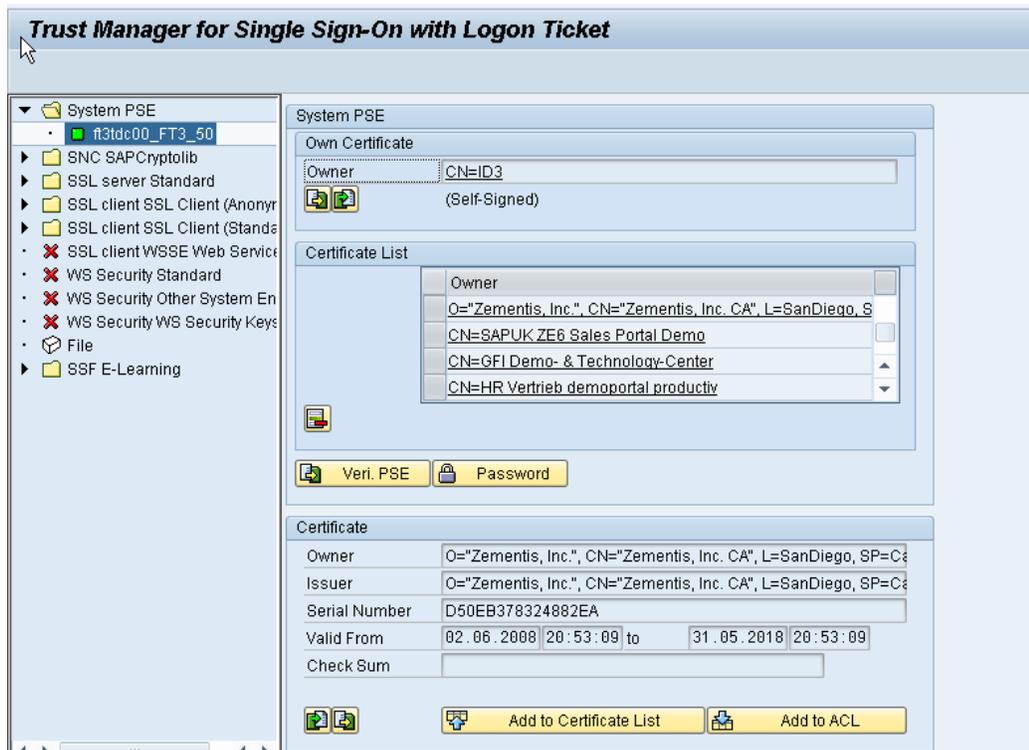


Figure 4: SAP Trust Manager

Click on Import in the Certificate section and click on “Add to Certificate List”. The new certificate will appear in the “Certificate List” section.

Proxy Configuration

Depending on the use of internet proxies, you may have to configure the SAP system with your company's proxy configuration.

(SAP Transaction: SICF -> click execute -> navigate via Menu: -> Client -> Proxy Settings)

Creating a Client Proxy based on the WSDL file

The SAP Object Navigator allows for an easy creation of client proxy objects based on a WSDL file.

- Transaction SE80 -> Edit Object -> Enterprise Services-> Client Proxy

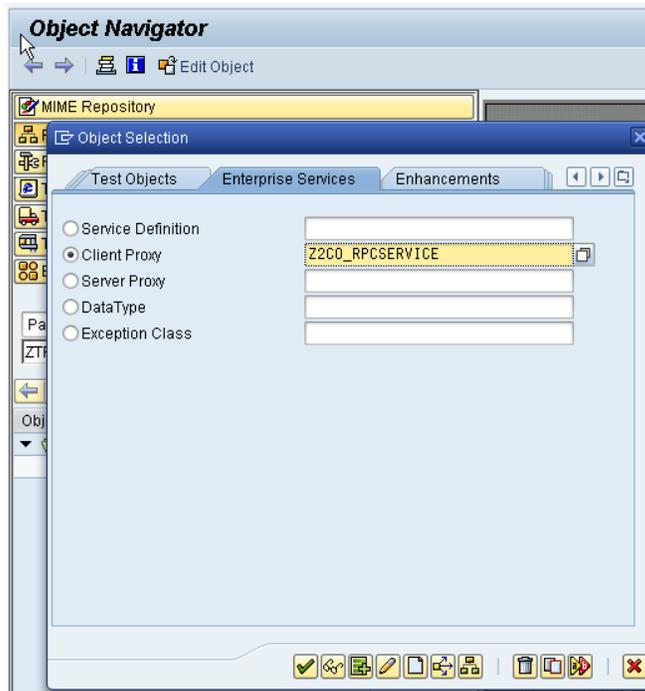


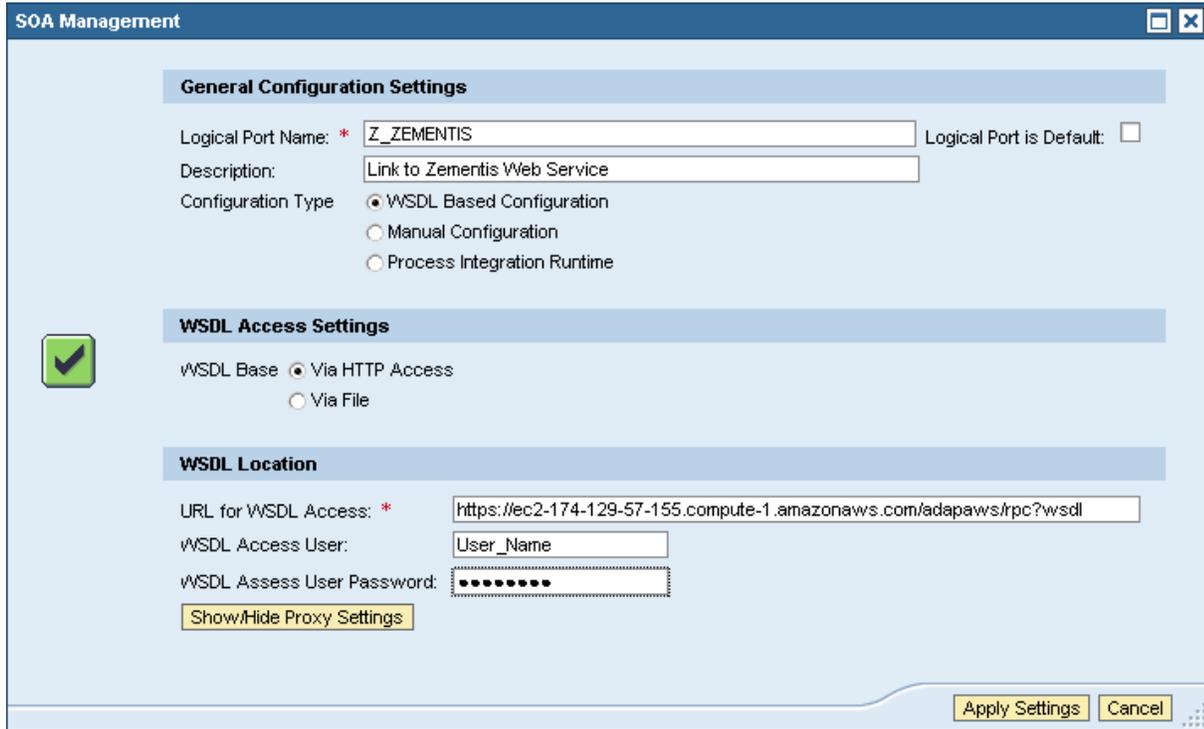
Figure 5: SAP Object Navigator

Creating a Logical Port

The web service destination, as well as the authorization definitions are maintained in the logical port definition.

SAP Transaction: SOAMANAGER -> Application and Scenario Communication -> Single Service Administration -> Change the “Search by” value to “Consumer Proxy” and enter client proxy name and change the “Field” value to “Internal Name”.

Click the “Apply Selection” button and click “Create Logical Port”. Define the following settings and click the “Apply Settings” button.



The screenshot displays the 'SOA Management' console window with the following configuration details:

- General Configuration Settings:**
 - Logical Port Name: * Z_ZEMENTIS
 - Logical Port is Default:
 - Description: Link to Zementis Web Service
 - Configuration Type:
 - WSDL Based Configuration
 - Manual Configuration
 - Process Integration Runtime
- WSDL Access Settings:**
 - WSDL Base:
 - Via HTTP Access
 - Via File
- WSDL Location:**
 - URL for WSDL Access: * https://ec2-174-129-57-155.compute-1.amazonaws.com/adapaws/rpc?wsdl
 - WSDL Access User: User_Name
 - WSDL Access User Password: [Masked]
 - Show/Hide Proxy Settings

Buttons at the bottom right: Apply Settings, Cancel

Figure 6: SAP SOA Management Console

Creating ABAP Code to Call the Client Proxy in the SAP Application Exit

Now, you only need to add some ABAP code into the SAP application exit and your SAP application is ready to consume results from your predictive models in real-time.

DATA:

```
* Reference variables for proxy and exception class
lo_clientProxy    TYPE REF TO Z2CO_RPCSERVICE,
lo_sys_exception  TYPE REF TO cx_ai_system_fault,
```

```
* Structures to set and get message content
ls_request        TYPE Z2TREE_MODEL_2009_input,
ls_response       TYPE Z2TREE_MODEL_2009_output.
```

TRY.

```
CREATE OBJECT lo_clientProxy
EXPORTING
  logical_port_name = 'Z_ZEMENTIS'.
```

```
* do synchronous client proxy call
```

```
CALL METHOD lo_clientProxy->TREE_MODEL
  EXPORTING input = ls_request
  IMPORTING output = ls_response.
```

```
CATCH cx_ai_system_fault INTO lo_sys_exception.
```

```
* Error handling
```

```
ENDTRY.
```

Copyright

© Copyright 2009 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects S.A. in the United States and in other countries. Business Objects is an SAP company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.