SPL™ Smart Pick Lists™

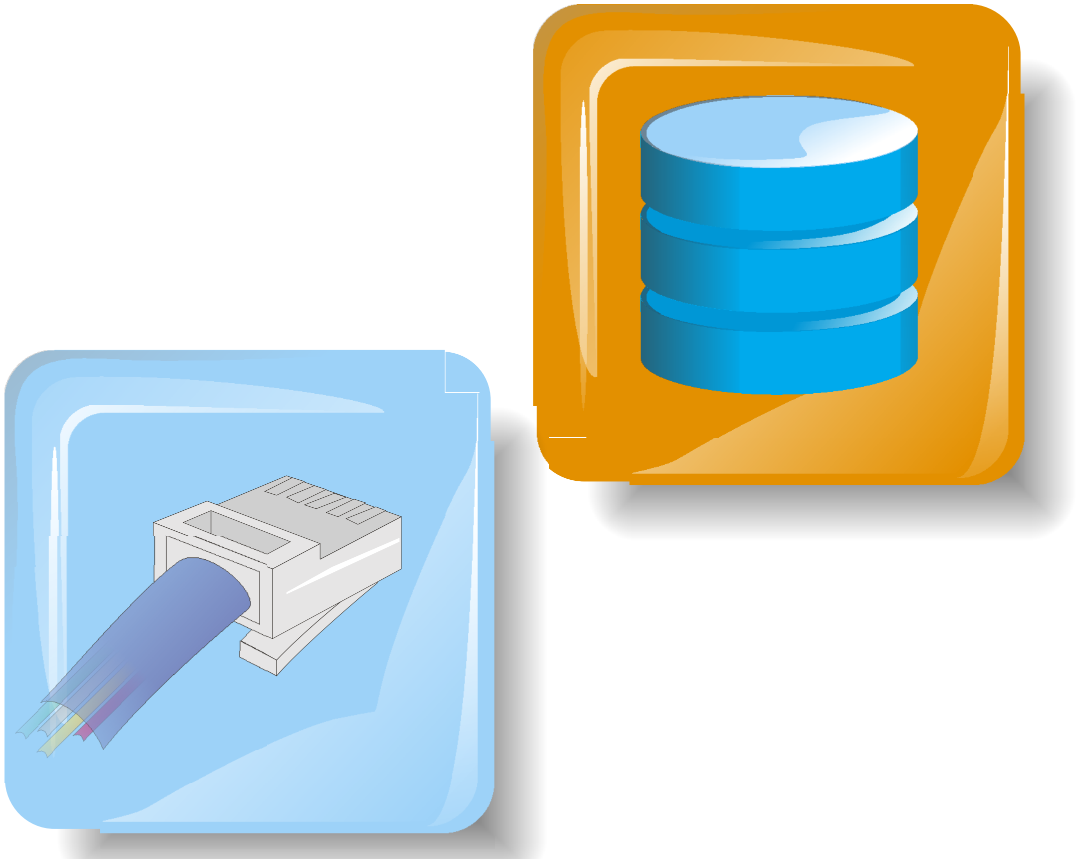
UTILITY

# SPL™ (Smart Pick Lists™)

The dynamic parameter pick lists solution for Crystal Enterprise™

Supports
Crystal Enterprise™ 9.0

# Architecture Overview and User Guide



VERSION 2

software Forces™

Software Forces, LLC 108-36 Queens Boulevard · Suite 201 Forest Hills · NY 11375
Phone: 718.544.7400 · Fax:718.544.4921 · www.softwareforces.com

Copyright © 2002 - 2004 by Software Forces, LLC · All rights reserved. · SPL, Smart Pick Lists, the SPL “pick list” logo, and the Software Forces logo are trademarks of Software Forces, LLC. All other brands and product names are Trademarks are (™) or registered trademarks (®) and the property of their respective companies.

## Copyright and Trademark Notice

**Copyright © 2003-2004 by Software Forces, LLC**
**All rights reserved.**

**SPL™ (Smart Pick Lists™) version 2**

Architecture Overview and User Guide

The contents of this manual and the associated SPL™ (Smart Pick Lists™) software are the property of Software Forces, LLC and are copyrighted.

This documentation may not be reproduced in any form, for any purpose, unless it complies with the terms presented on the SOFTWARE LICENSE AGREEMENT and/or END USER LICENSE AGREEMENT supplied.

Software Forces, LLC and the program authors have no liability to the purchaser or any other entity, with respect to any liability, loss, or damage caused, directly or indirectly by this software, including but not limited to, any interruptions of service, loss of business, anticipatory profits, or consequential damages resulting from the use of or operation of this software.

Updates may be made to this documentation and incorporated into later editions without prior notice.

**Smart Pick Lists, SPL, the SPL "pick list" logo, and the Software Forces logo are trademarks of Software Forces, LLC.**

**All other brands and product names are trademarks (™) or registered trademarks (®) and are the property of their respective companies.**

Fourth Printing May 2004
Printed in the United States of America

## Table of Contents

# Chapter 1

## 1.0 End User License Agreement

SOFTWARE FORCES, LLC
END USER LICENSE AGREEMENT
AND WARRANTY DISCLAIMER

NOTICE: This is a legally binding contract between you, the end user, and SOFTWARE FORCES LLC.

SOFTWARE FORCES LLC ("Software Forces" or "LICENSOR") LICENSES THE ENCLOSED SOFTWARE TO YOU ("USER" or "LICENSEE") ONLY UPON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS LICENSE AGREEMENT. PLEASE READ THE TERMS CAREFULLY BEFORE INSTALLING THE SOFTWARE, AS INSTALLING THE SOFTWARE WILL INDICATE YOUR ASSENT TO THEM. IF YOU DO NOT AGREE TO THESE TERMS, THEN SOFTWARE FORCES IS UNWILLING TO LICENSE THE SOFTWARE TO YOU, IN WHICH EVENT YOU SHOULD RETURN THE FULL PRODUCT WITH PROOF OF PURCHASE TO LICENSOR, AND YOUR MONEY WILL BE REFUNDED.

LICENSE AND WARRANTY:
The software that accompanies this license is the property of Software Forces or its licensors, and is protected by copyright and other intellectual property law. Although Software Forces at all times owns the Software, you will have certain rights to use the Software after your acceptance of this license. Except as may be modified by an addendum which may accompany or be added to this license, Your rights and obligations with respect to the use of this Software are as follows:

You may:
(i) only allow the specified number of Users to use the Software and documentation, as indicated as "Quantity" or "Number of Licenses" on the invoice, Quotation or electronic confirmation issued by Software Forces. "User(s)" means Licensee, if Licensee is an individual purchasing the Software for use at office or home (in which case Licensee's immediate family members residing in the same household shall not be considered additional Users), or, if Licensee is a business or commercial entity or government agency, its current employees. For qualifying educational institutions, "Users" also means faculty and staff teaching for or employed by Licensee and registered students enrolled at a single campus operated by Licensee. Subject to the limitations of this License, each authorized User may only use the Software on any central processing unit ("CPU"), workstation or portable, which is owned or controlled by Licensee; (ii) use the Software on a network, provided that Licensee has a licensed copy of the Software for each computer that can access the Software over that network; (iii) after written notice to Software Forces, transfer the Software on a permanent basis to another person or entity, provided that you retain no copies of the Software and the transferee agrees to the terms of this agreement

You may not:
(i) rent, lease, distribute, license, or otherwise transfer the Software or its documentation to any other party. Licensee may make a reasonable number of back-up copies for archival purposes only. The Software contains copyrighted material, trade secrets and other proprietary material. If Licensee has the right to duplicate the Software for multiple Users, then Licensee must reproduce on all such copies of the Software the copyright notices and any other proprietary legends that were on the original copy of the Software; or (ii) install or otherwise operate the software in a multi-user and/or CPU sharing mode and/or operate the software in a virtual machine environment unless Licensee has requested and has been granted an explicit right to do so in writing; or (iii) decompile, reverse engineer, disassemble, make any attempt to discover the source code of the Software or otherwise reduce the Software to a human perceivable form, or to modify, network, or create derivative works based upon the Software or the documentation in whole or in part, nor permit any other party to do so.

Notwithstanding anything herein, if the Software is lawfully acquired outside of the United States within a jurisdiction which is a member of the European Union subject to the EEC Council Directive 91/250/EEC of May 14, 1991, Licensee agrees that within that jurisdiction it shall not, and shall not allow any party on Licensee's behalf to, attempt to reverse engineer or decompile the Software into another computer language, except as expressly and specifically provided in the EEC Council Directive 91/250/EEC of May 14, 1991. Any and all information obtained during such lawful reverse engineering and/or decompiling activities, including but not limited to, the organization, logic, algorithms and processes of the Software, shall be deemed to be the confidential and proprietary information of Software Forces or its Licensors. Licensee shall not make copies of the copyrighted Software documentation without the prior written permission of Software Forces provided that for electronic transactions, Licensee may make one (1) hard copy of such documentation for each User.

In the case that this software is installed on a trial basis, Licensee shall have 30 days to decide whether to purchase the software. If the software is not purchased within 30 days and a “Purchase Key” obtained from Software Forces, then the software will render itself useless. Software Forces shall not be responsible for any loss of data, delay, or any other result caused by this Software rendering itself unusable. Licensee shall be responsible to remove the Software from any computer or network under where the trial copy may have been installed.

Technical Support:
Licensee must register online in order to be eligible for technical support via electronic mail.

Export Law Assurances:
Licensee agrees and certifies that neither the Software nor any other technical data received from Software Forces, nor the direct product thereof, will be exported outside the United States except as authorized and as permitted by the laws and regulations of the United States. If the Software has been rightfully obtained by Licensee outside of the United States, Licensee agrees that Licensee will not re-export the Software nor any other technical data received from Software Forces, nor the direct product thereof, except as permitted by the laws and regulations of the United States and the laws and regulations of the jurisdiction in which Licensee obtained the Software. The Software may not be exported to any of the following countries: Cuba, Iran, Iraq, Libya, North Korea, Sudan or Syria.

Termination:
This License is effective until terminated. Licensee may terminate this License at any time by destroying all copies of the Software and its documentation. This License will terminate immediately without notice from Software Forces if Licensee fails to comply with any provision of this License. Upon termination, Licensee must destroy all copies of the Software and its documentation and cease and desist from any further use of the Software.

Limited Warranty:
Software Forces warrants that the Software shall perform substantially as described in its documentation for a period of thirty (30) days from purchase. Licensee's sole remedy in the event of a breach of this warranty will be that Software Forces will, at its option refund the money you paid for the Software. Software Forces does not warrant that the Software will meet your requirements or that operation of the Software will be uninterrupted or that the Software will be error-free.

THE ABOVE WARRANTY IS EXCLUSIVE AND IN LIEU OF ALL OTHER WARRANTIES, WHETHER EXPRESS OR IMPLIED, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHER RIGHTS, WHICH VARY FROM STATE TO STATE.

Disclaimer of Damages:
REGARDLESS OF WHETHER ANY REMEDY SET FORTH HEREIN FAILS OF ITS ESSENTIAL PURPOSE, IN NO EVENT WILL SOFTWARE FORCES BE LIABLE TO YOU FOR ANY SPECIAL, CONSEQUENTIAL, INDIRECT OR SIMILAR DAMAGES, INCLUDING ANY LOST PROFITS OR LOST DATA ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE EVEN IF SOFTWARE FORCES HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU. IN NO CASE SHALL SOFTWARE FORCES' LIABILITY EXCEED THE PURCHASE PRICE FOR THE SOFTWARE.

The disclaimers and limitations set forth above will apply regardless of whether you (Licensee) accept the Software.

U.S. Government Restricted Rights:
DISTRIBUTION TO THE U.S. GOVERNMENT. This Software is commercial software developed exclusively at private expense. Use, duplication, or disclosure by civilian agencies of the U.S. Government shall be in accordance with subparagraphs (c)(1) and (2) of the Commercial Computer Software-Restricted Rights clause at 48 CFR 52.227-19. Use, duplication, or disclosure by Department of Defense agencies is subject solely to the terms of this software licensing agreement pursuant to DFARS 227.7202. Contractor/manufacturer of the Software is Software Forces LLC, 108-36 Queens Blvd. Suite 201, Forest Hills, NY 11375.

General:
This Agreement will be governed by the laws of the State of New York. This Agreement may only be modified by a license addendum which may accompany or be added to this license. Should you have any questions concerning this Agreement, or if you desire to contact Software Forces for any reason, please write to: Software Forces, LLC, 108-36 Queens Blvd. Suite 201, Forest Hills, NY 11375.

# Chapter 2

## 2.0 Introduction

SPL™ 2 (Smart Pick Lists™) for Crystal Enterprise™ is a third generation dynamic parameter product for Crystal Enterprise.

The original SPL was designed for Crystal Enterprise 8.0. During our consulting engagements, we found numerous customers requesting the same set of features, specifically dynamic values for their parameters.

While the first generations of SPL were mostly custom designs integrated with Crystal Enterprise and typically custom frontends, today SPL has evolved from customer driven requests for an even more dynamic product that can support both ePortfolio out-of-the-box as well as custom frontends, including the use of custom definable controls.

This third generation of SPL is a complete architecture redesign that has been engineered to improve performance (by reducing the network traffic consumed, and intelligently handle many requests server side instead of client side), provide a rich API for building custom frontends and custom controls.

This architecture has also been designed to reduce, where possible, additional coding required for many areas (without sacrificing customization) either server-side or client side – by handling as much as possible internally server-side by SPL.

# Chapter 3

## 3.0 Architecture Overview

### 3.1 Design goals

The Software Forces SPL™ 2 (Smart Pick Lists™) product was designed with several goals in mind:

- ease of use
  1. not require modification to existing published reports
  2. support updates to Crystal Enterprise (via patches or version upgrades) with out forcing you to make extensive updates to ePortfolio or your custom frontend(s)
  3. provide a seamless interface to ePortfolio that supports both View-On-Demand and Schedule

- performance
  1. handle as much processing as possible server-side
  2. reduce the bandwidth consumption

- customization
  1. minimize direct changes to ePortfolio source files
  2. support custom frontend(s) development and integration
  3. support customization to the look and feel delivered

and with three audiences in mind:

- **End-user**:
  To provide an easy to use extensible and modern web based front-end that integrates seamlessly in to ePortfolio and maintains the same look and feel that users are already used to in ePortfolio.

- **Administrator**:
  To provide an environment that allows minimum changes to the Crystal Enterprise source files and at the same time ensure an easy migration to future versions of Crystal Enterprise.

- **Developer**:
  To provide a set of API's that allow modifications to the out-of-the-box ePortfolio integration or support design of a completely custom frontend.

## 3.2 Components

SPL 2 was designed and architected as a unique extension to the Crystal Enterprise architecture. Designed specifically to not require modification to existing published reports, and designed to require minimal effort to integrate and use with ePortfolio.

SPL 2 consists of 4 main components:

- SPL backend
- SPL MMC
- ePortfolio frontend
- SPL repository

### 3.2.1 SPL backend

The SPL backend is a COM object with an extensive and Smart API™, it is the "brain" of the SPL architecture.

Through this Smart API, both the SPL MMC and ePortfolio frontend communicate with the SPL backend.

SPL MMC communicates via this API to store the following values in to the SPL repository:

- Drivers description
- Connections settings
- Reports you've selected from Crystal Enterprise for SPL to interact with
- Reports' Parameter(s) you've defined to be dynamic and/or cascading
- Options you define

The ePortfolio frontend communicates via this API to tell the SPL backend which report to work with. The SPL backend then automatically determines which parameters (based on definitions made in the SPL MMC) should receive dynamic values and which parameters have a cascade.

Additionally the SPL backend determines the type of parameter (discrete value, multiple value, discrete range, etc.) based on the type defined in the report, and then dynamically builds the HTML code which is what is displayed in ePortfolio.

The HTML code that is assembled by the SPL backend is generated using our patent pending XML based Control Factory. Using our XML Control API we have defined a set of controls (input box, list box, drop-down box, etc.), their behavior, and validation that is processed by our XML Control Factory.

Should you require a different look and feel to the controls, you can define your own control(s) using your favorite XML editor and our XML Control API (described later in this guide).

In creating the above XML Control Factory and XML Control API, we intentionally separated the controls that we define and ship with the product, from the custom controls that you define. This separation allows us to continue evolving and extending the standard controls without overwriting your custom controls in future versions.

### 3.2.2 SPL MMC

The SPL MMC is a Microsoft Management Console (MMC) version 1.1 (or higher) compliant snap-in that communicates with the SPL backend via the Smart API.

Administration and management of SPL is done via this MMC interface.

We chose to create our administration and management frontend using MMC due to several factors:

- available with every version of Windows, dating back to Windows 95
- user interface (UI) is one that most administrators are already familiar with
- supports remote management (coming in a future version of SPL MMC)
- supports multiple server management (coming in a future version of SPL MMC)

The SPL MMC is therefore used by a report designer or appropriate administrator to:

- create and manage connections to databases
- select reports directly from Crystal Enterprise to be managed by SPL
- configure parameters of selected reports to define which connection is used to retrieve dynamic values
- configure parameters of selected reports to define cascading hierarchy

- configure the look and feel presented in ePortfolio
- configure custom controls

### 3.2.3 ePortfolio frontend

ePortfolio is the frontend that ships with Crystal Enterprise. This frontend out-of-the-box does not support dynamic and/or cascading parameters during View-On-Demand or Schedule of reports.

Although this frontend was originally intended to show off the capabilities of Crystal Enterprise and it was expected that custom frontends would be build by companies using the Crystal Enterprise COM objects, the ePortfolio interface has been adapted by many companies as their standard interface in to Crystal Enterprise.

Because of this acceptance, and request of our customers, we have written extensive integration of SPL in to ePortfolio using our SPL backend COM object Smart APIs.

However, in designing our integration, we were careful to:

- not tie our ePortfolio integration to any specific version of Crystal Enterprise or ePortfolio
- support patches or version updates to Crystal Enterprise
- provide extensibility via support of our custom controls (using our XML Control API)
- support customers wishing to use custom frontends outside of ePortfolio
- minimize the development effort required to add SPL to a custom frontend

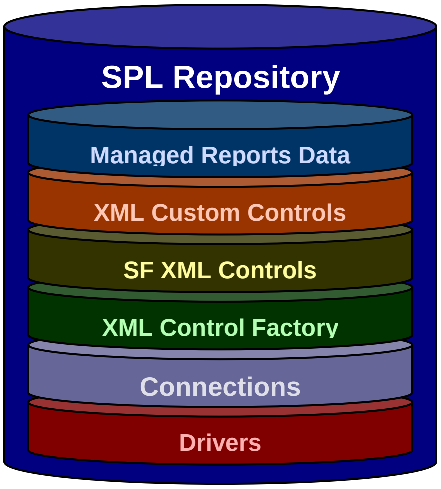
To accomplish these tasks SPL's backend was developed to handle the bulk of the work and return to the web frontend optimized HTML. This approach also increases performance and reduces the bandwidth consumed.

With the above approach to integration with e-Portofolio we have ensured that as long as the APIs that ePortfolio uses remain the same in future versions of Crystal Enterprise, upgrading will require specifying the same redirects to SPL in future versions of Crystal Enterprise.

### 3.2.4 SPL repository

The SPL repository is set of tables residing in a database that the SPL backend in conjunction with the SPL MMC use to store connections, reports and parameter definitions, options and custom controls.

Supported databases to host this repository are MSDE, Oracle, and SQL Server. DB2, Informix, and Sybase support is expected in a future release.



## 3.3 Out-of-the-box integration with ePortfolio and support of custom frontends

We wanted to provide out-of-the-box integration with the frontend that users are most familiar with (ePortfolio) and at the same time make SPL flexible enough to be used outside of ePortfolio using a custom frontend if desired.

Critical to our design philosophy was to design an architecture that would allow for easy service pack installs and/or upgrades to future versions of both Crystal Enterprise and SPL.

The result is an architecture that has a rich API and an easy to use frontend.

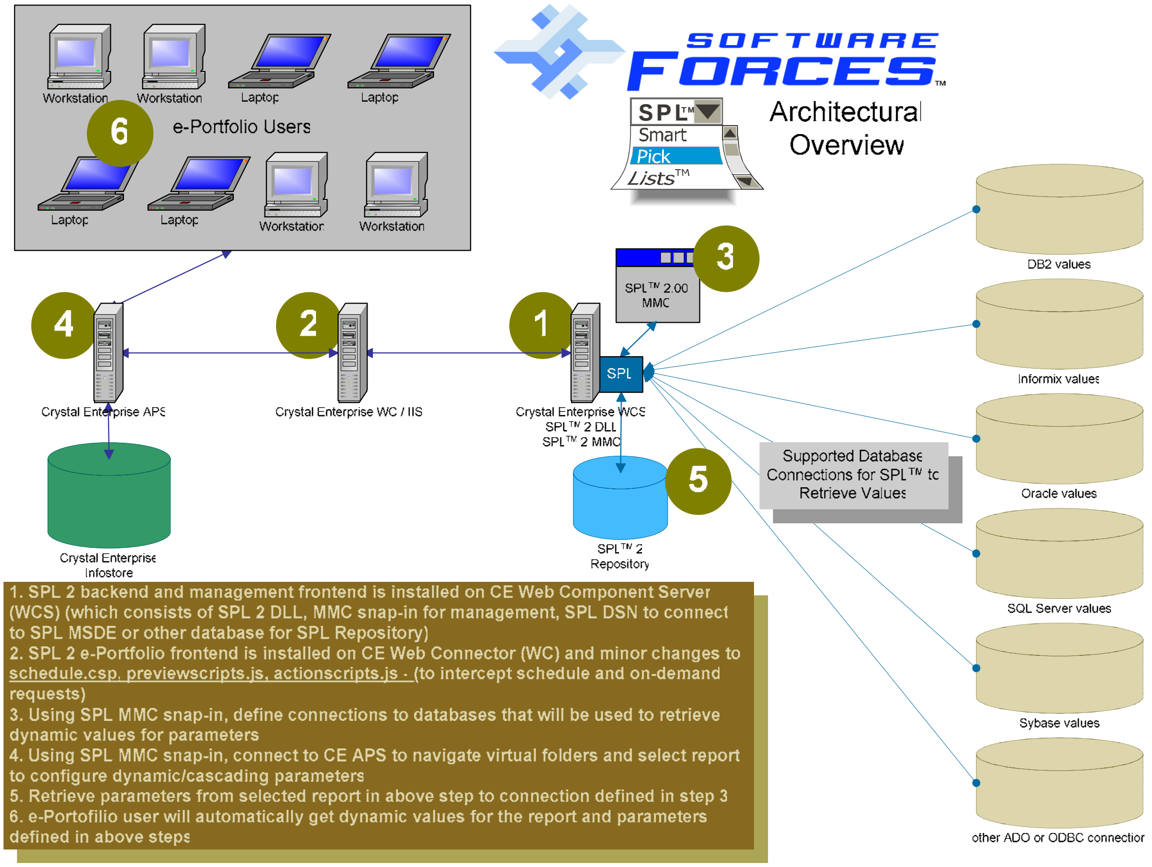
Introduced with service pack 2 for SPL 2, is a new feature called “Web Styles”. Web Styles (can be configured per report) allow you to change the look and feel of the web frontend displayed for the parameter selection screen in ePortfolio. With service pack 2 for SPL 2, there are now six (6) Web Styles to choose from. Chapter 5, section 5.1.4 on page 28 provides further details.

Rather than making extensive changes to the Crystal Enterprise .csp and .js code pages, our approach instead requires 1 redirect in “**schedule.csp**”, 1 redirect in “**previewscripts.js**”, and as of service pack 2 for SPL now 1 redirect in "**actionscripts.js**" to a set of SPL .csp and .js intercept code pages. Chapter 4, section 4.3 (on page 21) provides instructions and details.

In the SPL .csp and .js code pages, we make all necessary calls to the SPL backend and deliver a frontend that looks seamless to ePortfolio. These API calls are responsible for connecting to the SPL backend and retrieving the dynamic values.

Using our SPL MMC snap-in, a report designer (or appropriate administrator) highlights a report from Crystal Enterprises, selects parameters and associates connections to the database to retrieve dynamic values for these parameters.

When an end-user logs in to ePortfolio and requests to view or schedule a report that has been assigned to the SPL backend (via the SPL MMC), all values for parameters are automatically retrieved from the database configured for the parameter in the same user interface that the user is already used to in ePortfolio.



1. SPL 2 backend and management frontend is installed on CE Web Component Server (WCS) (which consists of SPL 2 DLL, MMC snap-in for management, SPL DSN to connect to SPL MSDE or other database for SPL Repository)
2. SPL 2 e-Portfolio frontend is installed on CE Web Connector (WC) and minor changes to schedule.csp, previewscripts.js, actionscripts.js - (to intercept schedule and on-demand requests)
3. Using SPL MMC snap-in, define connections to databases that will be used to retrieve dynamic values for parameters
4. Using SPL MMC snap-in, connect to CE APS to navigate virtual folders and select report to configure dynamic/cascading parameters
5. Retrieve parameters from selected report in above step to connection defined in step 3
6. e-Portofilio user will automatically get dynamic values for the report and parameters defined in above steps

**Figure 1 – SPL™ 2 architectural overview**

## 3.4 XML Control Factory

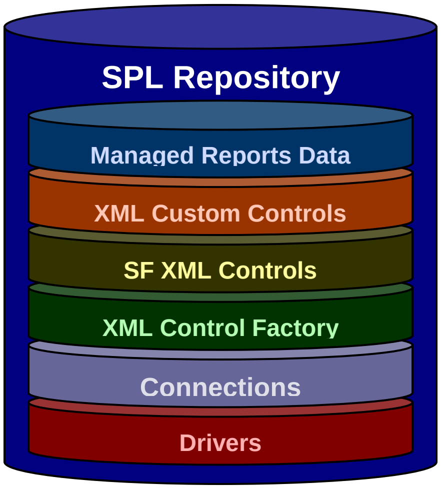
The HTML code that is assembled by the SPL backend is generated using our patent pending XML based Control Factory. Using our XML Control API we have defined a set of controls (input box, list box, drop-down box, etc.), their behavior, and validation that is processed by our XML Control Factory.

## 3.5 XML SF Controls

When the "**SPL Repository**" is installed, it is populated with a pre-defined set of controls that we have already built for you to use with either ePortfolio or your custom frontend. These XML SF Controls can not be modified; instead you can create your own XML Custom Controls using the XML Control API and even override the pre-defined XML SF Control(s) with your own.

In creating the XML Control Factory and XML Control API, we intentionally separated the controls that we define and ship with the product, from the custom controls that you define. This separation allows us to continue evolving and extending the standard controls without overwriting your custom controls in future versions.

## 3.6 XML Custom Controls



Should you require a different look and feel to the pre-defined controls, you can define your own using your favorite XML editor and our XML Control API (described in chapter 8, section 8.6.3 on page 110 and chapter 10 on page 113).

In creating the XML Control Factory and XML Control API, we intentionally separated the controls that we define and ship with the product, from the custom controls that you define. This separation allows us to continue evolving and extending the standard controls without overwriting your custom controls in future versions.

# Chapter 4

## 4.0 Installation

### 4.1 SPL 2 Installation Program(s)

The SPL installer is a self contained installer for three (3) major components of SPL.

If your Crystal Enterprise environment (all services / servers) are on the same computer, then you will want to run the installation program multiple times to select the major component to install.

If however your Crystal Enterprise environment is spread out on multiple computers (i.e. APS on one computer, WCS on another computer, and WC on yet another computer) then you will want to run this same installation program on the different computers, each time selecting the appropriate main component to be installed on that server.

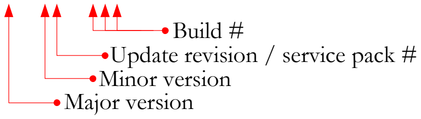
A description of the three (3) main components and an explanation on which Crystal Enterprise server(s) they need to be installed is described later in this section.

The installation program name for SPL is made up of the four (4) informational items part of the program name:

- Major version
- Minor version
- Update revision / service pack #
- Build #

For example, if the major version of SPL is 2; the minor version is 0; the update revision / service pack number is 0 and the build is 485 then the installation program name would be:

SPLSetup_2_00_485.exe



While the version you have may already be newer than the above example, the above syntax will be similar.

To start the SPL 2 installation, please run **SPLSetup_2_00_485.exe** (you may need to look for a file that will have a different major version, minor version, update revision / service pack #, and build #).

Once the installation program loads, you will be presented with an informational screen that will display the program you are installing.

After clicking on the "Next" button, you will be presented with a screen that contains the End User License Agreement (EULA). By selecting "I accept the agreement" you make a legal commitment that you have read, understand, and agree to the terms set forth in the EULA.

After clicking on the "Next" button, you will be presented with a screen that contains information from our "**readme.txt"** file. This information contains various information that you should be aware of before proceeding with the installation (such as last minute known issues, how to contact technical support, and minimum requirements). Please take a few moments to review information here before proceeding.

After clicking on the "Next" button, you will be presented with a screen to provide your User Name and Organization name (which are usually read from your Windows environment).

Once you click on the "Next" button, you will be presented with a drop-down box to select the main component you would like to install.

The SPL 2 installation program consists of three main components:

- Backend Install (on WCS) – this will launch a sub-installer that allows you to select backend related components to install on Web Component Server (WCS). These components are "**SPL 2 backend**", "**SPL 2 MMC**", "**SPL 2 Repository**" (see section 4.1.1 for more details)
- SPL ePortfolio Integration (on WC) – this will launch a sub-installer that allows you to select the frontend related components to install on the Web Connector (WC). These components are "**SPL ePortfolio Integration (on WC)**", "**Overwrite original "previewscripts.js" & "schedule.csp" for SPL redirection (on WC)**" (see section 4.1.2 for more details)
- Documentation

### 4.1.1 Backend Install (on WCS)

Selecting the "**Backend Install (on WCS)**" component from the main installer will launch a sub-installer with these components:

- Backend Full Install (on WCS) – this will install the "**SPL 2 backend DLL**", "**SPL 2 MMC"**, launch the "**SPL 2 Repository**" sub-installer, and install the "**Documentation"**
- SPL 2 Repository (on WCS) – this will launch the "**SPL 2 Repository**" sub-installer (see section 4.1.1.1 for more details)
- Custom Install – this will allow you to select individual components from those that are available in the "**Backend**

**Full Install (on WCS)**" component (selecting a sub-installer does not allow you to select its individual components until the sub-installer is launched)

- Documentation – this will install this user guide, readme.txt, whatsnew.txt, wcs_steps.txt, and wc_steps.txt

#### 4.1.1.1 SPL 2 Repository (on WCS)

Selecting the "**Backend Full Install (on WCS)**" or "**SPL 2 Repository**" component from the "**Backend Install (WCS)**" sub-installer will launch the "**SPL 2 Repository**" sub-installer with these components:

- SPL Repository Full (on WCS) – this will install the "**SPL 2 Repository Populate / Update Utility**", "**SPL 2 Repository Populate / Update Data**", "**SPL 2 Repository Database Scripts**"
- Custom Install – this will allow you to select individual items to install from those that are available in the "**SPL Repository Full (on WCS)**" component

### 4.1.2 SPL ePortfolio Integration (on WC)

Selecting the "**SPL ePortfolio Integration (on WC)**" component from the main installer will launch a sub-installer with these components:

- SPL ePortfolio Integration (on WC) – this will install all the necessary files that the View-On-Demand and Schedule requests will redirect to. This will not overwrite or update the two ePortfolio files that do the actual redirection.
- Overwrite original "previewscripts.js" & "schedule.csp" & "actionscripts.js" for SPL redirection (on WC) – this will overwrite the three ePortfolio files that redirect to the SPL files for View-On-Demand and Schedule,.

  **CAUTION:** The files used to overwrite are based on original CE9 and your original files are not backed up.

  It is not recommended to use this option. Instead it is recommended that you make the necessary changes to these two files manually, based on instructions provided in section 4.3 of this guide.

- Custom Install – this will allow you to select both of the above choices

### 4.1.3 Documentation

This will install this user guide, spl2.chm, readme.txt, whatsnew.txt, wcs_steps.txt, and wc_steps.txt.

## 4.2 Recommended Installation Order and Procedures

### 4.2.1 SPL 2 backend DLL, SPL 2 MMC snap-in, SPL 2 Repository

1. Install "**Backend Full Install (on WCS)**". In addition to the SPL backend DLL and SPL MMC snap-in, this will also install the "**SPL 2 Repository Full (on WCS)**" which by default will install three necessary parts on your WCS for the SPL repository:

   in the "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository**" a program called "**SPLdbUpdate.exe**";

   in "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository\Data**" the necessary data files to populate a new SPL repository table structure. Any none structure related changes or fixes to the contents of the repository will be provided as updates to these files in this directory;

   in "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository\SQL Scripts**" the necessary SQL Scripts for MSDE, Oracle, and SQL Server to both create the necessary table structure(s) in your database as well as to drop those table structure(s) from your database.

2. Run the provided database SQL script (by default installed in "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository\SQL Scripts**") on your database to create the necessary SPL 2 repository table structure(s). If the database server supports naming the database, it is recommended that you name it "SmartPickLists" (but this is not required).

3. Create an ODBC DSN called "**SPL**" to the database where the newly created SPL 2 repository table structures exist. This ODBC DSN should be created on the WCS where the SPL backend and SPL MMC is/will be installed.

   If the driver used to create the ODBC DSN supports "Test Connection", use this option to ensure that a connection to your database can be established.

4. Run the "**SPL 2 Repository Update**" utility (a shortcut should be in the "**Software Forces|Smart Pick Lists|2_0**" programs group or you can run the executable called "**SPLdbUpdate.exe**"

in "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository**").

Specify the location to the "**repository data files**" (by default this should be "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository\Data**");
specify the ODBC DSN (default is "**SPL**");
specify the Login ID needed to connect to the database;
specify the Password needed to connect to the database.

Click on the "**Update**" button to begin populating / updating the SPL repository table structure(s).

5. If the repository was installed as part of a custom install, and the SPL backend and SPL MMC were not installed yet, then install "**Backend Full Install (on WCS)**" now (this will install the SPL backend and SPL MMC), otherwise skip to the next step.

6. On the WCS where the SPL MMC was installed, run the SPL MMC and in "**Tools|Options**" specify the ODBC DSN, Login ID, and Password that was configured in step #3.

   For these settings to take affect, you must restart the SPL MMC snap-in.

### 4.2.2 SPL 2 ePortfolio integration

1. Install "**SPL ePortfolio Integration (on WC)**".

   This will install all the necessary files needed when ePorfolio will redirect to SPL.

   By default these files will be installed in "**Program Files\Crystal Decisions\Web Content\Enterprise9\ePortfolio\en\SPL**"

2. Manually update three original ePortfolio files as described in section 4.3.

## 4.3 ePortfolio frontend integration

The ePortfolio integration code must be installed on the Crystal Enterprise Web Connector (WC) server. This is often the same server where your web server (i.e. IIS) resides.

Using the "WC Base Install" option in the setup program will create a SPL directory under the ePortfolio directory structure you specify and place in to this directory the necessary SPL .csp and .js files.

Once setup has completed you will need to make redirection modifications to three ePortfolio files to point to the new SPL based .csp and .js files.

Please note that in the code snippets below, you will notice that the SPL updates are enclosed in a commented out SPL start tag ("**//<SPL>**") and end tag ("**//</SPL>**"). It is recommended that you keep this part of the update you make, as in the future it will be easier for your to search for these tags to find where the SPL update code resides -- should you wish to update it or remove it.

1. The first file you will need to modify is called "**schedule.csp**" and resides in "**Program Files\Crystal Decisions\Web Content\Enterprise9\ePortfolio\en**"

   Look for a line that says (line #330 in most cases):

   ```
       if(Request.QueryString.Item("id").Count != 0 &&
   Request.QueryString.Item("id") != "")
   ```

   this line should be a start of an "if" condition that should look like this:

   ```
       if(Request.QueryString.Item("id").Count != 0 &&
   Request.QueryString.Item("id") != "")
       {
               id = Request.QueryString.Item("id");
       }
       else if(Request.Form.Item("id").Count != 0 && Request.Form.Item("id")
   != "")
       {
               id = Request.Form.Item("id");
       }
       else if(Request.QueryString.Item("oid").Count != 0 &&
   Request.QueryString.Item("oid") != "")
       {
               id = Request.QueryString.Item("oid");
       }
   ```

after this "if" condition add (after line #342 in most cases):

```
//<SPL>
if(Request.QueryString.Item("click") == "yes"){
        SetSession("ScheduleNew", "true");
}
//</SPL>
```

Then search for a line that says (line # 1223 in most cases):

**//PARAMETERS DIALOGS GO HERE  *****

Look for a line that says (line #1224 in most cases):

```
if(tab == "parameters")  {
```

Immediately following the above line, add:

```
//<SPL>
    var vFirstTime = "";
    if(GetSession("ScheduleNew")=="true"){
        SetSession("ScheduleNew", "false");
        vFirstTime = "&refreshReport=true";
    }

    var SPL = Request.QueryString.Item("SPL");
    if (SPL != "true"){
        Response.Redirect("SPL/router.csp?reportid="+id+vFirstTime);
    }
//</SPL>
```

2. The second file you need to modify is called “**previewscripts.js**” and resides in **“Program Files\Crystal Decisions\Web Content\Enterprise9\ePortfolio\en”**

   Search for a line that says (line # 21 in most cases):

   **//FUNCTION WHICH LAUNCHES REPORT VIEWER WHEN USER CLICKS ON REPORT PREVIEW**

   Look for a line that says (line #24 in most cases):

```
if (IsAShortcut == 0){
```

   this line should be a start of an "if" condition that should look like this:

```
if (IsAShortcut == 0){
        ShowReport(PreviewTitle, PreviewLaunch, PreviewType);
}
else {
        ShowReport(PreviewTitle, IsAShortcut, PreviewType);
}
```

replace this "if" condition with:

```
    //<SPL>
    var LeftPosition = (screen.width) ? (screen.width-800)/2 : 0;
    var TopPosition = (screen.height) ? (screen.height-600)/2 : 0;
    var winProps = "width=800,height=600,location=no,scrollbars=yes,
    menubar=no,toolbar=no,resizable=yes,top="+TopPosition+",left="+LeftP
osition;
    var VOD = new String("vod");
    var redirUrl = "SPL/router.csp?reportid="+PreviewLaunch+"&
    purpose="+VOD;
    var reportWindow = window.open(redirUrl,"viewer",winProps);
    //</SPL>
```

*** Note that the contents of the "var winProps" and "var redirUrl" lines above should be all on one line in the actual script . In this documentation it is being wrapped to the next line due to font size. ***

the function called "**ShowReportPreview**" (that appears on line #22 in most cases) should now look like this:

```
    function ShowReportPreview() {
      if(PreviewLaunch != 0){
             //<SPL>
              var LeftPosition = (screen.width) ? (screen.width-800)/2 : 0;
              var TopPosition = (screen.height) ? (screen.height-600)/2 : 0;
              var winProps = "width=800,height=600,location=no,
scrollbars=yes,menubar=no,toolbar=no,resizable=yes,top="+TopPosition+",lef
t="+LeftPosition;
              var VOD = new String("vod");
              var redirUrl = "SPL/router.csp?reportid="+PreviewLaunch+"&
purpose="+VOD;
              var reportWindow = window.open(redirUrl,"viewer",winProps);
             //</SPL>
      }
    }
```

*** Note that the contents of the "var winProps" and "var redirUrl" lines above should be all on one line in the actual script . In this documentation it is being wrapped to the next line due to font size. ***

3. The third file you need to modify is called "**actionscripts.js**" and resides in
   **"Program Files\Crystal Decisions\Web Content\Enterprise9\ePortfolio\en"**

   Search for a line that says (line # 90 in most cases):

   **//FUNCTION WHICH OPENS THE SCHEDULE DIALOG WINDOW**

Look for a line that says (line #98 in most cases):

```
UtilWindow = window.open("schedule.csp?id="+PreviewLaunch, "u",
winProps);
```

*** Note that the contents of the "UtilWindow = " line above should be all on one line in the actual script . In this documentation it is being wrapped to the next line due to font size. ***

Replace this line with:

```
//<SPL>
UtilWindow =
window.open("schedule.csp?id="+PreviewLaunch+"&click=yes", "u",
winProps);
//</SPL>
```

*** Note that the contents of the "UtilWindow = " line above should be all on one line in the actual script . In this documentation it is being wrapped to the next line due to font size. ***

## 4.4 Custom frontend samples

An example of using the SPL architecture outside of ePortfolio is provided as several custom frontend samples. Each sample's code contains comments and a readme.txt file detailing specific information about the sample.

Samples are not installed by default, but can be installed on the Web Connector (WC) server, by selecting the "Samples" option on the installation program.

# Chapter 5

## 5.0 SPL user interface integrated with ePortfolio user interface

When we set out to design the ePortfolio integration, our goal was to design a user interface (UI) that would be a natural progression for existing ePortfolio users. We wanted to maintain the training investment made to learn ePortfolio, to carry over to using the SPL integration. Therefore, we have maintained naming conventions for labels and text, buttons, and color schemes, etc.

At the same time we wanted to improve on ePortfolio's design and performance in the area of parameter selection when working with SPL.

While, the SPL integration with ePortfolio is designed to be activated for reports that have been assigned to SPL to provide dynamic values for parameter(s), there is an option that can be turned on to use the SPL based frontend even on those reports that have not been assigned to SPL.

When an On-Demand-View or a Schedule request is received, the SPL integration performs a check to see if the requested report has been assigned to SPL. If it has, then the displayed parameter(s) page will come from SPL. If on the other hand, it was not, then the displayed parameter(s) page will come from the standard ePortfolio page (as if SPL was never integrated).

### 5.1 SPL integration improves performance and consumes less bandwidth

ePortfolio frontend with SPL integration changes the typical ePortfolio frontend behavior (as described above) by implementing one of 2 techniques (depending on browser support and your settings in the SPL MMC):

- iFrame
- Pop-up

iFrame (supported by most 4.x and higher browsers) is a technique that allows only a specific portion of a page to be refreshed instead of the entire page. This helps reduce the consumption of bandwidth and server load as it requires less data for the server to generate and therefore less data to be sent back to the client. This also improves the user experience as they do not see the entire page change, but rather just the portion of the page that needs to.

Pop-up (supported by most 4.x and higher browsers) is a technique that does not require the main page to be refreshed. And the dynamic values for a parameter a then displayed in a Pop-up window. Once the initial values are retrieved and displayed in the Pop-up window, all additional processing, including applying the selected values back to the main page is done client side. This helps reduce the consumption of bandwidth and server load as it requires less data for the server to generate and therefore less data to be sent back to the client. This also improves the user experience as they do not see the entire page change, but rather just the portion they selected in the Pop-up window.

The use of the above techniques is especially important when cascading (when the selected values you choose for one parameter act as a constraint for a dependant parameter's values) to reduce both the server side processing and bandwidth consumption.

The global setting of how to present the dynamic values (as iFrame or Pop-up) is configured via the SPL MMC.

However, in keeping with SPL's smart integration, we designed a fall-back mechanism in cases where iFrame is not supported by a browser.

Given a scenario where iFrame support was enabled globally, then clients with browsers capable of supporting iFrame will have the UI generated for iFrame. For each client that does not have a browser capable of supporting iFrame, we will automatically fall-back to display a Pop-up dialog.

If in the above scenario Pop-up was globally enabled, then no clients will ever receive a UI generated for iFrame. The assumption is that you always want Pop-up to be used.

A possible third technique based on the tradition ePortfolio page refresh may be added in a future version based on customer feedback.

### 5.1.1 Static values

SPL was designed to solve the problem of static values in parameters that are published to Crystal Enterprise. Maintaining the lists of values is a very tedious effort and usually requires reopening reports in Crystal Reports® and updating the values available for each parameter, then saving the report back to Crystal Enterprise.

This can quickly grow out of hand if you have lots of reports, especially if many of the reports have common values that need to be used for the parameter(s). This also still continues to maintain static values in the parameters because as new values are added to a database, the report does not automatically receive these values to use as choices for a parameter.

### 5.1.2 Dynamic values without modifying the report

SPL was designed to solve the problem of static values in parameters that are published to Crystal Enterprise. SPL was also designed to support dynamic parameters on existing published reports in Crystal Enterprise, all without ever modifying the report.

What this means to you is the ability to now take as many of the reports you have published in Crystal Enterprise and give them the ability to display to the user dynamic values for their parameters without ever opening them in Crystal Reports and without making a single change to the report.

Additionally this provides you with a means to make many changes to the report over time and not worry about having to configure SPL to support it. As long as 2 areas in the report do not change:

- the report id (opening a report in Crystal Reports from Crystal Enterprise and doing a Save As will change the report id but Save will not)
- the parameter name (SPL MMC is used to configure a parameter's dynamic value based on the parameter name in a report)

If either of these areas changes, then it's a simple process of doing a refresh (a feature added in service pack 1 – see chapter 8, section 8.4.3 on page 98) on the report already managed by SPL.

### 5.1.3 Dynamic values with Cascades

Cascades is the concept similar to a parent and child relationship. Where depending on the value selected in a parent parameter determines the values presented to be selected for a child parameter. So the values you select in the parent parameter work like a constraint for child or dependant parameter's available values to choose from.

For example, if you had a report with the following parameters (and in the order as below) all configured for dynamic values and cascades:

- Country
- Region
- City
- Employee

You would start with the Country parameter that will list all available countries as the values to be selected. Assuming you select only United States as your choice, you then display in the Region parameter a list of all regions for United States (because Country now acts as a parent to Region and the constraint for Region is the value(s) selected in Country).

Next in the Region parameter (which now lists only regions for United States) you select New York. Since the City parameter receives its constraint from the value(s) selected in Region, the list of available cities will be those of New York. Assuming you select Brooklyn, Manhattan, and Queens.

Now the final parameter which is Employee will display a list of available employees that are based on the constraint of Brooklyn, Manhattan, and Queens. So only employees from those cities will be available as values to be selected.

While SPL allows you to configure and display cascades for any report and any set of parameters, your report does need to be built properly to support this feature. Otherwise, you may need to modify your report(s) for this feature to function properly.

### 5.1.4 Web Styles (new as of service pack 2 for SPL 2)

With the release of service pack 2 for SPL 2, we have introduced a "Web Styles" feature and now ship six (6) web styles for you to choose from.

Web styles allow you to dynamically change the look and feel of how the web frontend of the parameters page is presented to the user.

A default web style to use when adding new reports can be set (see chapter 8, section 8.6.1 on page 105), and can also be overridden per report (see chapter 8, section 8.4.4 on page 98). This allows you to set a different web style to use for each report if so desired.

The six (6) web styles that are now available are:

- Style 1 - Default SPL style. Identical to CE 9 ePortfolio parameter selection web frontend. All parameters are displayed on one page. Available and current values are loaded on initial page load. Each parameter's available values are displayed first, and then under them the current values are displayed
- Style 2 – Based on default SPL style 1, but makes better use of screen real estate. All parameters are displayed on one page. Available and current values are loaded on initial page load. Each parameter's available values are displayed on the left and the current values are displayed on the right
- Style 3 – A wizard style that is unique to SPL. This web frontend aims to improve on both usability and performance. Usability is improved by not having to force the user to scroll through a long list of values by limiting one parameter per page in a Windows like wizard. The user is shown at the top the number of parameters that exist, and these are hyperlinks so that users can quickly go back to a previous parameter. The performance is improved by not needing to retrieve all

parameter's values in one shot, as they are only retrieved when on a specific parameter page

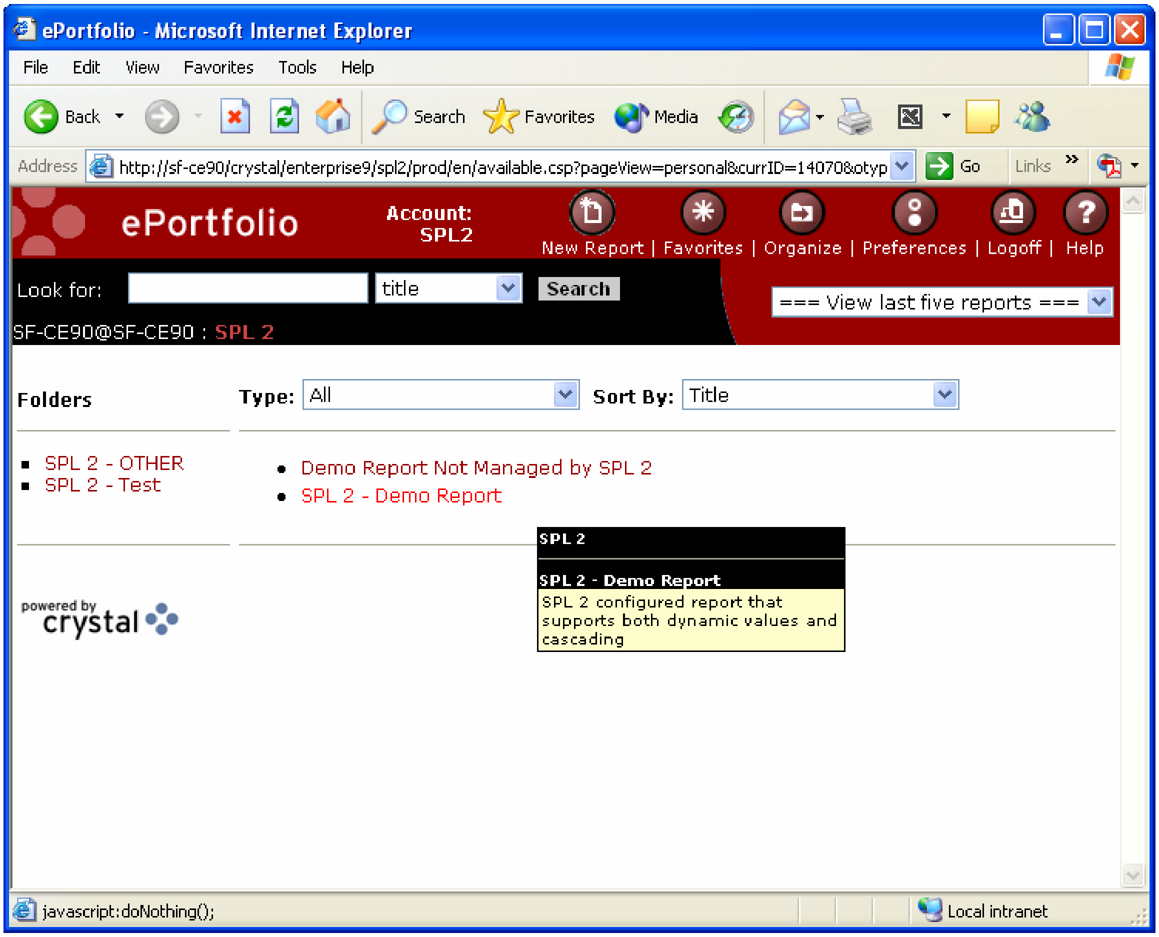
- Style 4 – Based on the wizard style 3, but makes better use of screen real estate. Each parameter's available values are displayed on the left and the current values are displayed on the right
- Style 5 – A wizard style that is unique to SPL. This web frontend aims to strike a balance between having everything displayed on one page and improving the performance by not needing to retrieve all parameter's values in one shot. Unlike the typical wizard (a multiple page interface with back and next buttons) as used by styles 3 and 4, this wizard style displays all parameters on one page but the on the initial page load, only the first parameter's values are retrieved. To apply/retrieve values for the next parameter, a click on Load Next is required
- Style 6 – Based on the wizard style 5, but makes better use of screen real estate Each parameter's available values are displayed on the left and the current values are displayed on the right

For a sampling of web styles, see screen shots:

- Style 1 – chapter 5, Figure 6 on page 33
- Style 2 – chapter 5, Figure 11 on page 39
- Style 3 – chapter 5, Figure 12 on page 40
- Style 6 – chapter 5, Figure 13 on page 41

## 5.2 Sample ePortfolio frontend with SPL integration

When a user requests to Schedule a report with parameters in ePortfolio, the same screen they've grown accustomed to in ePortfolio is still displayed. The moment that the user clicks on Parameters in the drop-down box, the ePortfolio frontend with SPL integration takes over.



Address: http://sf-ce90/crystal/enterprise9/spl2/prod/en/available.csp?pageView=personal&currID=14070&otyp

SF-CE90@SF-CE90 : SPL 2

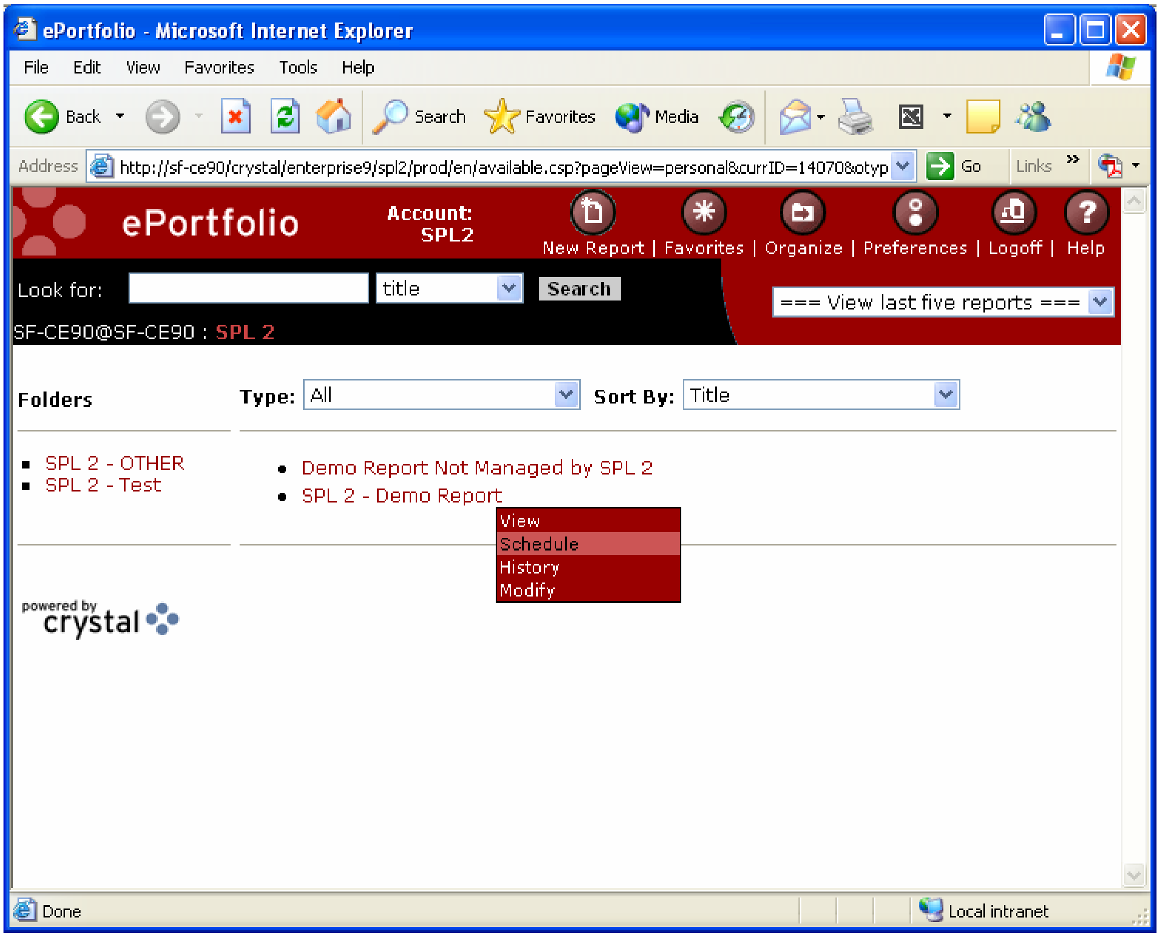
SPL 2

SPL 2 - Demo Report

SPL 2 configured report that supports both dynamic values and cascading

javascript:doNothing();

Figure 2 - ePortfolio's standard virtual folder view.


ePortfolio - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Media
Address
Go
Links
ePortfolio
Account: SPL2
New Report | Favorites | Organize | Preferences | Logoff | Help
Look for:
Search
=== View last five reports ===
Folders
SPL 2 - OTHER
SPL 2 - Test
Demo Report Not Managed by SPL 2
SPL 2 - Demo Report
View
Schedule
History
Modify
powered by crystal
Done
Local intranet

Address: http://sf-ce90/crystal/enterprise9/spl2/prod/en/available.csp?pageView=personal&currID=14070&otyp

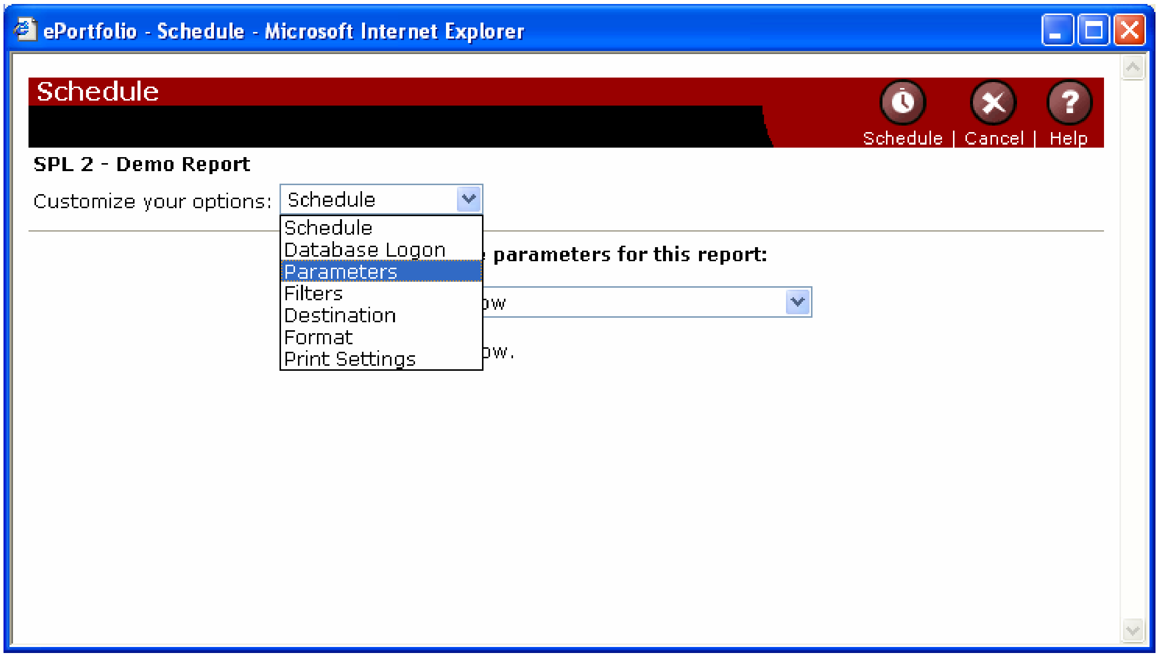
Look for: title

SF-CE90@SF-CE90 : SPL 2

Type: All

Sort By: Title

**Figure 3 – ePortfolio’s standard virtual folder view. SPL 2 – Demo Report was clicked and Schedule selected from Pop-up menu.**



ePortfolio - Schedule - Microsoft Internet Explorer

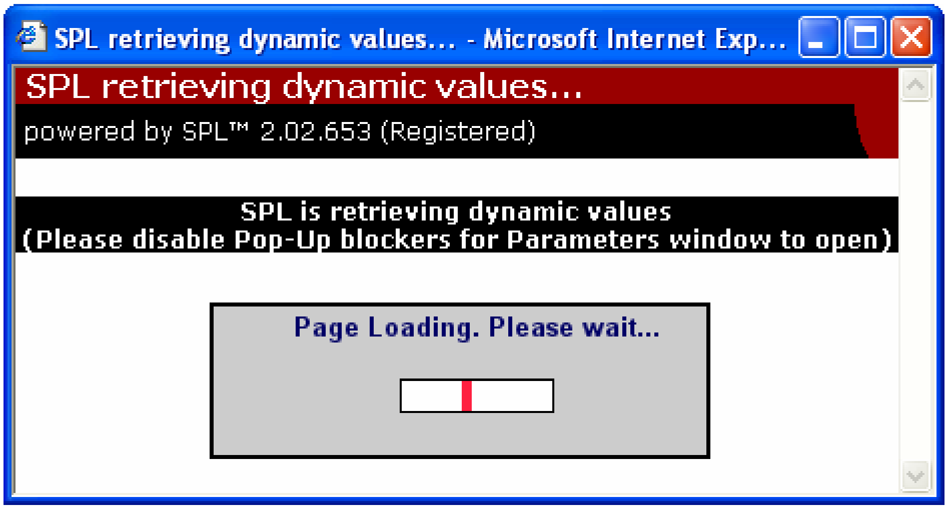
Schedule

SPL 2 - Demo Report

Customize your options: Schedule

parameters for this report:

**Figure 4 - By choosing the 'Parameters' option in the drop-down box, the user invokes the SPL integration which is used to generate the parameters page for the given report instance (as displayed in Figure 5).**



SPL retrieving dynamic values... - Microsoft Internet Exp...

SPL retrieving dynamic values...

powered by SPL™ 2.02.653 (Registered)

SPL is retrieving dynamic values
(Please disable Pop-Up blockers for Parameters window to open)

Page Loading. Please wait...

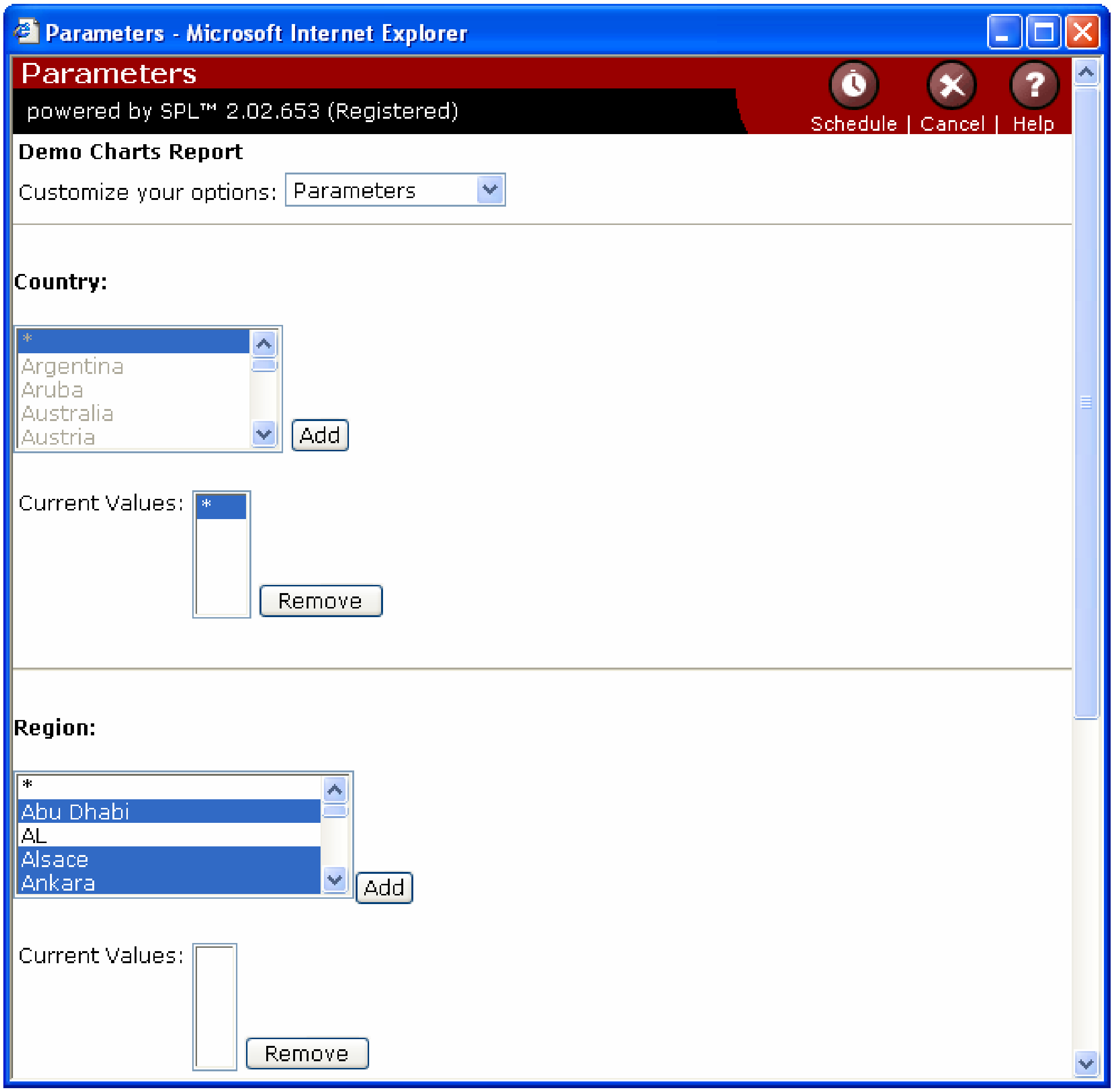
**Figure 5 – If enabled, a Splash screen is displayed while the parameters selection page is being built and automatically closes when the parameters selection page is built. The amount of time necessary to build the parameters selection page depends on the number of parameters, and number of values for each parameter, plus speed of servers, etc. This is useful to inform users that something is loading.**

Figure 6 (below), shows the SPL generated dynamic and cascading parameters page with the standard ePortfolio look and feel (which is using web style 1).

Everything in the frontend is built dynamically and assembled by the SPL backend (including all the titles, labels, controls; such as listbox, combo-box, etc. depending on the type of parameter in the report) using the XML Controls API and XML Control factory.

The example below also uses iFrame to dynamically retrieve values for Region based on the Current Values selected for Country.

This means that without refreshing the entire page, only the small section of the page that contains the Region parameter (specifically the control that holds the available values for Region) makes a server request to retrieve dynamic values based on the constraint selected in Country.



Parameters - Microsoft Internet Explorer

Parameters

powered by SPL™ 2.02.653 (Registered)

**Demo Charts Report**

Customize your options: Parameters

**Country:**

*
Argentina
Aruba
Australia
Austria

Current Values: *

**Region:**

*
Abu Dhabi
AL
Alsace
Ankara

Current Values:

**Figure 6 - SPL ePortfolio integrated parameters page displayed for schedule request (using web style 1). The page is generated dynamically and the parameters representation is unique to each and every report. Thus, the parameters representing a report will be transformed into the controls that are assigned to represent them in Crystal Reports®**

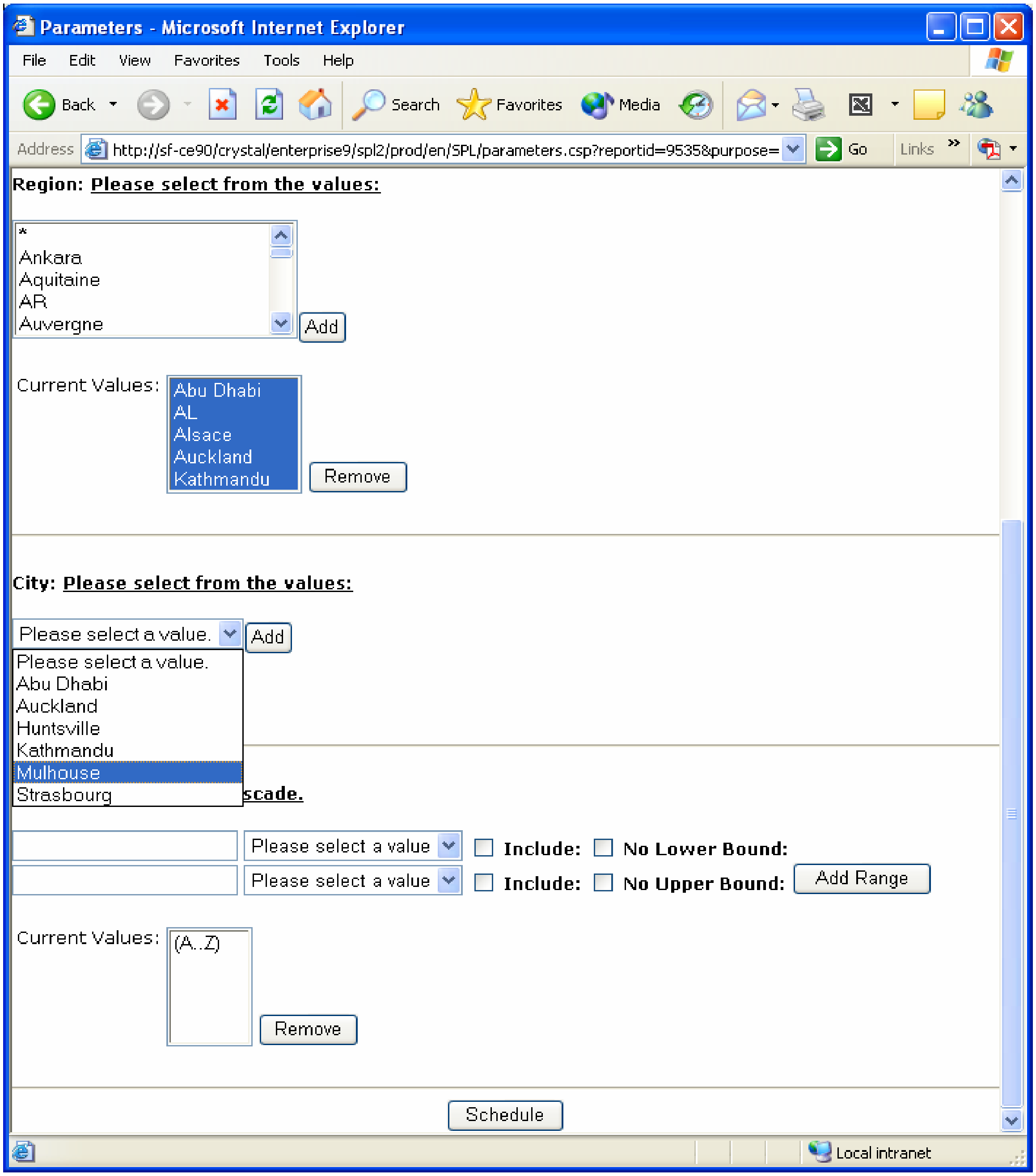
Figure 6 (above) shows an example of:

- dynamic parameter values

- cascade parameters
- iFrame based interface for cascade controls

Parameters “Country” and “Region” are part of a dynamic cascade. The current value(s) selected for “Country” will automatically change the values available for selection in “Region”.

When the available values for “Region” change, only that portion of the screen will be refreshed (instead of the entire page) because an iFrame based control has been rendered for “Region”.



http://sf-ce90/crystal/enterprise9/spl2/prod/en/SPL/parameters.csp?reportid=9535&purpose=

**Region:** **Please select from the values:**

*
Ankara
Aquitaine
AR
Auvergne

Add

Current Values:
Abu Dhabi
AL
Alsace
Auckland
Kathmandu

Remove

**City:** **Please select from the values:**

Please select a value.

Add

Please select a value.
Abu Dhabi
Auckland
Huntsville
Kathmandu
Mulhouse
Strasbourg

**scade.**

Please select a value ☐ **Include:** ☐ **No Lower Bound:**

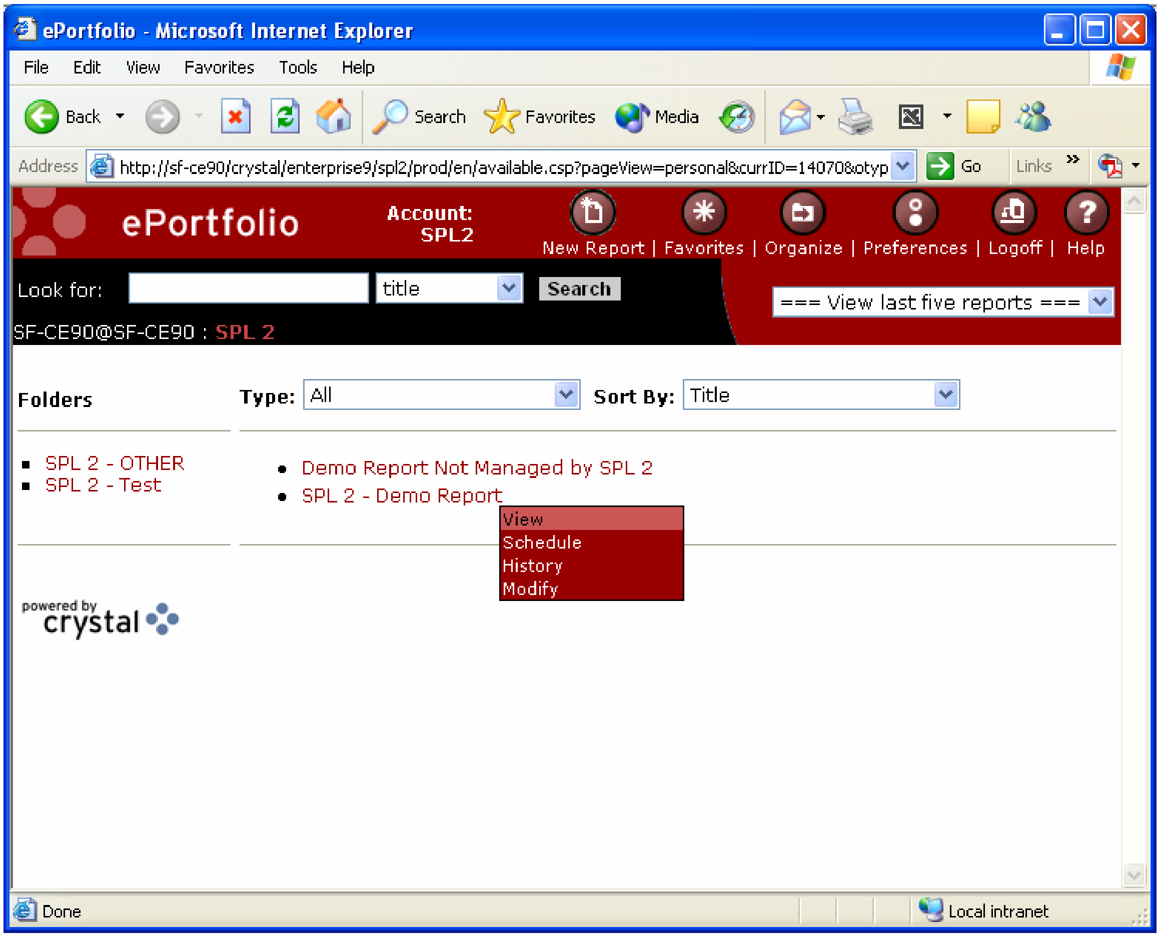
Please select a value ☐ **Include:** ☐ **No Upper Bound:** Add Range

Current Values:
(A..Z)

Remove

Schedule

**Figure 7 - Continuation of the SPL ePortfolio integrated parameters page displayed for schedule request (using web style 1). The page is generated dynamically and the parameters representation is unique to each and every report. Thus, the parameters representing a report will be transformed into the controls that are assigned to represent them in Crystal Reports®**



Address: http://sf-ce90/crystal/enterprise9/spl2/prod/en/available.csp?pageView=personal&currID=14070&otyp

Account: SPL2

Look for: title

=== View last five reports ===

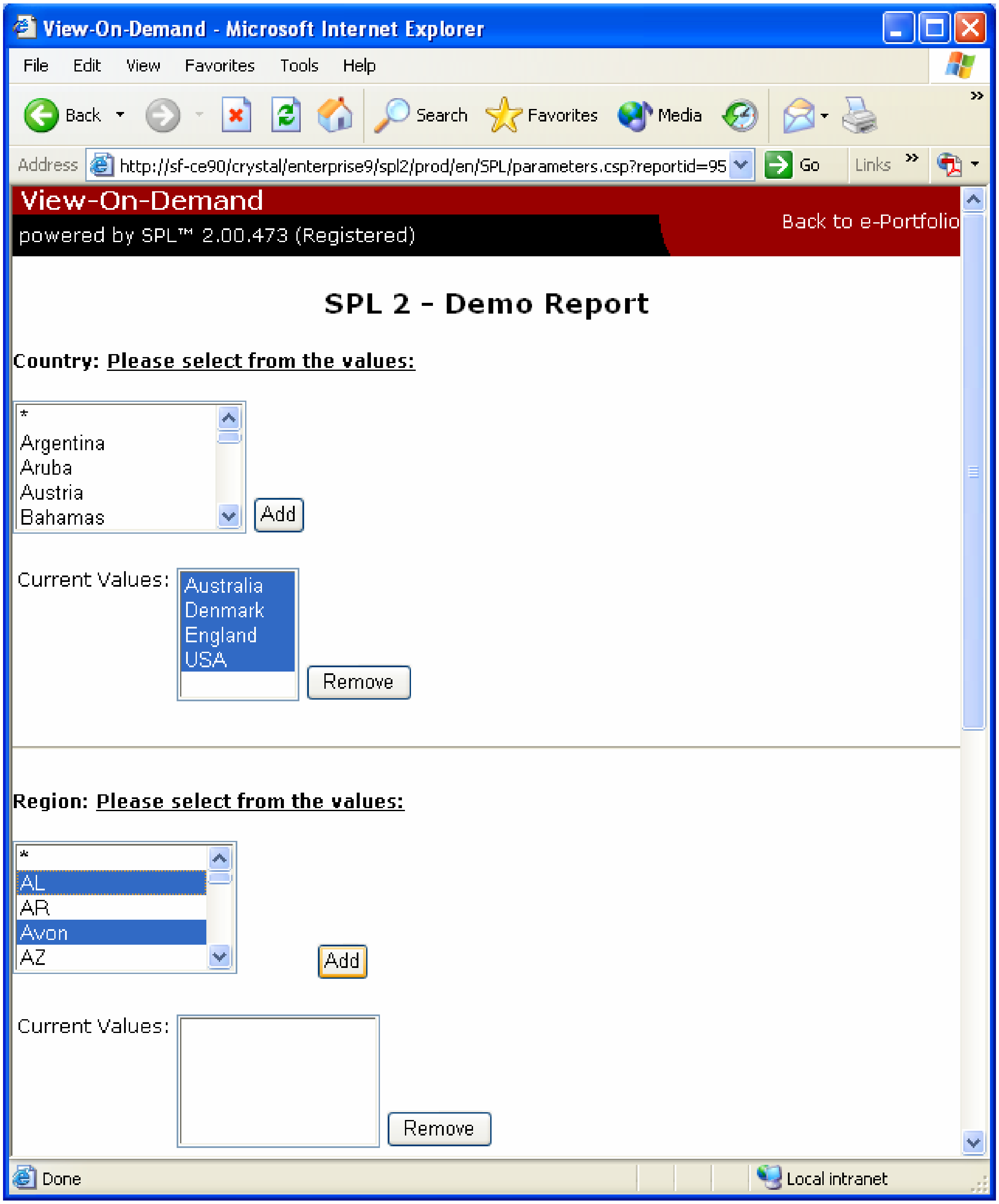
SF-CE90@SF-CE90 : SPL 2

Type: All    Sort By: Title

- SPL 2 - OTHER
- SPL 2 - Test

- Demo Report Not Managed by SPL 2
- SPL 2 - Demo Report

**Figure 8 - In case, the report has to be viewed on demand, and the report has been added to SPL to be managed or configuration is set to use SPL frontend for all reports with parameters, then the same parameters page as that is used when Schedule is selected, will be displayed for reports with parameters when View is selected.**



http://sf-ce90/crystal/enterprise9/spl2/prod/en/SPL/parameters.csp?reportid=95

View-On-Demand

powered by SPL™ 2.00.473 (Registered)

SPL 2 - Demo Report

Country: Please select from the values:

*
Argentina
Aruba
Austria
Bahamas

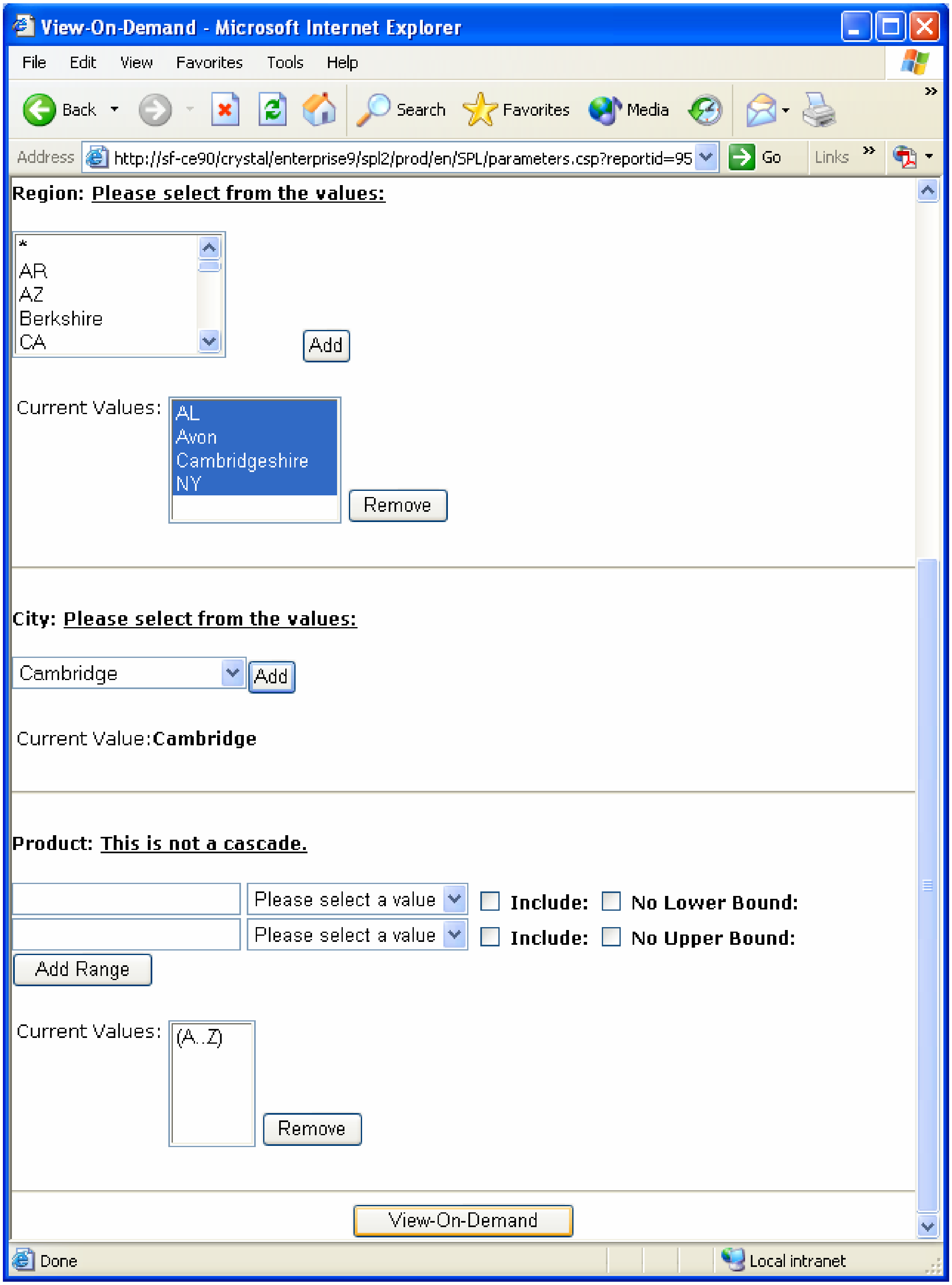
Current Values: Australia
Denmark
England
USA

Region: Please select from the values:

*
AL
AR
Avon
AZ

Current Values:

**Figure 9 – The same report is now displayed for us with View-On-Demand and the labels on the page have been slightly adjusted to reflect this. Also, the schedule button has now become View-On-Demand and the behavior of this button has been adjusted as well.**



http://sf-ce90/crystal/enterprise9/spl2/prod/en/SPL/parameters.csp?reportid=95

**Region:** **Please select from the values:**

*
AR
AZ
Berkshire
CA

Add

Current Values: AL
Avon
Cambridgeshire
NY

Remove

**City:** **Please select from the values:**

Cambridge Add

Current Value:**Cambridge**

**Product:** **This is not a cascade.**

Please select a value ☐ **Include:** ☐ **No Lower Bound:**

Please select a value ☐ **Include:** ☐ **No Upper Bound:**

Add Range

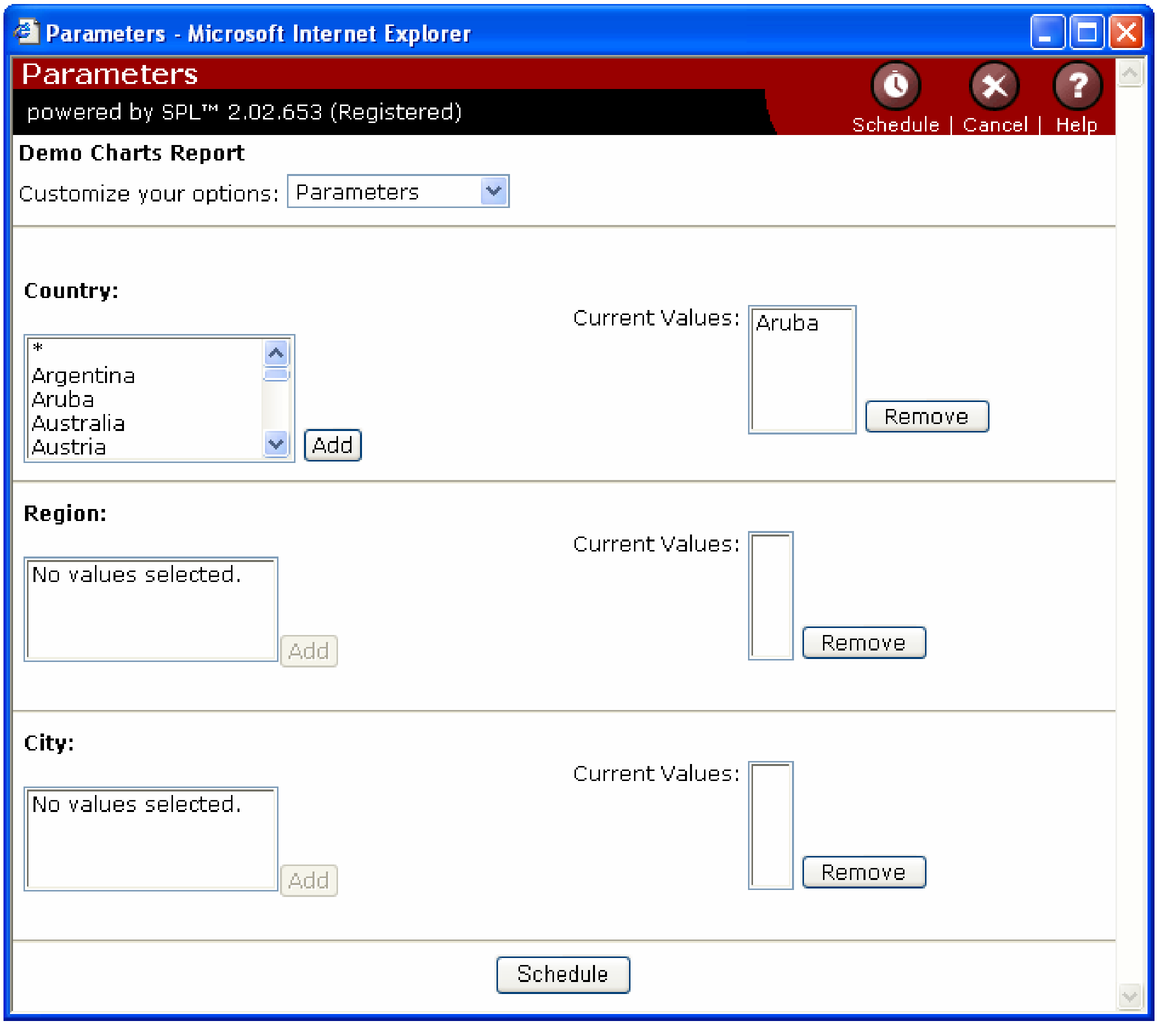
Current Values: (A..Z)

Remove

View-On-Demand

**Figure 10 - Continuation of the SPL ePortfolio integrated parameters page displayed for View-On-Demand.**

### 5.2.1 Highlighted web styles -- 2, 3, and 6 (new as of service pack 2 for SPL 2)



Parameters - Microsoft Internet Explorer

Parameters

powered by SPL™ 2.02.653 (Registered)

Demo Charts Report

Customize your options: Parameters

Country:

Current Values: Aruba

Region:

No values selected.

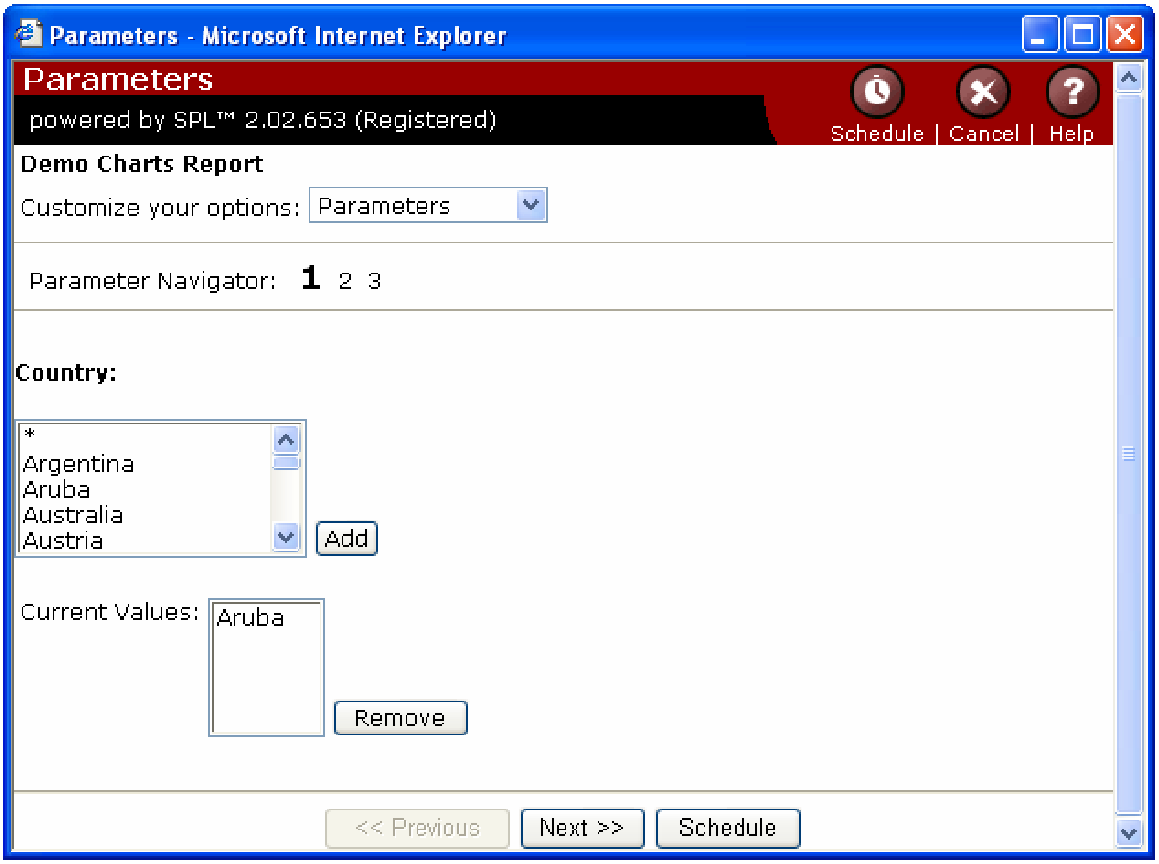
Current Values:

City:

No values selected.

Current Values:

**Figure 11 - SPL ePortfolio integrated parameters page displayed for schedule request (using web style 2). The page is generated dynamically and the parameters representation is unique to each and every report. Thus, the parameters representing a report will be transformed into the controls that are assigned to represent them in Crystal Reports®**



Parameters - Microsoft Internet Explorer

Parameters

powered by SPL™ 2.02.653 (Registered)

Demo Charts Report

Customize your options: Parameters

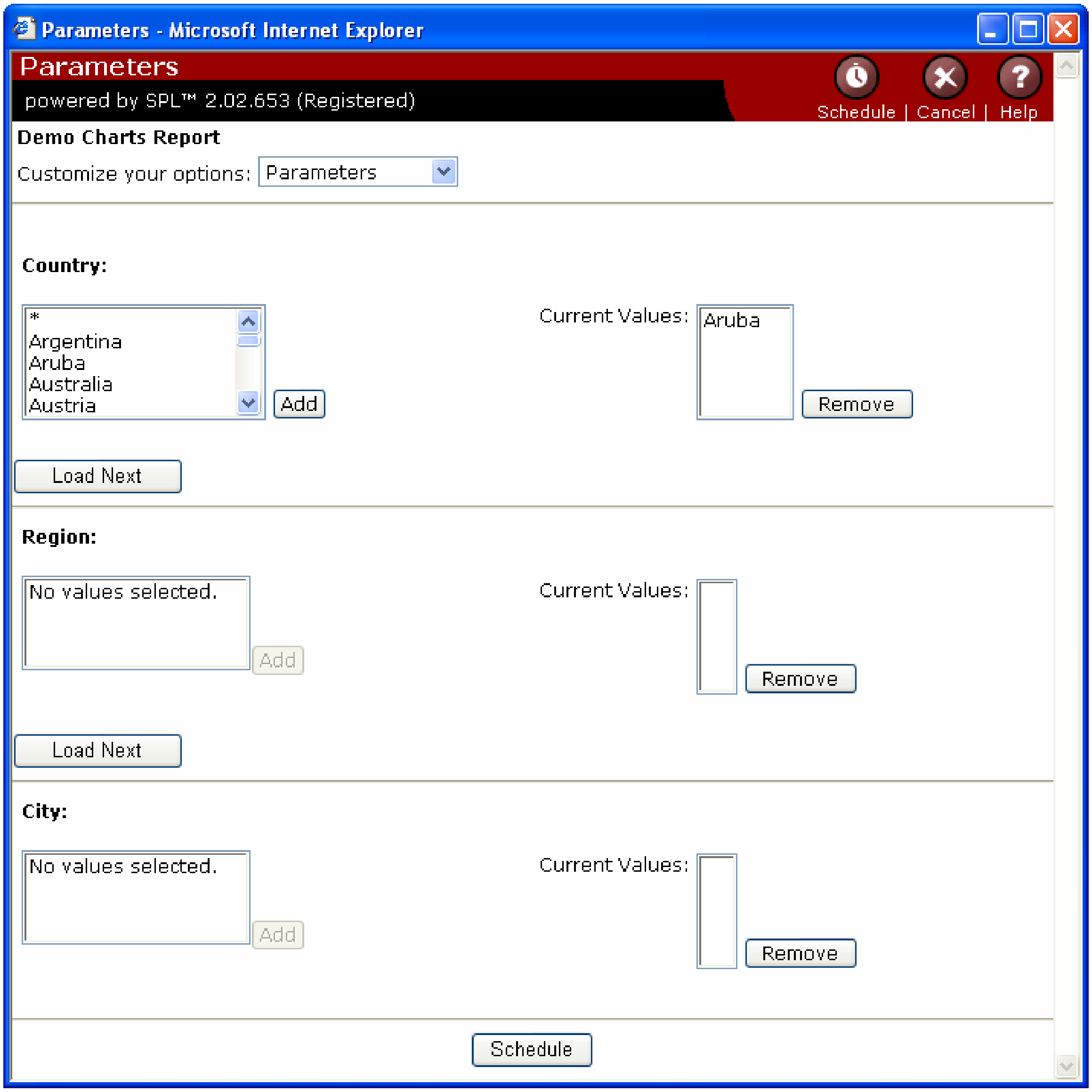
Parameter Navigator: 1 2 3

Country:

*
Argentina
Aruba
Australia
Austria

Current Values: Aruba

**Figure 12 - SPL ePortfolio integrated parameters page displayed for schedule request (using web style 3). The page is generated dynamically and the parameters representation is unique to each and every report. Thus, the parameters representing a report will be transformed into the controls that are assigned to represent them in Crystal Reports®**



Parameters - Microsoft Internet Explorer

Parameters

powered by SPL™ 2.02.653 (Registered)

**Demo Charts Report**

Customize your options: Parameters

**Country:**

*
Argentina
Aruba
Australia
Austria

Current Values: Aruba

**Region:**

No values selected.

Current Values:

**City:**

No values selected.

Current Values:

**Figure 13 - SPL ePortfolio integrated parameters page displayed for schedule request (using web style 6). The page is generated dynamically and the parameters representation is unique to each and every report. Thus, the parameters representing a report will be transformed into the controls that are assigned to represent them in Crystal Reports®**

# Chapter 6

## 6.0 License, Product Activation, and Moving Activated License

SPL is licensed on a per Web Component Server (WCS) basis.

When SPL is installed for the first time, it:

- places itself in to "trial" mode
- associates itself to the specific computer

This "trial" mode allows you to evaluate SPL features for a period of 30-days. After 30-days, SPL will automatically place itself in to "expired" mode and will no longer function.

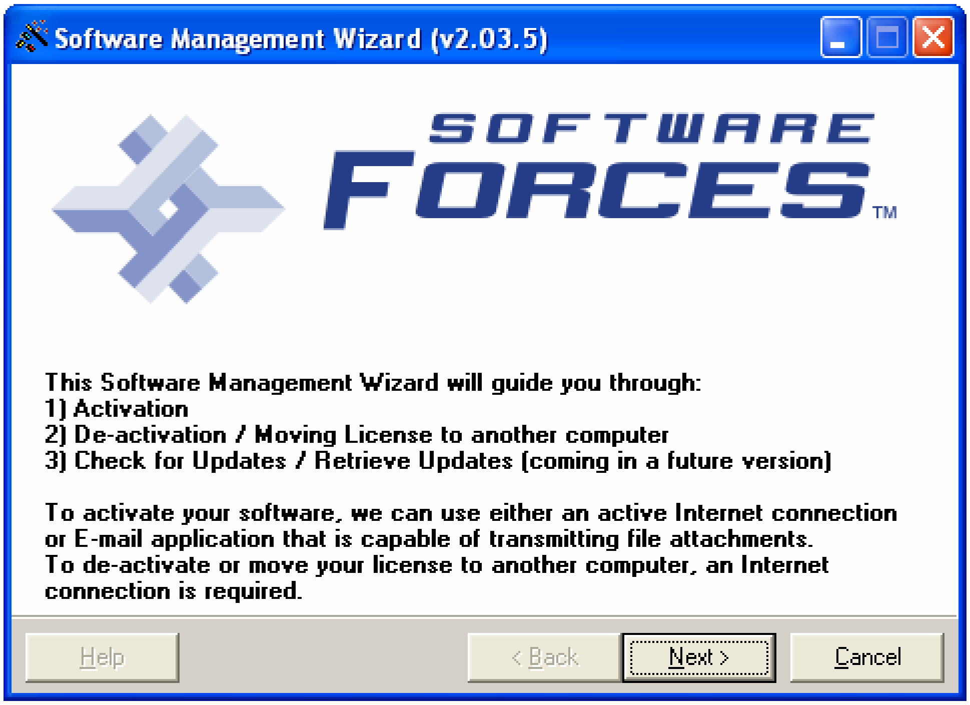
If SPL meets your needs, and you've purchased a license (either before the product expired or after), then you'll need to activate the license on each WCS where SPL is installed.

### 6.1 Purchasing

Several options are available to purchase SPL:

- on-line from Software Forces (via credit card) by visiting www.softwareforces.com/purchase
- off-line from Software Forces (via credit card, purchase order, check, or wire transfer) by calling our Sales department (718.544.7400 option 2 for sales)
- from our resellers (listing available on www.softwareforces.com/purchase)

## 6.2 Software Forces "Software Management Wizard" / Product Activation Wizard



This Software Management Wizard will guide you through:
1) Activation
2) De-activation / Moving License to another computer
3) Check for Updates / Retrieve Updates (coming in a future version)

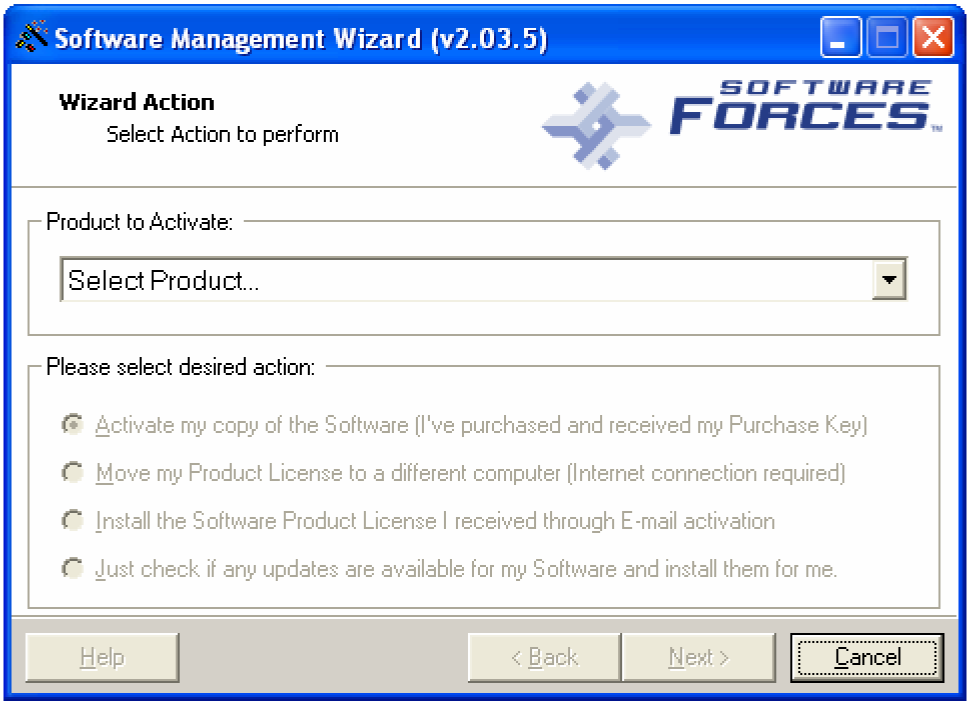
To activate your software, we can use either an active Internet connection or E-mail application that is capable of transmitting file attachments.
To de-activate or move your license to another computer, an Internet connection is required.

**Figure 14 – First screen of the Software Management Wizard**

The Software Forces "Software Management Wizard" is a wizard that is designed to guide you through several product management tasks of an installed Software Forces product. These tasks include:

- product activation
- moving an existing activation to another computer

This wizard ships with all Software Forces products and is installed (by default) in "\Program Files\Software Forces\SFShared" as "SFProdManager.exe". A shortcut to this wizard is created in "start|Programs|Software Forces|SFShared" program group called "SF Product Manager".



Software Management Wizard (v2.03.5)

**Wizard Action**

Select Action to perform

Product to Activate:

Select Product...

Please select desired action:

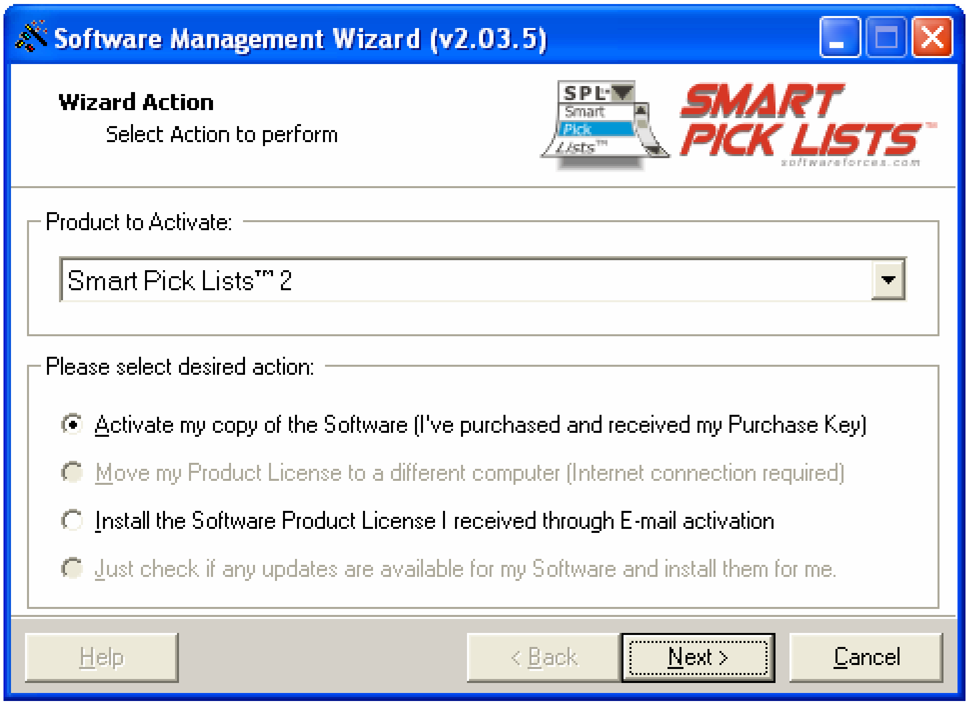
● Activate my copy of the Software (I've purchased and received my Purchase Key)

○ Move my Product License to a different computer (Internet connection required)

○ Install the Software Product License I received through E-mail activation

○ Just check if any updates are available for my Software and install them for me.

**Figure 15 – Second screen of the Software Management Wizard – showing no selected product**



Software Management Wizard (v2.03.5)

**Wizard Action**

Select Action to perform

Product to Activate:

Smart Pick Lists™ 2

Please select desired action:

● Activate my copy of the Software (I've purchased and received my Purchase Key)

○ Move my Product License to a different computer (Internet connection required)

○ Install the Software Product License I received through E-mail activation

○ Just check if any updates are available for my Software and install them for me.

**Figure 16 – Second screen of the Software Management Wizard – showing a selected product and available actions for the product – with “Activate my copy of the Software (I’ve purchased and received my Purchase Key)” selected**

Once this wizard is started, on the second page, there are four (4) choices presented:

- Activate my copy of the Software (I've purchased and received my Purchase Key)
- Move my Product License to a different computer (Internet connection required)
- Install the Software Product License I received through E-Mail registration
- Just check if any updates are available for my Software and install them for me

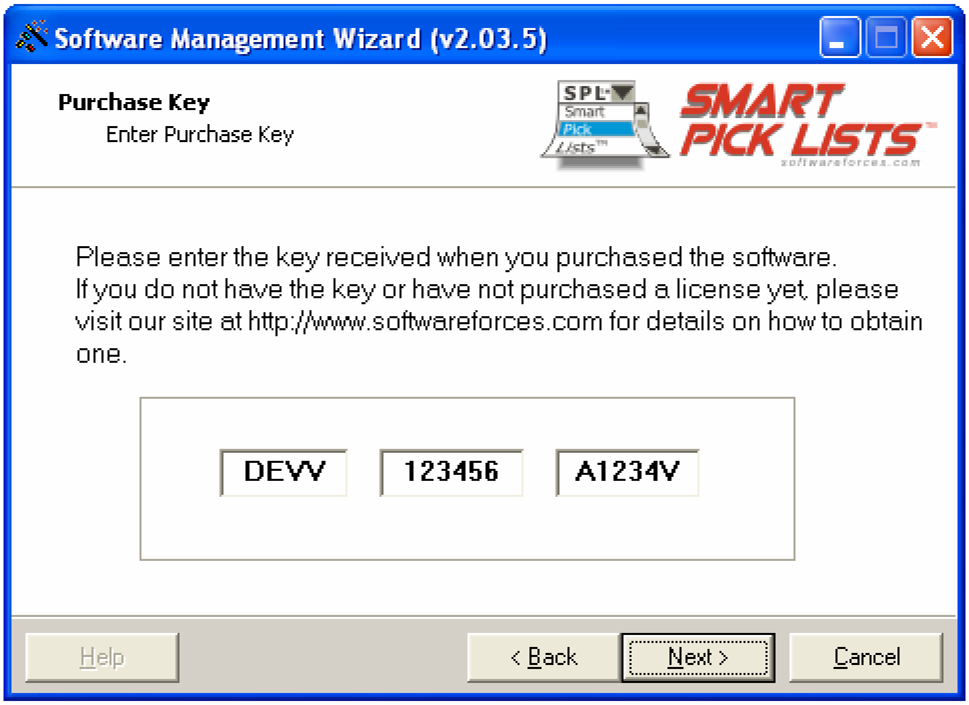
### 6.2.1 Activating your License (via Internet & via E-mail)

Once you have selected the product and the action to perform is selected to "**Activate my copy of the Software (I've purchased and received my Purchase Key)**", click on the Next button.

You will now be prompted to enter the purchase key. The first couple of screens on the wizard are identical for Internet and E-mail activation (see Figure 17 through Figure 21 for details).

Section 6.2.1.1 on page 50, provides details specific to Internet Activation.

Section 6.2.1.2 on page 57, provides details specific to E-mail Activation.



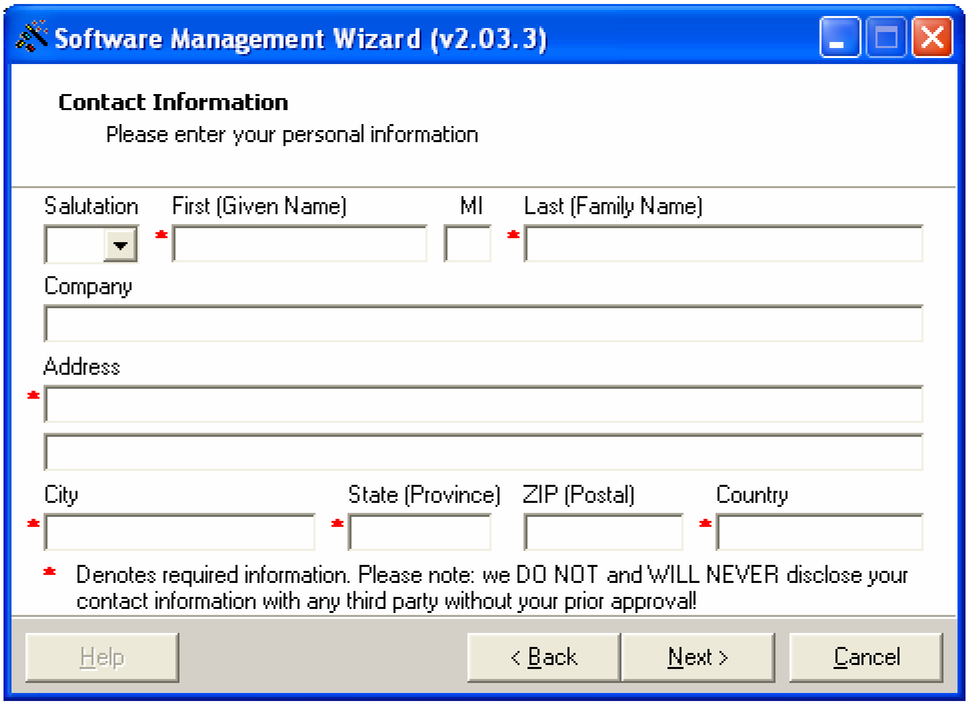
Please enter the key received when you purchased the software.
If you do not have the key or have not purchased a license yet, please visit our site at http://www.softwareforces.com for details on how to obtain one.

DEVV 123456 A1234V

**Figure 17 – Purchase Key entry screen**

When you select to activate your license and click on the Next button, you will then be presented with a screen to enter your purchase key (as shown in Figure 17).

Please do not use/send the purchase key "DEVV-123456-A1234V" which is used in the screen shot(s), as this is not a real key and will only return back with an error.



Software Management Wizard (v2.03.3)

Contact Information

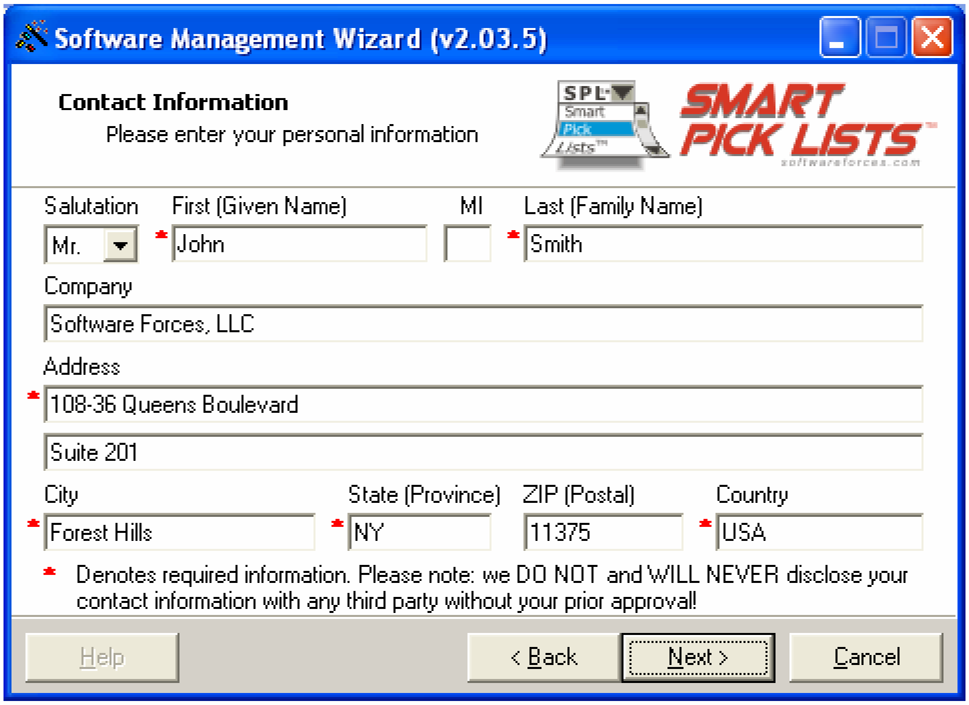
Please enter your personal information

* Denotes required information. Please note: we DO NOT and WILL NEVER disclose your contact information with any third party without your prior approval!

**Figure 18 – Contact Information entry screen, with out values filled in**

After you enter the purchase key and click on the Next button, you will be presented with a form to enter your personal information (as shown in Figure 18).

Please enter the information here of the person that the product should be registered to (as shown in Figure 19). Then click on the Next button to continue.



Software Management Wizard (v2.03.5)

**Contact Information**

Please enter your personal information

Salutation: Mr.

First (Given Name) *: John

MI:

Last (Family Name) *: Smith

Company: Software Forces, LLC

Address *: 108-36 Queens Boulevard

Suite 201

City *: Forest Hills

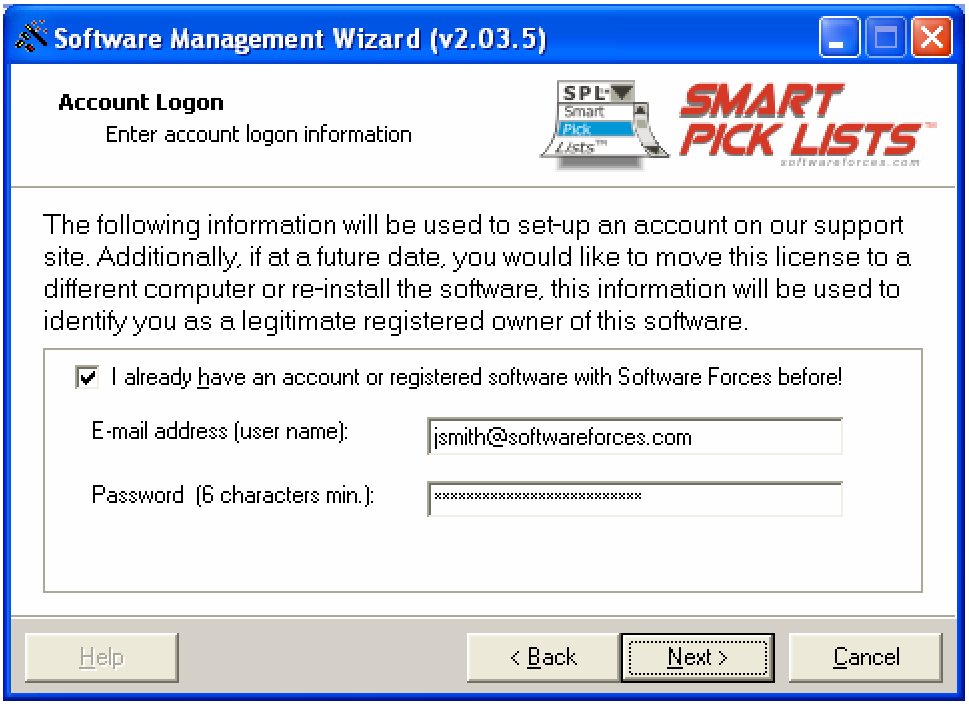
State (Province) *: NY

ZIP (Postal): 11375

Country *: USA

* Denotes required information. Please note: we DO NOT and WILL NEVER disclose your contact information with any third party without your prior approval!

**Figure 19 – Contact Information entry screen – with values filled in**



Software Management Wizard (v2.03.5)

**Account Logon**

Enter account logon information

The following information will be used to set-up an account on our support site. Additionally, if at a future date, you would like to move this license to a different computer or re-install the software, this information will be used to identify you as a legitimate registered owner of this software.

☑ I already have an account or registered software with Software Forces before!

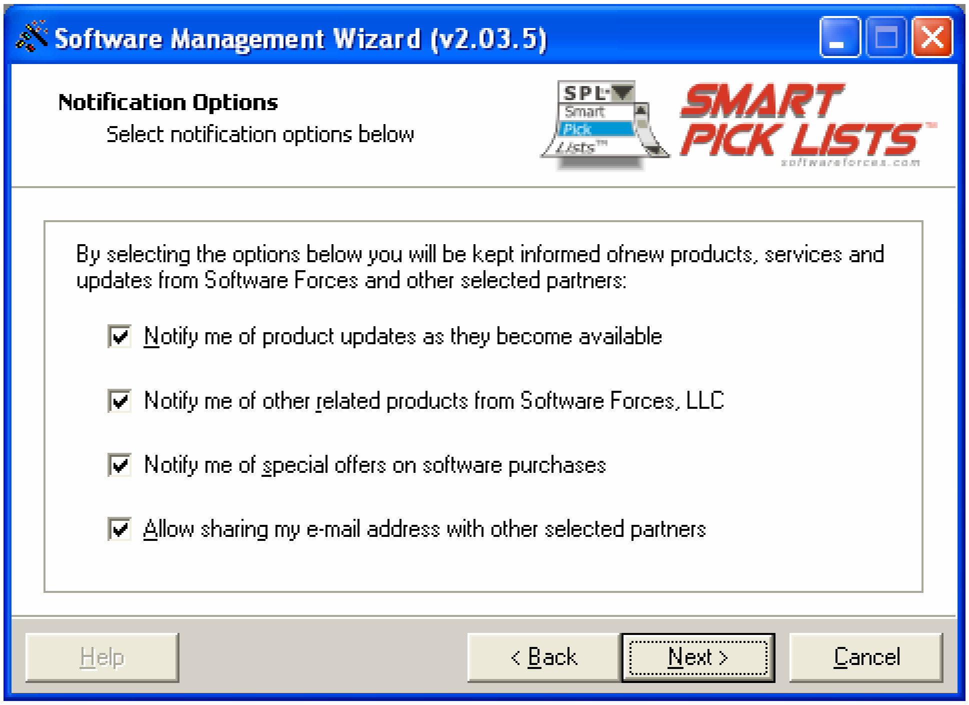
E-mail address (user name): jsmith@softwareforces.com

Password (6 characters min.): ****************************

**Figure 20 – Account Logon / Account Creation entry screen**

After personal / registration information is entered, you will be prompted to enter an E-mail address and password.

If you've already purchased a product from Software Forces before and have activated it, please specify the E-mail address and password that you used. Otherwise, please enter an E-mail address and password that you would like our system to retain for future activations and de-activations (as shown in Figure 20) and click on the Next button.



**Notification Options**

Select notification options below

By selecting the options below you will be kept informed ofnew products, services and updates from Software Forces and other selected partners:

☑ Notify me of product updates as they become available

☑ Notify me of other related products from Software Forces, LLC

☑ Notify me of special offers on software purchases

☑ Allow sharing my e-mail address with other selected partners

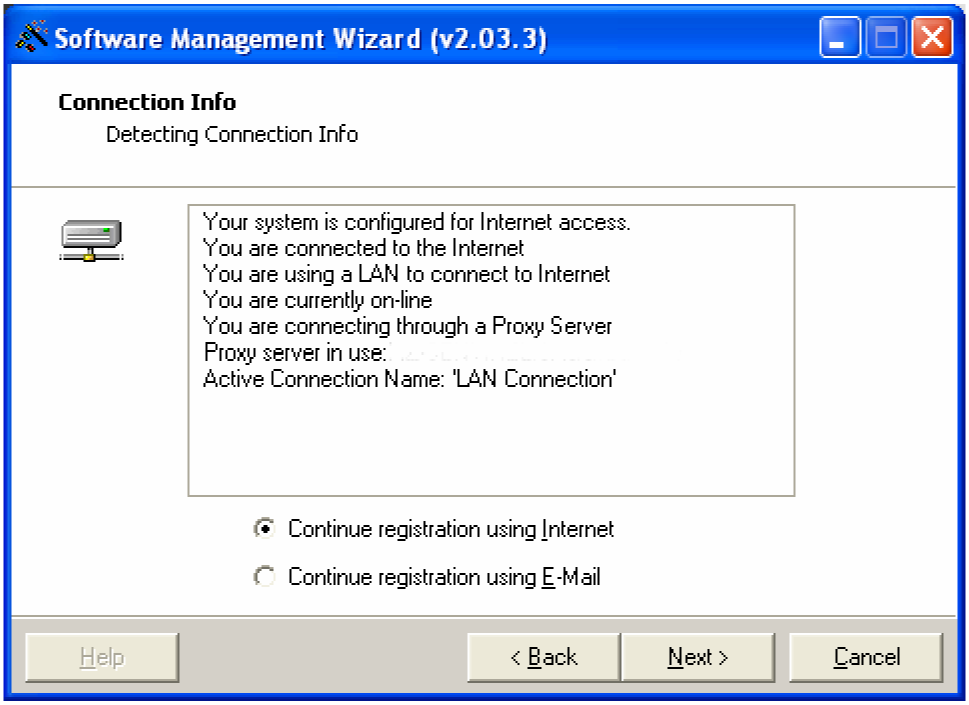
**Figure 21 – Notification Options entry screen**

After providing an E-mail address and Password, you will be prompted to select optional notification options (as shown in Figure 21). Once you select the options, click on the Next button.

For your ease and convenience there are two methods that are provided to activate your license. The first method is to activate via the Internet and the second method is to activate via E-mail.

### 6.2.1.1 Internet Activation

Selecting **Internet Activation** is the recommended method as this will be the fastest to activate.



**Connection Info**

Detecting Connection Info

Your system is configured for Internet access.
You are connected to the Internet
You are using a LAN to connect to Internet
You are currently on-line
You are connecting through a Proxy Server
Proxy server in use: [redacted]
Active Connection Name: 'LAN Connection'

● Continue registration using Internet

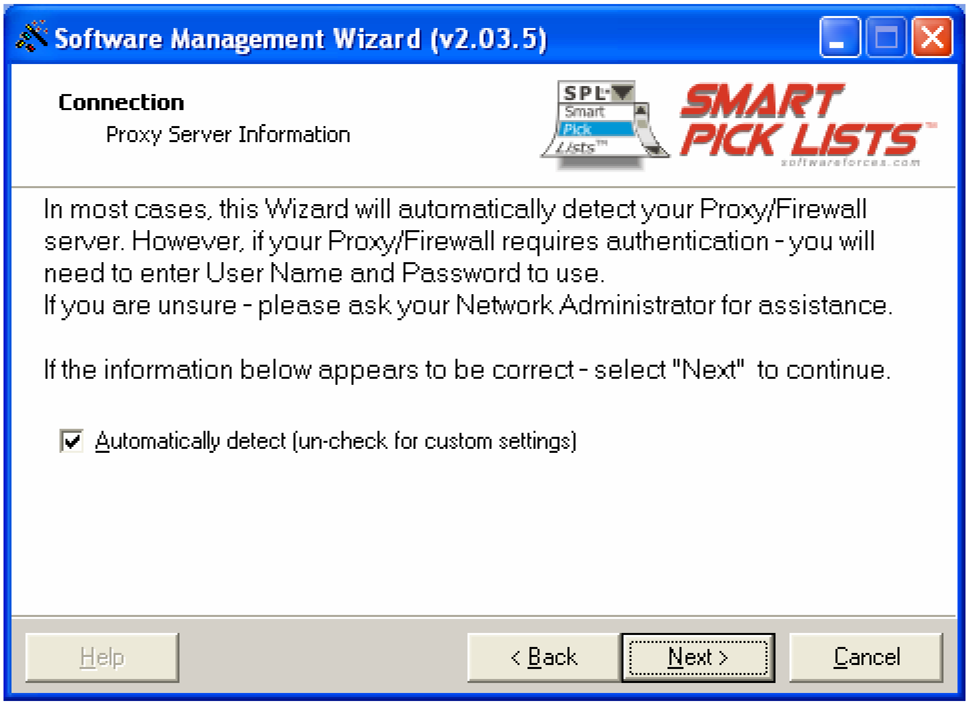
○ Continue registration using E-Mail

**Figure 22 – Connection Info, with Internet Connection selected**

After providing the notification options (as shown in Figure 21), you are shown the information we were able to retrieve from your connection settings and you are prompted to select the type of method you will be using to provide the activation information to our servers (as shown in Figure 22).

Since this section describes the steps for Internet activation, click on “Continue registration using Internet” and then click on the Next button.

If you wish to do E-mail activation, see section 6.2.1.2 on page 57 for details.



**Connection**

Proxy Server Information

In most cases, this Wizard will automatically detect your Proxy/Firewall server. However, if your Proxy/Firewall requires authentication - you will need to enter User Name and Password to use.
If you are unsure - please ask your Network Administrator for assistance.

If the information below appears to be correct - select "Next" to continue.

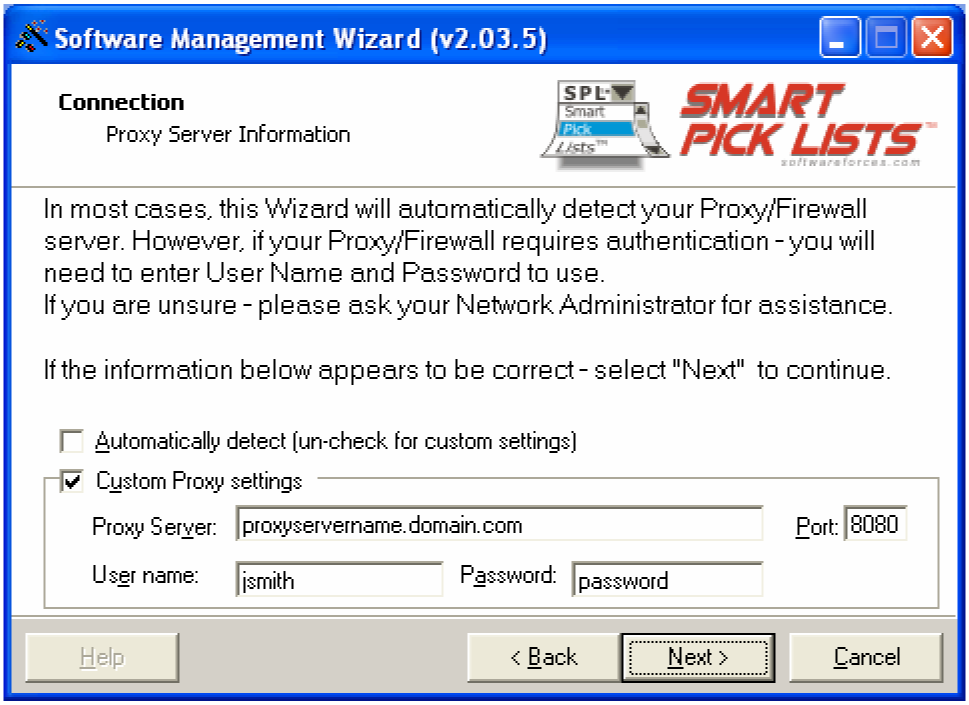
☑ Automatically detect (un-check for custom settings)

**Figure 23 – Connection Proxy Information – with Automatically detect selected**

After selecting to continue with Internet activation, you are prompted with a dialog that warns you about a possible need to authenticate with your firewall/proxy (as shown in Figure 23).

In most cases, it is best to leave “Automatically detect (un-check for custom settings)” selected, and then before clicking on Next button, to first open a browser window and try connecting to the Software Forces web page (www.softwareforces.com).

If your browser asks you to authenticate, then please do so (our connection will reuse that active connection) and then come back to the wizard and continue with its steps.



In most cases, this Wizard will automatically detect your Proxy/Firewall server. However, if your Proxy/Firewall requires authentication - you will need to enter User Name and Password to use.
If you are unsure - please ask your Network Administrator for assistance.

If the information below appears to be correct - select "Next" to continue.

☐ Automatically detect (un-check for custom settings)

☑ Custom Proxy settings

Proxy Server: proxyservername.domain.com Port: 8080

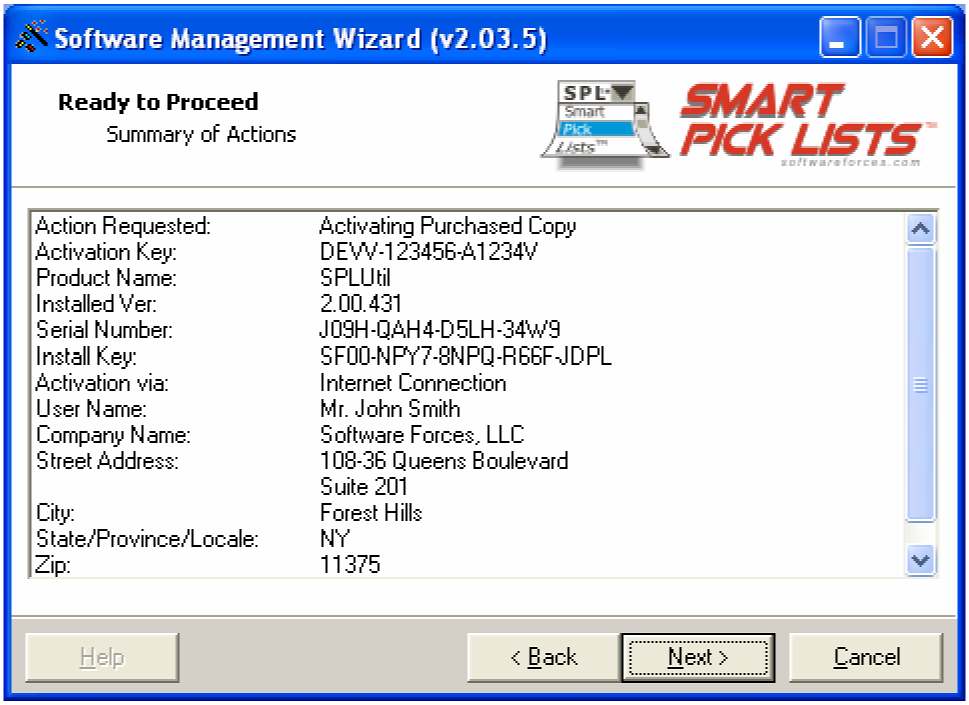
User name: jsmith Password: password

**Figure 24 – Connection Proxy Information – with Automatically detect unselected and Custom Proxy settings revealed**

If the wizard is unable to make a connection and the above described method (setting to “Automatically detect (un-check for custom settings)”) fails, then try turning on “Custom Proxy settings”.

As shown in Figure 24, enter the Proxy Server, Port, User name, and Password (if you are unsure of the appropriate values, contact your network administrator).

If both of these methods fail, and you are still unable to make a connection through the wizard to our activation servers over the Internet, consider trying to do the activation via E-mail (as described in section 6.2.1.2 on page 57).



**Ready to Proceed**

Summary of Actions

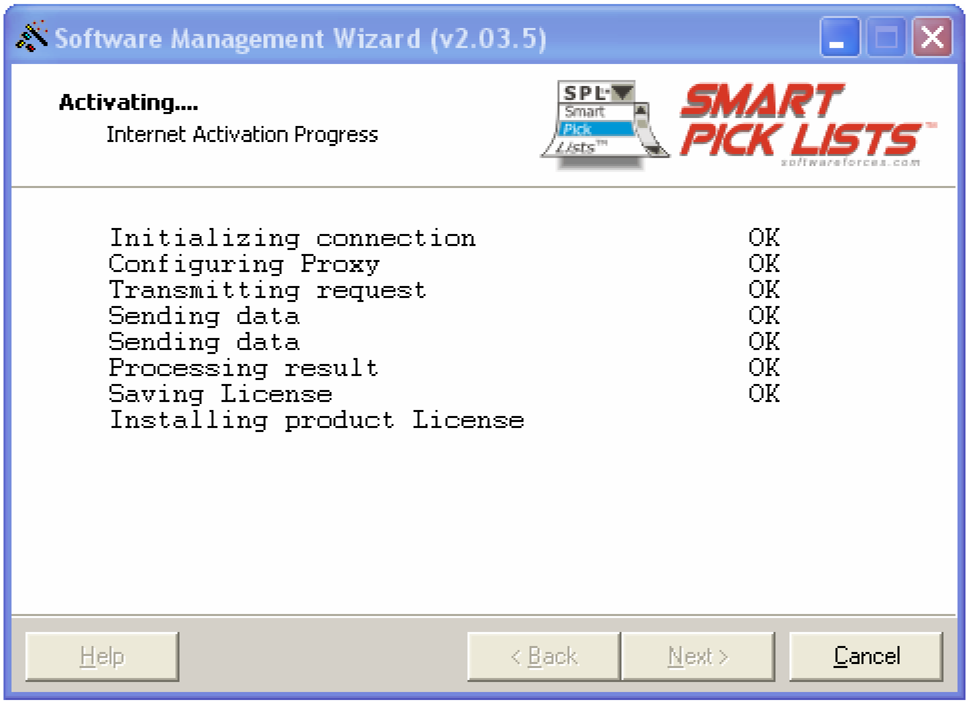
| | |
|---|---|
| Action Requested: | Activating Purchased Copy |
| Activation Key: | DEVV-123456-A1234V |
| Product Name: | SPLUtil |
| Installed Ver: | 2.00.431 |
| Serial Number: | J09H-QAH4-D5LH-34W9 |
| Install Key: | SF00-NPY7-8NPQ-R66F-JDPL |
| Activation via: | Internet Connection |
| User Name: | Mr. John Smith |
| Company Name: | Software Forces, LLC |
| Street Address: | 108-36 Queens Boulevard<br>Suite 201 |
| City: | Forest Hills |
| State/Province/Locale: | NY |
| Zip: | 11375 |

**Figure 25 – Summary of Actions – displaying information gathered from previous screens and showing Activation via Internet Connection**

Once you continue on the firewall/proxy page, you are presented with a summary page (as shown in Figure 25).

When you are ready to transmit the information to our activation server, click on the Next button to get started.

It is only at this time that we make the actual Internet connection and transmit/receive information.



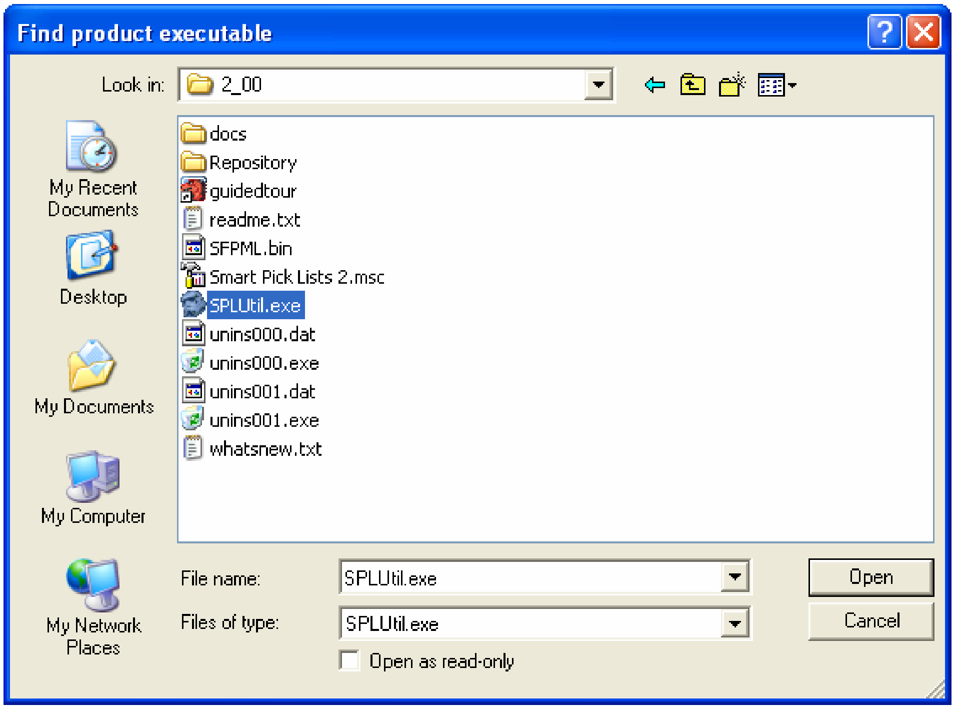
```
Initializing connection          OK
Configuring Proxy                OK
Transmitting request             OK
Sending data                     OK
Sending data                     OK
Processing result                OK
Saving License                   OK
Installing product License
```

**Figure 26 – Internet Activation Progress screen – before product executable was found**

If a connection was established, you should see OK for everything.

At the point of "Installing product License", if the "Software Management Wizard" was unable to locate the product automatically that you are activating (on your computer), a browse dialog will be opened (as shown in Figure 27).

There you can navigate to select the executable in the path that it resides, so that the wizard can continue with the activation procedure.



Find product executable

Look in: 2_00

File name: SPLUtil.exe

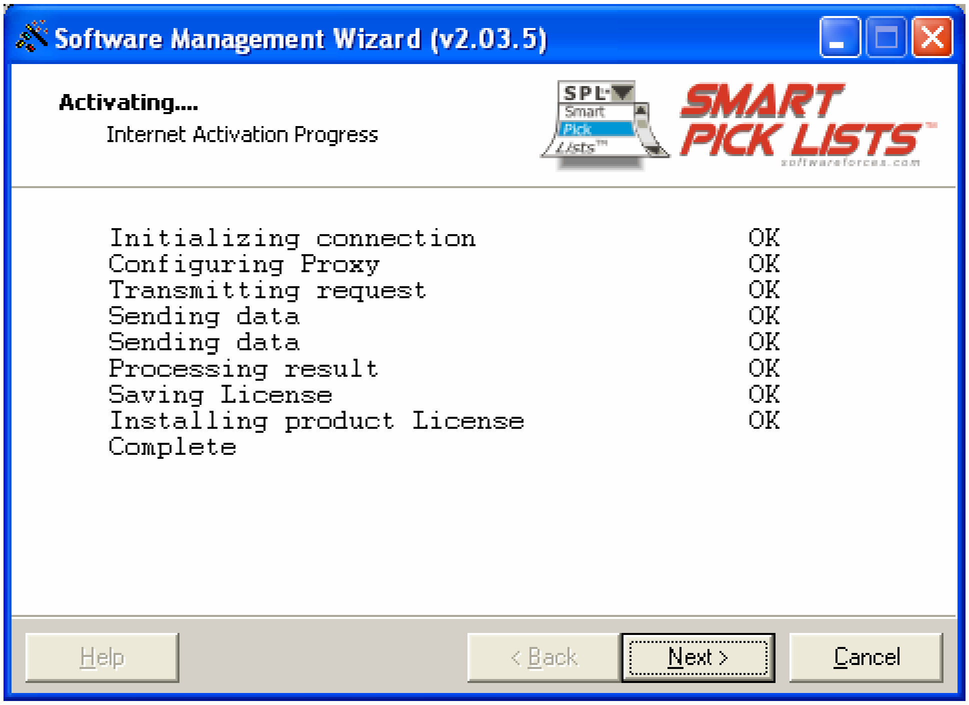
Files of type: SPLUtil.exe

☐ Open as read-only

**Figure 27 – Browse dialog opened to search for product executable, if it was not automatically found**

The dialog displayed in Figure 27 is only displayed if the Software Management Wizard was unable to locate the product automatically that you are activating.

If this dialog is opened, navigate to the path where the product's executable ("**SPLUtil.exe**") resides, so that the wizard can continue with the activation procedure.



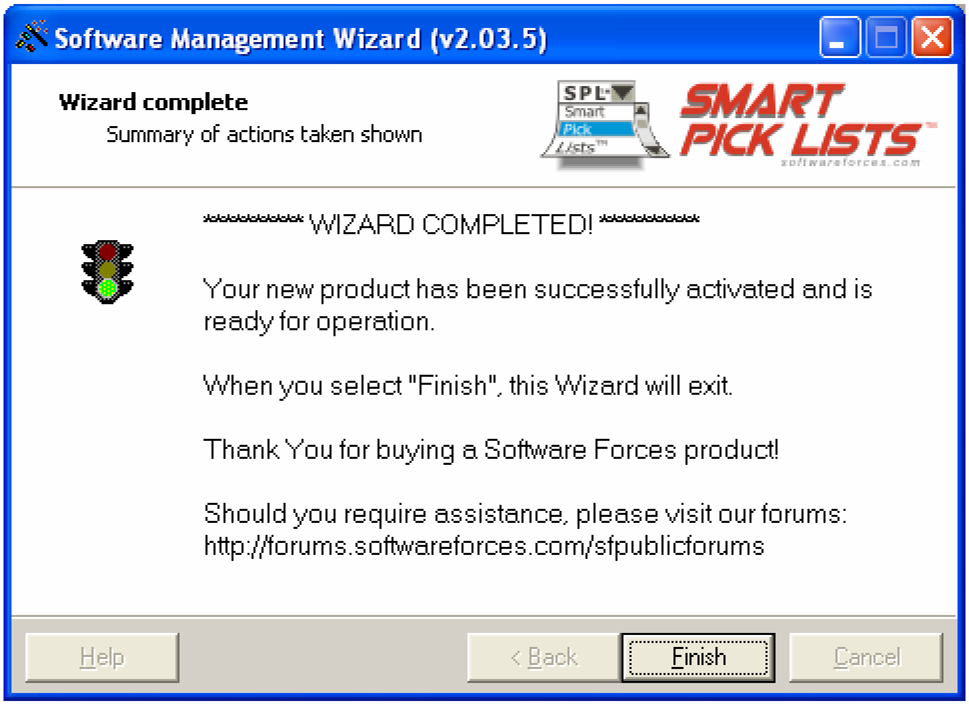
**Activating....**

Internet Activation Progress

```
Initializing connection            OK
Configuring Proxy                  OK
Transmitting request               OK
Sending data                       OK
Sending data                       OK
Processing result                  OK
Saving License                     OK
Installing product License         OK
Complete
```

**Figure 28 – Internet Activation Progress screen – after product executable was found**

Once the product was located (either automatically, or via the browse dialog as shown in Figure 27), you should see "Installing product License" turn to "OK" and receive a "Complete" (as shown in Figure 28). At this point you should click on the Next button.



**Wizard complete**

Summary of actions taken shown

********** WIZARD COMPLETED! **********

Your new product has been successfully activated and is ready for operation.

When you select "Finish", this Wizard will exit.

Thank You for buying a Software Forces product!

Should you require assistance, please visit our forums: http://forums.softwareforces.com/sfpublicforums

**Figure 29 – Wizard complete and product activated screen**

If you receive the "WIZARD COMPLETED!" screen (as shown in Figure 29) you have now successfully completed the activation procedure and your product has been put in to registered mode.

Should you need to de-activate the license on this computer or move it to another computer, please review section 6.2.2 on page 68 for details.

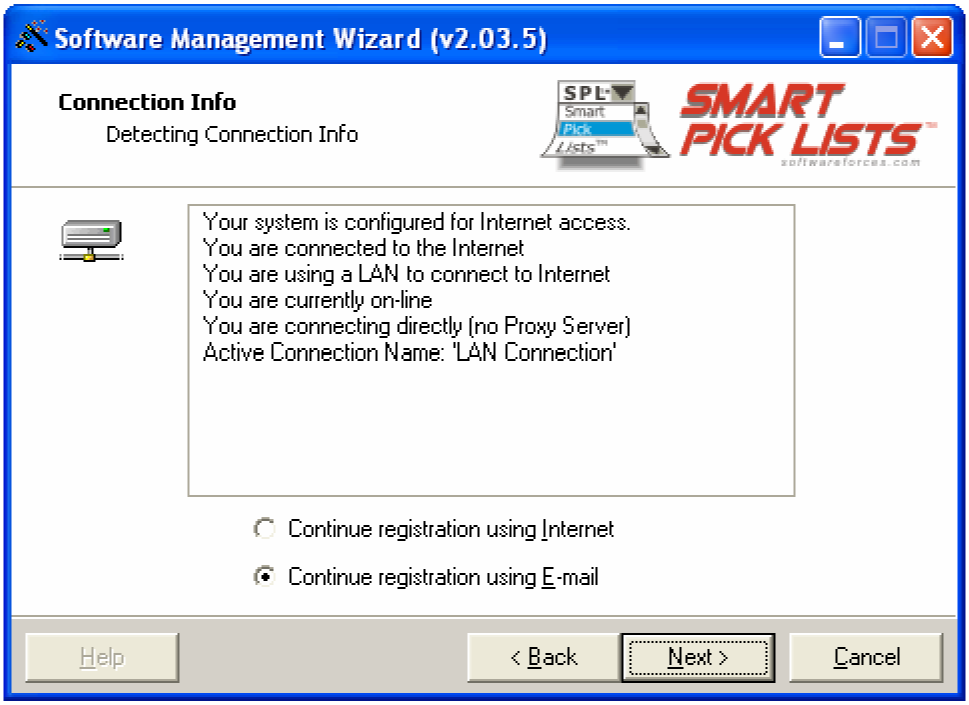
#### 6.2.1.2 E-mail Activation

Selecting **E-mail Activation** is recommended in situations where an Internet connection is unavailable or in situations where an Internet Activation is failing.

Proceeding with E-mail Activation will require you to send an attachment generated by the Software Management Wizard. This attachment contains a certificate specific to your installation.

Once our activation server receives your E-mail with the attachment, it will take up to 30 minutes (on average) to process your request.

If the received attachment is verified to be valid, we will then respond to your E-mail request by sending you a file which you will need to provide to the activation wizard to complete the E-mail Activation process.


Software Management Wizard (v2.03.5)
Connection Info
Detecting Connection Info
SMART PICK LISTS
Help
< Back
Next >
Cancel

Your system is configured for Internet access.
You are connected to the Internet
You are using a LAN to connect to Internet
You are currently on-line
You are connecting directly (no Proxy Server)
Active Connection Name: 'LAN Connection'

○ Continue registration using Internet

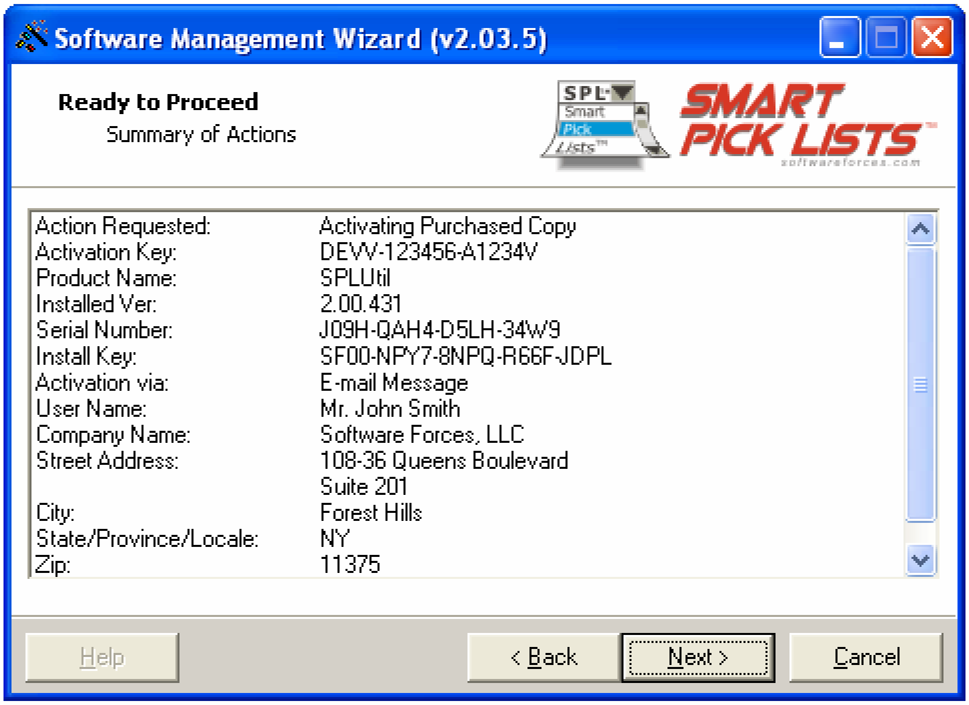
● Continue registration using E-mail

**Figure 30 – Connections Info, with E-mail Connection selected**

After providing the notification options (as shown in Figure 21), you are shown the information we were able to retrieve from your connection settings and you are prompted to select the type of method you will be using to provide the activation information to our activation server (as shown in Figure 30).

Since this section describes the steps for E-mail activation, click on “Continue registration using E-Mail” and then click on the Next button.

If you wish to do Internet activation, see section 6.2.1.1 on page 50 for details.



**Ready to Proceed**

Summary of Actions

| | |
|---|---|
| Action Requested: | Activating Purchased Copy |
| Activation Key: | DEVV-123456-A1234V |
| Product Name: | SPLUtil |
| Installed Ver: | 2.00.431 |
| Serial Number: | J09H-QAH4-D5LH-34W9 |
| Install Key: | SF00-NPY7-8NPQ-R66F-JDPL |
| Activation via: | E-mail Message |
| User Name: | Mr. John Smith |
| Company Name: | Software Forces, LLC |
| Street Address: | 108-36 Queens Boulevard<br>Suite 201 |
| City: | Forest Hills |
| State/Province/Locale: | NY |
| Zip: | 11375 |

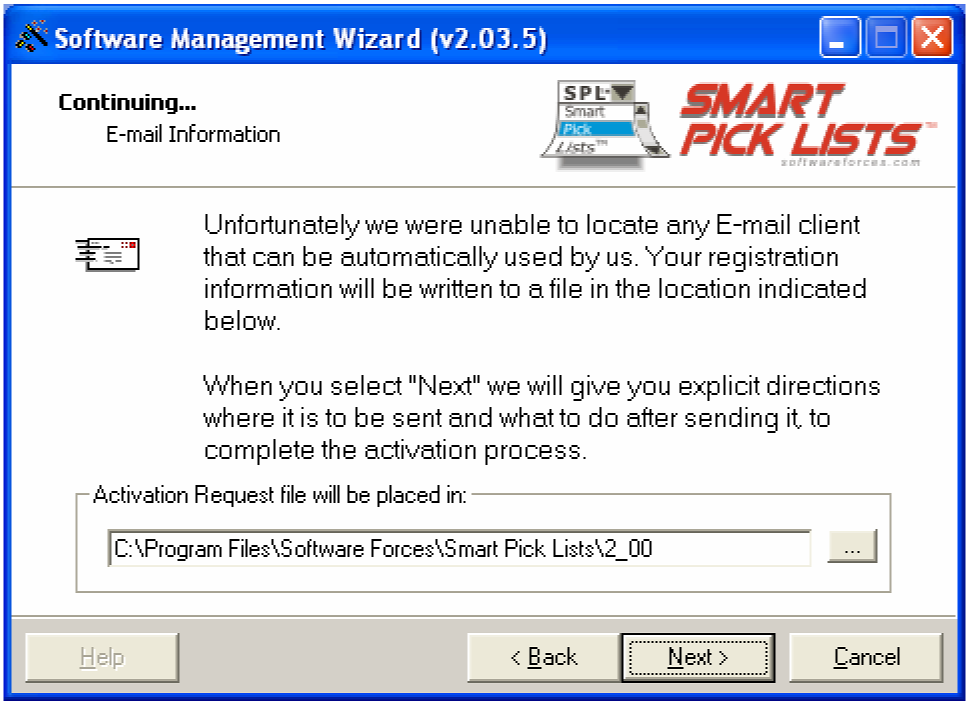
**Figure 31 – Summary of Actions – displaying information gathered from previous screens and showing Activation via E-mail Message**

Once you continue on the connection selection page, you are presented with a summary page (as shown in Figure 31).

When you click on the Next button, an attempt is made to locate and launch a registered E-mail client on your computer.

If we are able to successfully launch your E-mail client, we will try to automatically fill in the “TO:” and “SUBJECT:” portions with the appropriate information and attach a “.dat” file.

This “.dat” file contains the certificate specific to your computer that needs to be sent to our activation server.



**Continuing...**

E-mail Information

Unfortunately we were unable to locate any E-mail client that can be automatically used by us. Your registration information will be written to a file in the location indicated below.

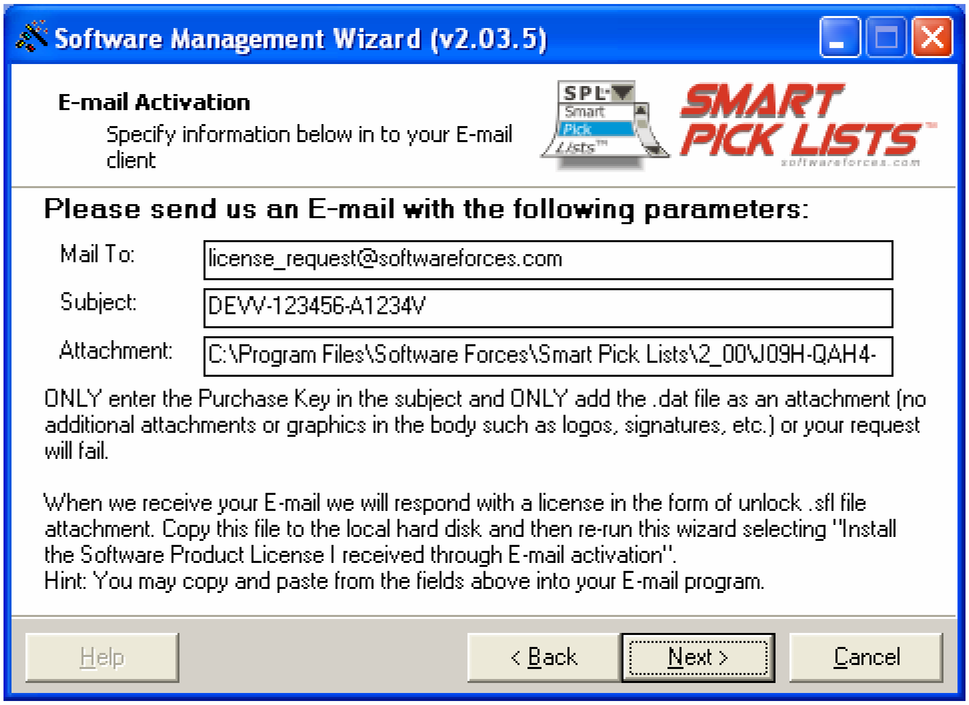
When you select "Next" we will give you explicit directions where it is to be sent and what to do after sending it, to complete the activation process.

Activation Request file will be placed in: C:\Program Files\Software Forces\Smart Pick Lists\2_00

**Figure 32 – E-mail Information screen – unable to automatically locate E-mail client**

If we are unable to locate a registered E-mail client on your computer, we present you with a dialog (as shown in Figure 32) that asks you for a path in which you want the Activation Request file (“.dat”) to be stored (you will later need to attach this file in to your E-mail client and E-mail it to us).

The recommended location to store this file is in your program directory for the product.



E-mail Activation

Specify information below in to your E-mail client

Please send us an E-mail with the following parameters:

Mail To: license_request@softwareforces.com

Subject: DEVV-123456-A1234V

Attachment: C:\Program Files\Software Forces\Smart Pick Lists\2_00\J09H-QAH4-

ONLY enter the Purchase Key in the subject and ONLY add the .dat file as an attachment (no additional attachments or graphics in the body such as logos, signatures, etc.) or your request will fail.

When we receive your E-mail we will respond with a license in the form of unlock .sfl file attachment. Copy this file to the local hard disk and then re-run this wizard selecting "Install the Software Product License I received through E-mail activation".
Hint: You may copy and paste from the fields above into your E-mail program.

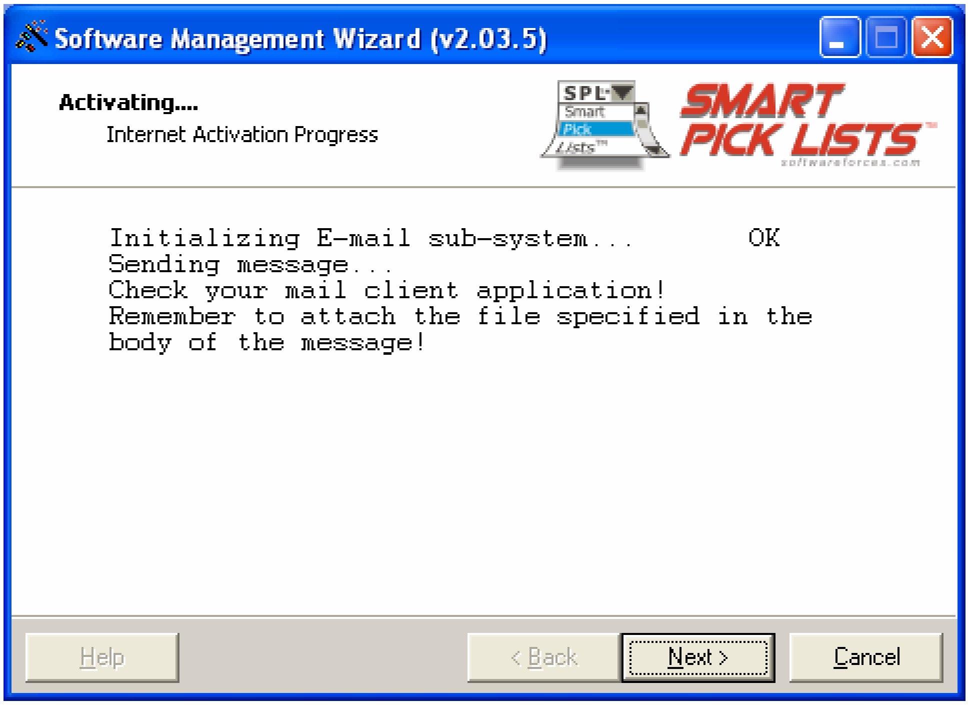
**Figure 33 – E-mail Activation screen – manual E-mail instructions and parameters**

If an E-mail client was not detected automatically, you will be shown the necessary information to manually send an E-mail with the attachment (as shown in Figure 33) to our activation server.

Please do not add anything to the subject line other than the purchase key. Also do not attach or paste any additional files (such as logos, signatures, etc.) in the body of your message (other than the ".dat" file created by the wizard) as this will cause the activation server to not process your request.

When you click on the Next button, even though we were unable to locate a registered E-mail client, we will attempt to send an "HTTP mailto" request. In most cases, this automatically launches an E-mail client that is associated with that type of request.

However, with this type of an E-mail client launch, we are unable to pass to the E-mail client the "TO", "SUBJECT" or attachment and you will need to provide the information manually (as shown in Figure 33).

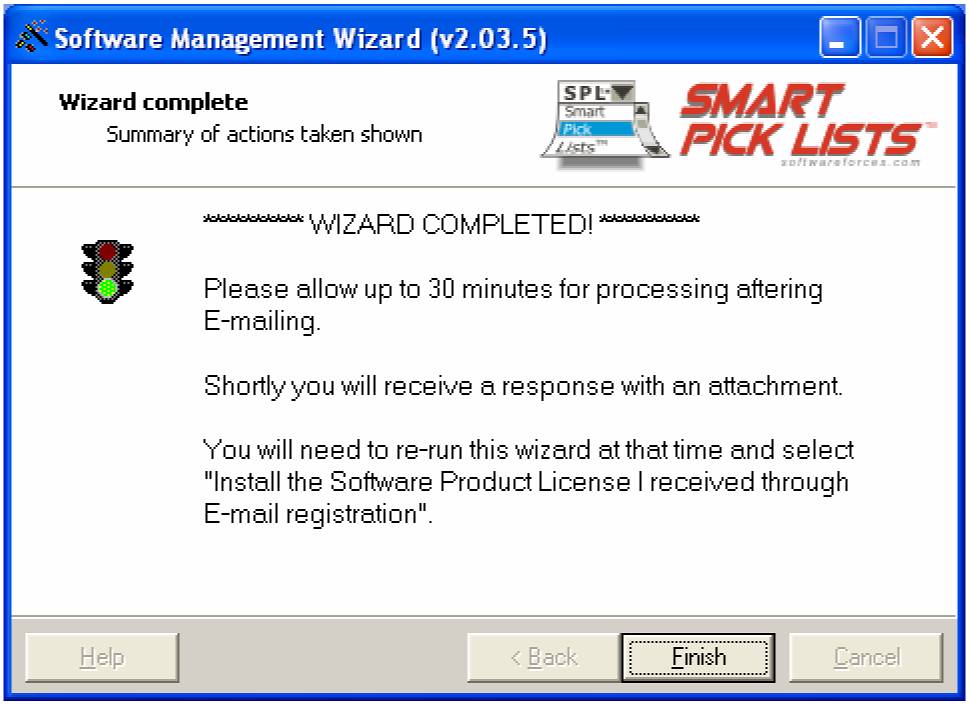


```
Initializing E-mail sub-system...        OK
Sending message...
Check your mail client application!
Remember to attach the file specified in the
body of the message!
```

**Figure 34 – E-mail Activation screen – automatically detect E-mail client**

If we are able to automatically detect a registered E-mail client and pass to it all the necessary information, at this point the E-mail will be submitted (as shown in Figure 34).

If we were unable to detect it, then we simply remind you to attach the file manually to your E-mail client (as shown in Figure 33).



Wizard complete

Summary of actions taken shown

********** WIZARD COMPLETED! **********

Please allow up to 30 minutes for processing aftering E-mailing.

Shortly you will receive a response with an attachment.

You will need to re-run this wizard at that time and select "Install the Software Product License I received through E-mail registration".

**Figure 35 - Wizard complete and instructed to wait for E-mail response**

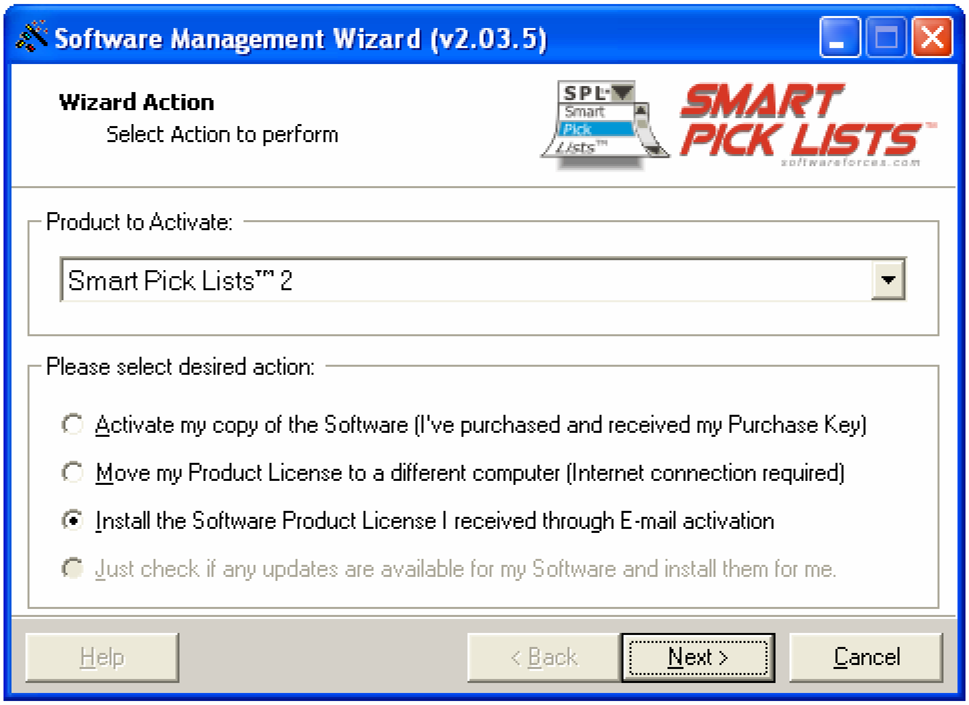
If you receive the "WIZARD COMPLETED!" screen (as shown in Figure 35) you have now successfully completed the first part of the E-mail activation procedure.

Once our activation server receives your E-mail with the attachment, it will take up to 30 minutes (on average) to process your request.

If the received attachment is verified to be valid, we will then respond to your E-mail request by sending you a file as an attachment (".sfl").

You will need to re-run the activation wizard and this time select "Install the Software Product License I received through E-mail activation" (as described in section 6.2.1.3 on page 64 and shown in Figure 36) to complete the E-mail Activation process.

### 6.2.1.3 Installing a License received through E-mail



Software Management Wizard (v2.03.5)

**Wizard Action**

Select Action to perform

Product to Activate: Smart Pick Lists™ 2

Please select desired action:

○ Activate my copy of the Software (I've purchased and received my Purchase Key)

○ Move my Product License to a different computer (Internet connection required)

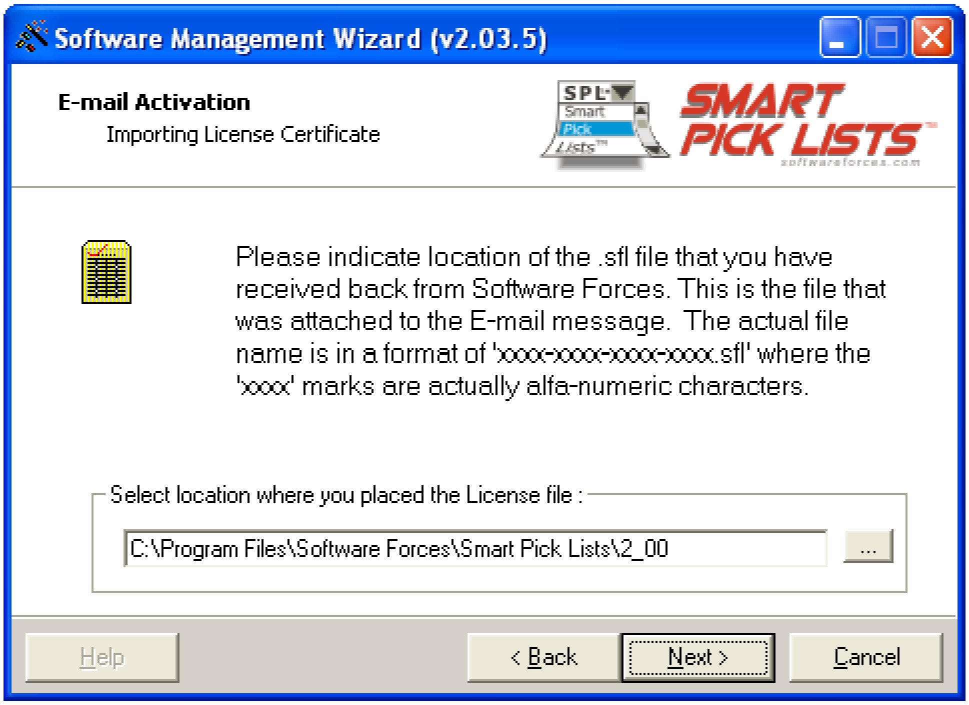
● Install the Software Product License I received through E-mail activation

○ Just check if any updates are available for my Software and install them for me.

**Figure 36 - Second screen of the Software Management Wizard – showing a selected product and available actions for the product, with "Install the Software Product License I received through E-mail activation" selected**

After you have successfully completed the first part of E-mail activation (by selecting "Activate my copy of the Software (I've purchased and received my Purchase Key)" and E-mailed a ".dat" attachment to our activation server), you are ready to continue with the last part of E-mail activation.

Once we respond to your E-mail and you receive a file as an attachment (".sfl"), you will need to re-run the activation wizard, select the same product to activate,  and this time select "Install the Software Product License I received through E-mail activation" (as shown in Figure 36) as your desired action. Then click on the Next button to continue.

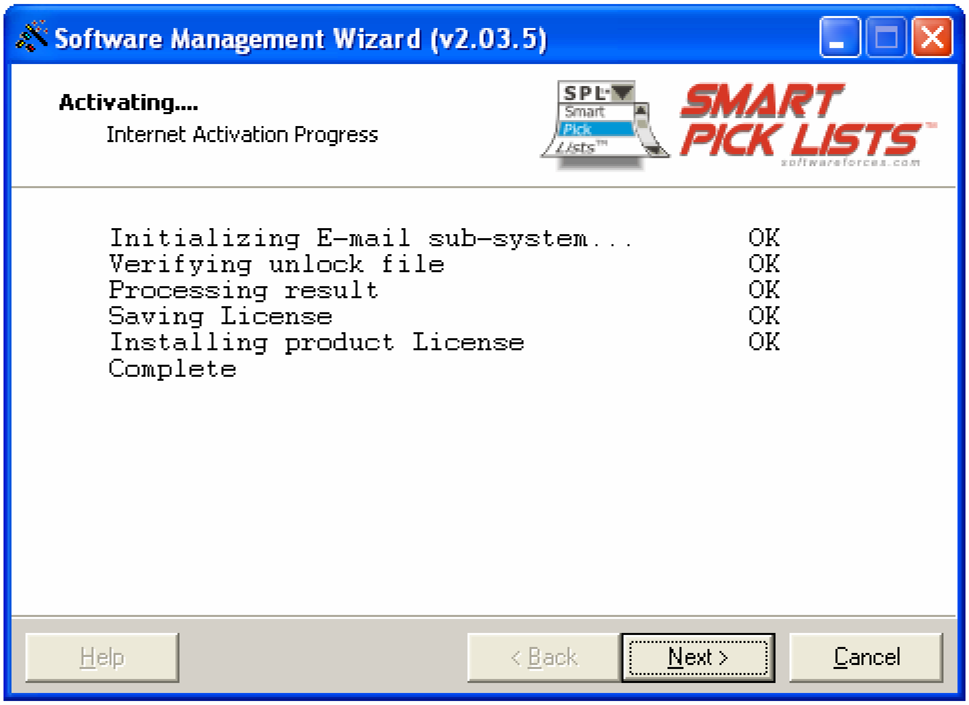


Please indicate location of the .sfl file that you have received back from Software Forces. This is the file that was attached to the E-mail message. The actual file name is in a format of 'xxxx-xxxx-xxxx-xxxx.sfl' where the 'xxxx' marks are actually alfa-numeric characters.

Select location where you placed the License file : C:\Program Files\Software Forces\Smart Pick Lists\2_00

**Figure 37 – E-mail Activation select location of .sfl file (from received E-mail response with .sfl attachment)**

As shown in Figure 37, specify the path to where you saved the ".sfl" file you received as an attachment in your E-mail from us and click on the Next button to continue.



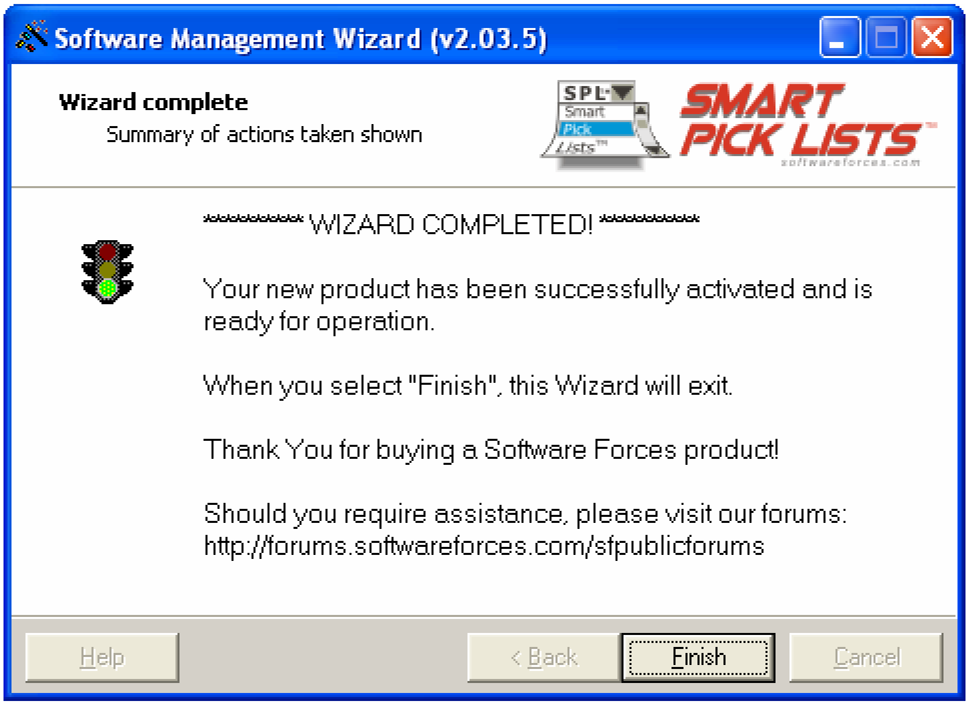
**Activating....**
Internet Activation Progress

```
Initializing E-mail sub-system...        OK
Verifying unlock file                    OK
Processing result                        OK
Saving License                           OK
Installing product License               OK
Complete
```

**Figure 38 – E-mail Activation, processing .sfl file**

If the activation wizard finds the ".sfl" file in the path you specified (as shown in Figure 37), it will begin processing the contents (as shown in Figure 38).

If the certificate that was sent back to you in this ".sfl" is confirmed to be valid for your computer, you will receive "OK" for all actions followed by a "Complete" and should now click on the Next button.



*********** WIZARD COMPLETED! ***********

Your new product has been successfully activated and is ready for operation.

When you select "Finish", this Wizard will exit.

Thank You for buying a Software Forces product!

Should you require assistance, please visit our forums:
http://forums.softwareforces.com/sfpublicforums

**Figure 39 – Wizard complete and product activated screen**

If you receive the “WIZARD COMPLETED!” screen (as shown in Figure 39) you have now successfully completed the activation procedure and your product has been put in to registered mode.

### 6.2.2 Moving your License

After a Software Forces product has been activated, the license is locked to the computer on which it was activated on.

There are however times, when you need to either temporarily remove the license from this computer and then place it back, or to move this license to another computer.

For example, if you are planning to reformat your hard drive or install a new operating system and are not sure of the affects this installation may have on your license, you may wish to remove the license from the computer before you initiate any further actions.

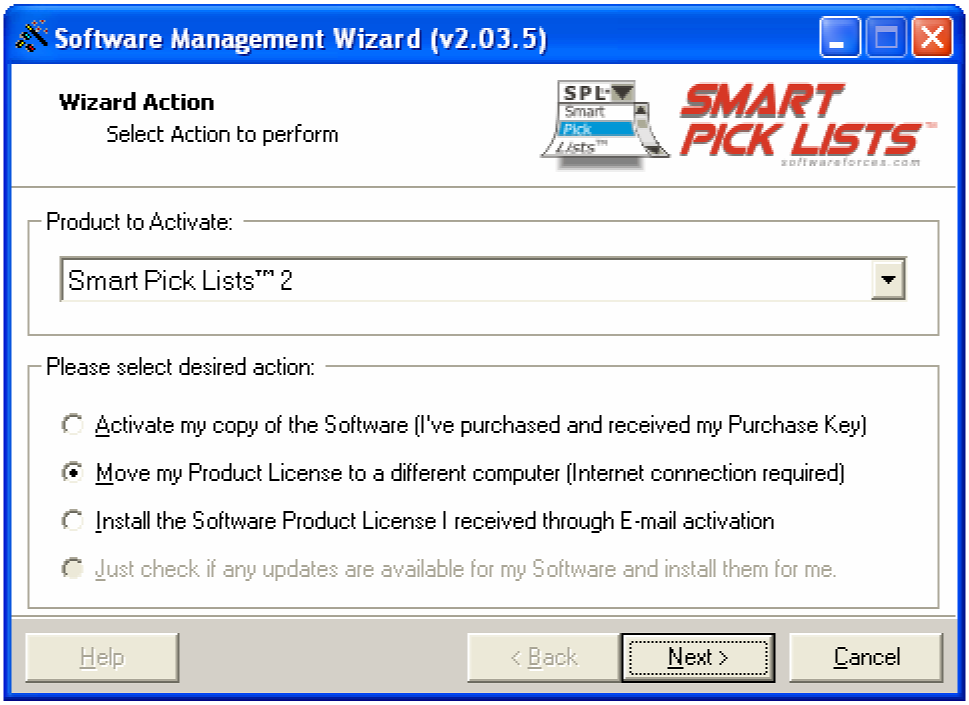
This will then allow you to run the activation wizard and put the license back on this computer by going through the same activation process as you used when you first activated the license.

Another example may be if you are no longer going to use the product on this computer and wish to remove the license from the computer, but will use it on another computer (perhaps a laptop).

This will then allow you to run the activation wizard on the other computer (perhaps a laptop) and put the license on this computer by going through the same activation process as you used when you first activated the license.

In either case, you must first remove the license from the activated computer by going through the "**Move my Product License to a different computer (Internet connection required)**" action in the activation wizard.

This option is only available via an Internet connection.



Software Management Wizard (v2.03.5)

**Wizard Action**
Select Action to perform

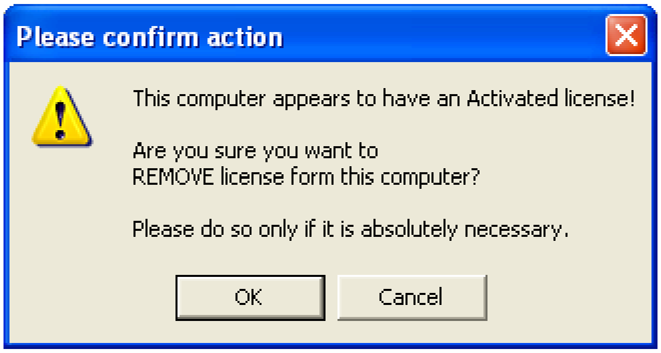
Product to Activate: Smart Pick Lists™ 2

Please select desired action:

○ Activate my copy of the Software (I've purchased and received my Purchase Key)
● Move my Product License to a different computer (Internet connection required)
○ Install the Software Product License I received through E-mail activation
○ Just check if any updates are available for my Software and install them for me.

**Figure 40 - Second screen of the Software Management Wizard – showing a selected product and available actions for the product, with "Move my Product License to a different computer (Internet connection required)" selected**

Once you have selected the product and action to perform is selected to "**Move my Product License to a different computer (Internet connection required)**", click on the Next button.



Please confirm action

This computer appears to have an Activated license!

Are you sure you want to
REMOVE license form this computer?

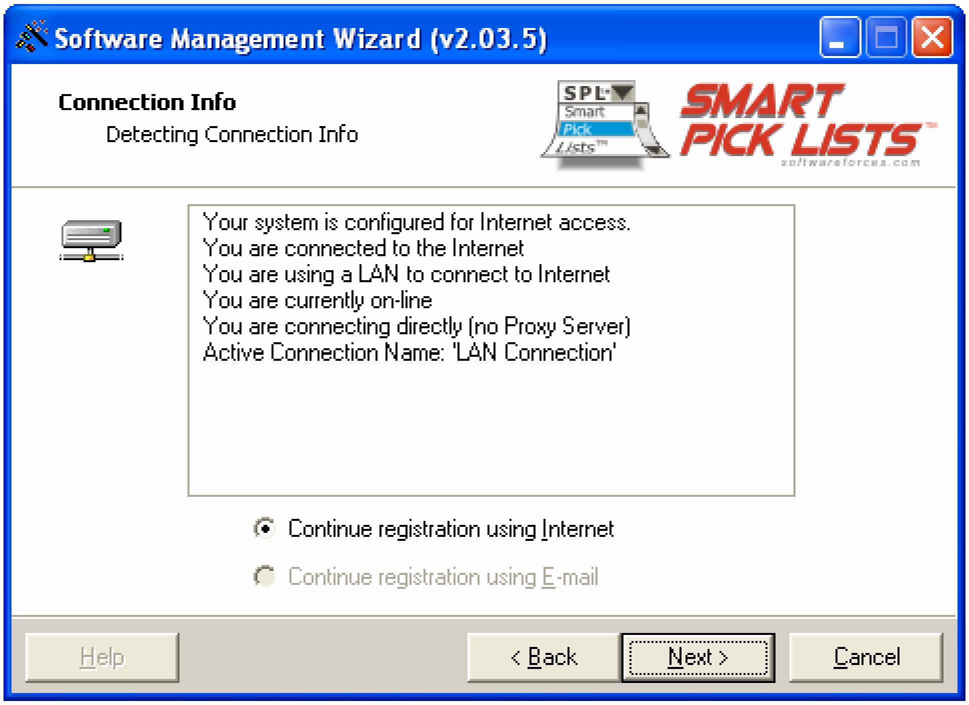
Please do so only if it is absolutely necessary.

**Figure 41 – Warning / Confirmation dialog to REMOVE license**

You will be presented with a warning dialog to confirm that you do wish to remove the license from this computer (as shown in Figure 41).

The next couple of screens are the same as the screens used in the "Activate my copy of the Software (I've purchased and received my Purchase Key)" and are therefore not described in this section.

For a description of those screens, refer to section6.2.1 on page 45 and Figure 17 through Figure 29.



Your system is configured for Internet access.
You are connected to the Internet
You are using a LAN to connect to Internet
You are currently on-line
You are connecting directly (no Proxy Server)
Active Connection Name: 'LAN Connection'

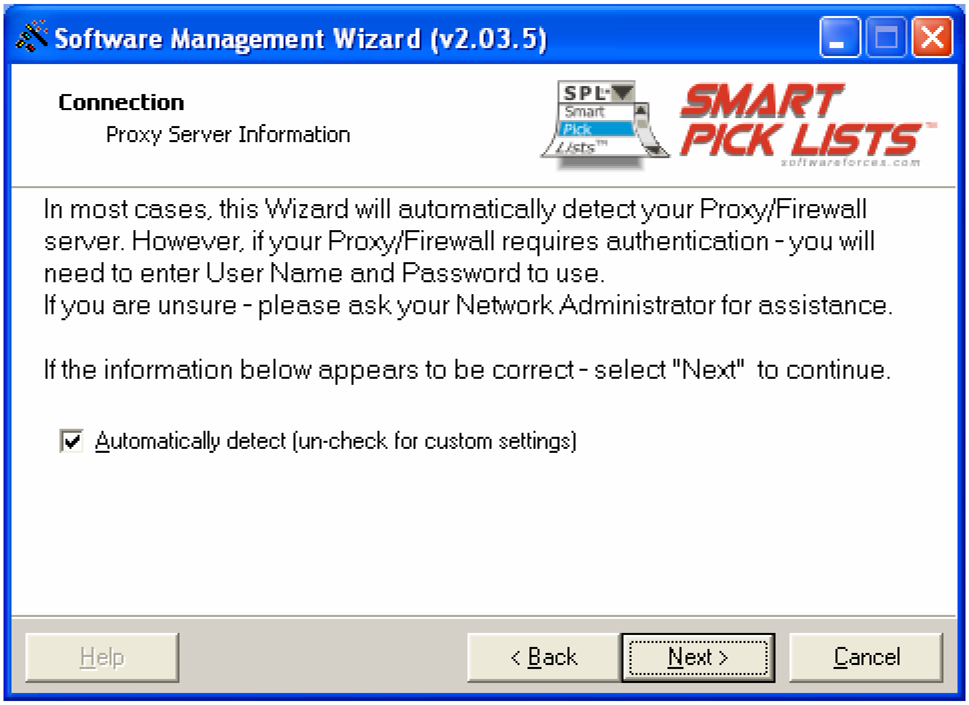
● Continue registration using Internet

○ Continue registration using E-mail

**Figure 42 - Connection Info, with Internet Connection selected**

After providing the notification options (as shown in Figure 21), you are shown the information we were able to retrieve from your connection settings and you are prompted to select the type of method you will be using to provide the activation information to our activation server (as shown in Figure 42).

Moving a product license is only available with an Internet connection, therefore, click on "Continue registration using Internet" and then click on the Next button.



**Connection**

Proxy Server Information

In most cases, this Wizard will automatically detect your Proxy/Firewall server. However, if your Proxy/Firewall requires authentication - you will need to enter User Name and Password to use.
If you are unsure - please ask your Network Administrator for assistance.

If the information below appears to be correct - select "Next" to continue.

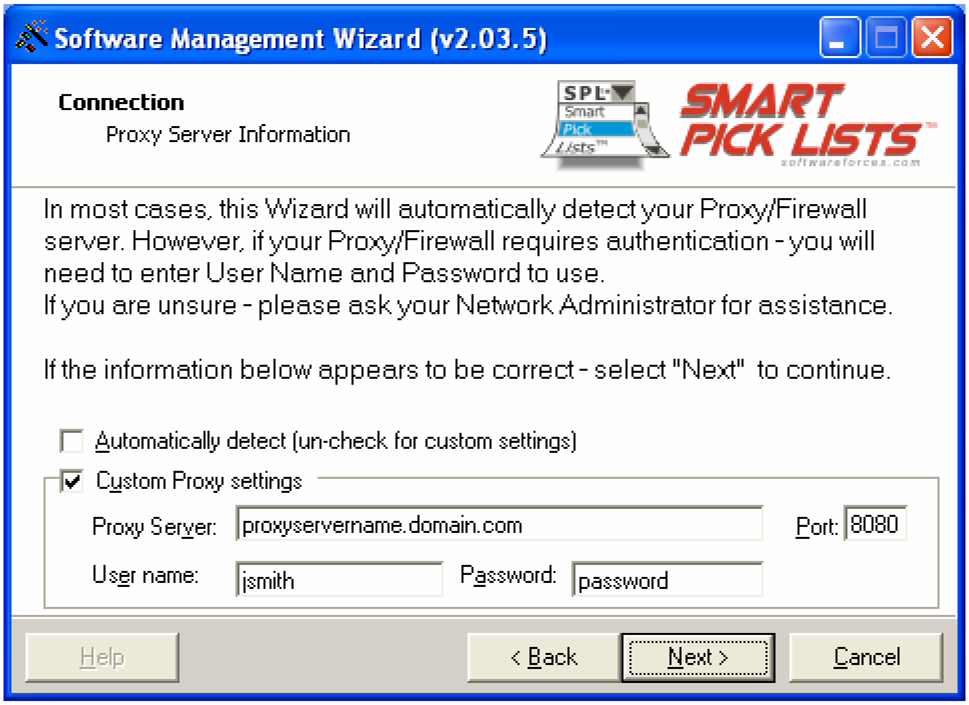
☑ Automatically detect (un-check for custom settings)

**Figure 43 - Connection Proxy Information – with Automatically detect selected**

After selecting to continue with Internet registration (move does not support E-mail), you are prompted with a dialog that warns you about a possible need to authenticate with your firewall/proxy (as shown in Figure 43).

In most cases, it is best to leave “Automatically detect (un-check for custom settings)” selected, and then before clicking on Next, to open a browser window and try connecting to the Software Forces web page (www.softwareforces.com).

If your browser asks you to authenticate, then please do so (our connection will reuse that active connection) and then, come back to the wizard and continue with its steps.



**Connection**

Proxy Server Information

In most cases, this Wizard will automatically detect your Proxy/Firewall server. However, if your Proxy/Firewall requires authentication - you will need to enter User Name and Password to use.
If you are unsure - please ask your Network Administrator for assistance.

If the information below appears to be correct - select "Next" to continue.

□ Automatically detect (un-check for custom settings)

☑ Custom Proxy settings

Proxy Server: proxyservername.domain.com Port: 8080

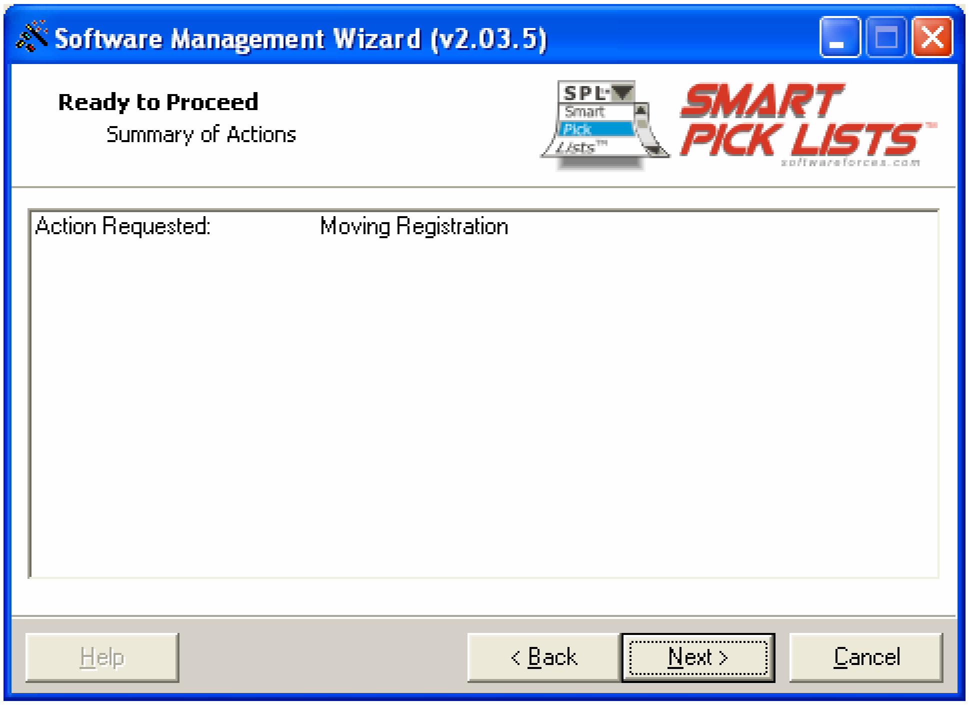
User name: jsmith Password: password

**Figure 44 - Connection Proxy Information – with Automatically detect unselected and Custom Proxy settings revealed**

If the wizard is unable to make a connection and the above described method (setting to “Automatically detect (un-check for custom settings)”) fails, then try turning on “Custom Proxy settings”.

As shown in Figure 44, enter the Proxy Server, Port, User name, and Password (if you are unsure of the appropriate values, contact your network administrator).

If both of these methods fail, and you are still unable to make a connection through the wizard to our activation server over the Internet, contact us for support.



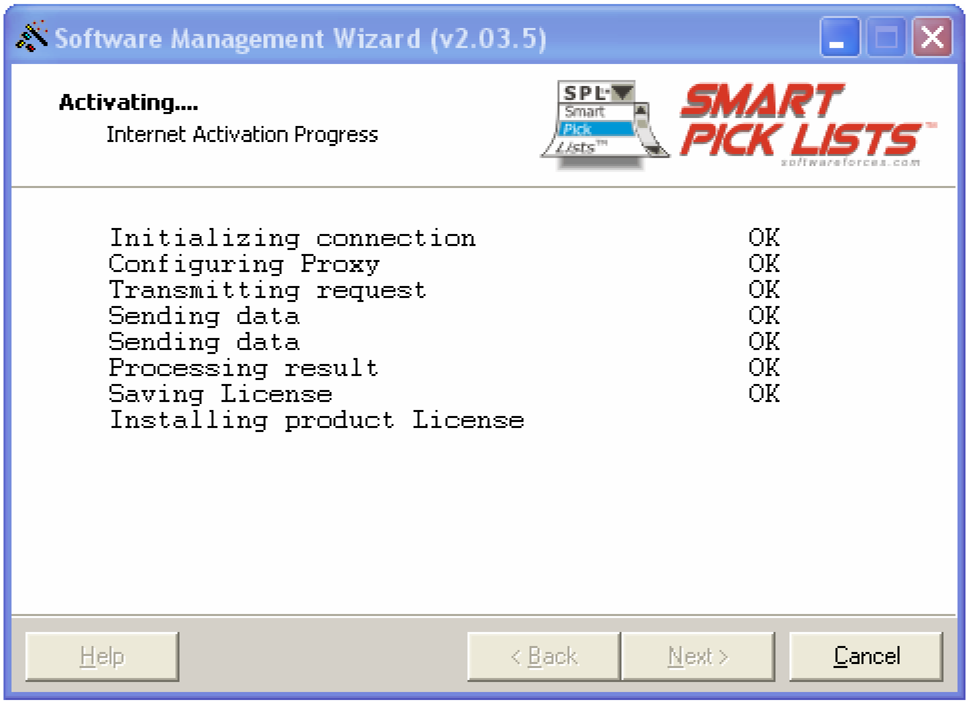
Action Requested: Moving Registration

**Figure 45 – Summary of Actions – displaying Action Requested is Move Registration (remove from this computer)**

Once you continue on the connection selection page, you are presented with a summary page (as shown in Figure 45).

When you are ready to transmit the information to our activation server, click on the Next button to get started.

It is only at this time that we make the actual Internet connection and transmit/receive information.



**Activating....**
Internet Activation Progress

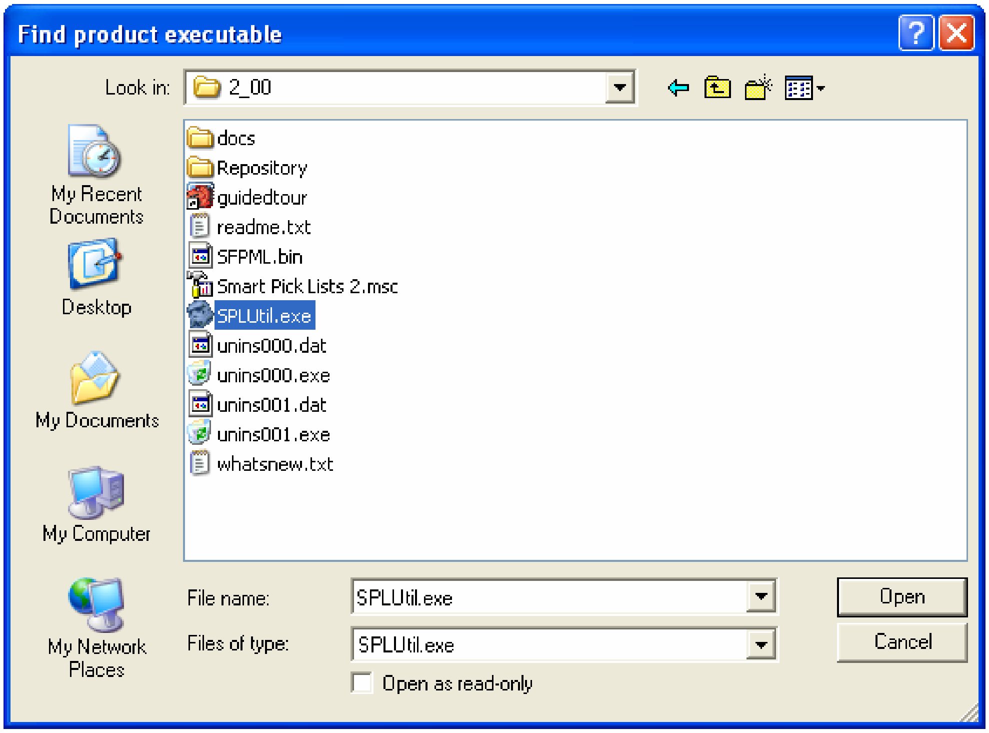
```
Initializing connection            OK
Configuring Proxy                  OK
Transmitting request               OK
Sending data                       OK
Sending data                       OK
Processing result                  OK
Saving License                     OK
Installing product License
```

**Figure 46 - Internet Activation Progress – removing activation – before product executable was found**

If a connection was established, you should see OK for everything.

At the point of “Installing product License”, if the “Software Management Wizard” was unable to locate the product automatically that you are activating (on your computer), a browse dialog will be opened (as shown in figure 41).

There you can navigate to select the executable in the path that it resides, so that the wizard can continue with the de-activation procedure.



Find product executable

Look in: 2_00

File name: SPLUtil.exe

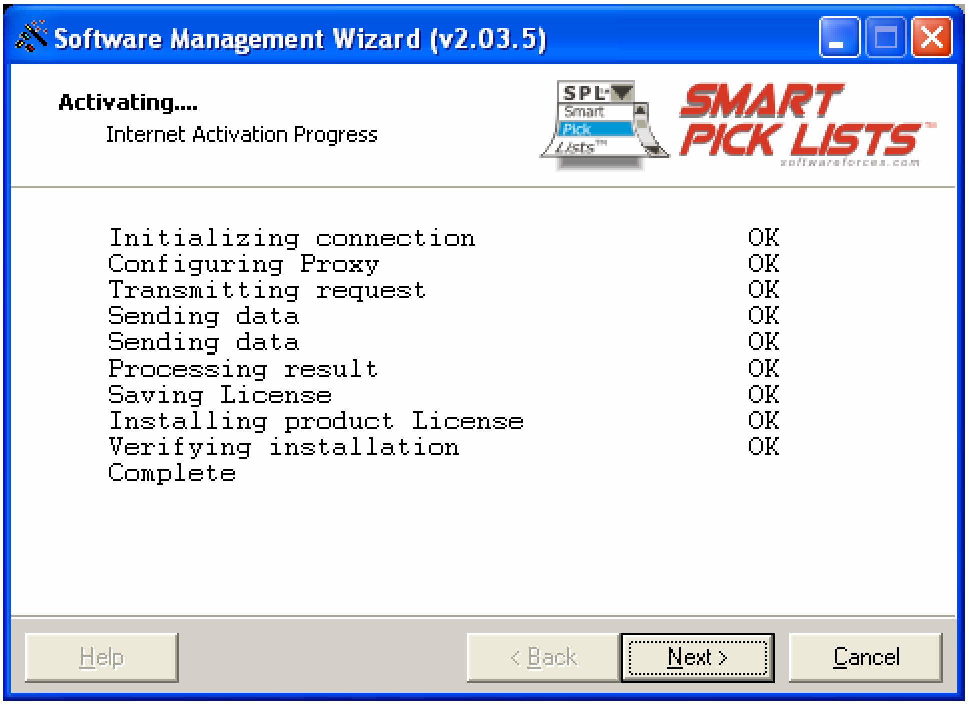
Files of type: SPLUtil.exe

□ Open as read-only

**Figure 47 - Browse dialog opened to search for product executable, if it was not automatically found**

The dialog displayed in Figure 47 is only displayed if the "Software Management Wizard" was unable to locate the product automatically, that you are de-activating.

If this dialog is opened, navigate to the path where the product's executable ("**SPLUtil.exe**") resides, so that the wizard can continue with the de-activation procedure.



**Activating....**

Internet Activation Progress

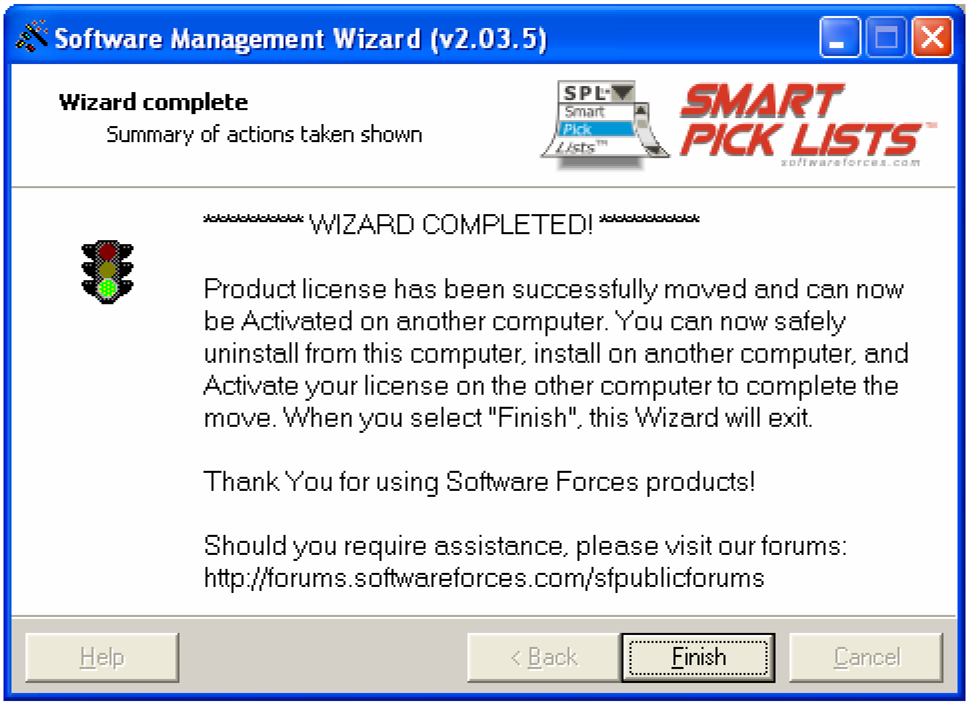
```
Initializing connection           OK
Configuring Proxy                 OK
Transmitting request              OK
Sending data                      OK
Sending data                      OK
Processing result                 OK
Saving License                    OK
Installing product License        OK
Verifying installation            OK
Complete
```

**Figure 48 - Internet Activation Progress – removing activation – after product executable was found**

Once the product was located (either automatically, or via the browse dialog as shown in Figure 47), you should see “Installing product License” and “Verifying installation” turn to “OK”.

Then you will receive a “Complete” (as shown in Figure 48).

At this point you should click on the Next button.



**Wizard complete**

Summary of actions taken shown

*********** WIZARD COMPLETED! ***********

Product license has been successfully moved and can now be Activated on another computer. You can now safely uninstall from this computer, install on another computer, and Activate your license on the other computer to complete the move. When you select "Finish", this Wizard will exit.

Thank You for using Software Forces products!

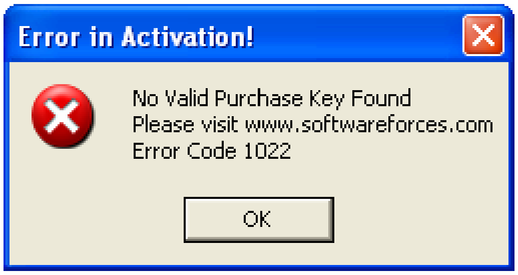
Should you require assistance, please visit our forums:
http://forums.softwareforces.com/sfpublicforums

**Figure 49 – Wizard complete and product activation successfully moved from computer**

If you receive the "WIZARD COMPLETED!" screen (as shown in Figure 49), you have now successfully completed the removal of the license from this computer.

You can now re-run the activation wizard on either this computer again (if you needed to remove the license due to a reformat, etc.) or on another computer and go through the product activation procedure as described in section 6.2.1 on page 45.

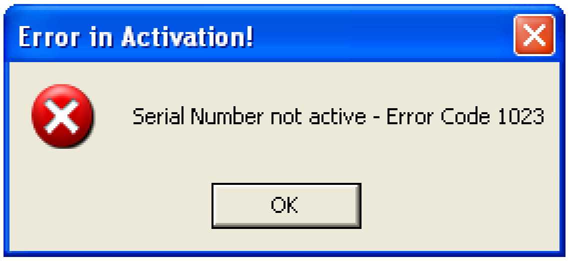
### 6.2.3 Common Errors


Error in Activation!

No Valid Purchase Key Found
Please visit www.softwareforces.com
Error Code 1022

**Figure 50 – Error Code 1022**

The error "No Valid Purchase Key Found" with code 1022 (as shown in Figure 50) is displayed if the purchase key specified could not be validated. Please verify that the key was typed correctly and try again.

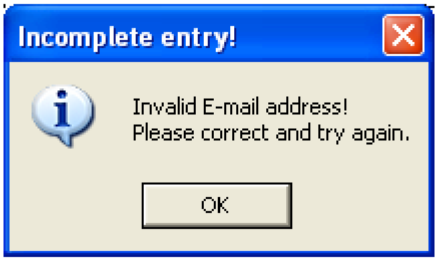

Error in Activation!

Serial Number not active - Error Code 1023

**Figure 51 – Error Code 1023**

The error "Serial Number not active" with code 1023 (as shown in Figure 51) is displayed if the serial number used to activate the product under the purchase, is not the same serial number used to attempt a move.

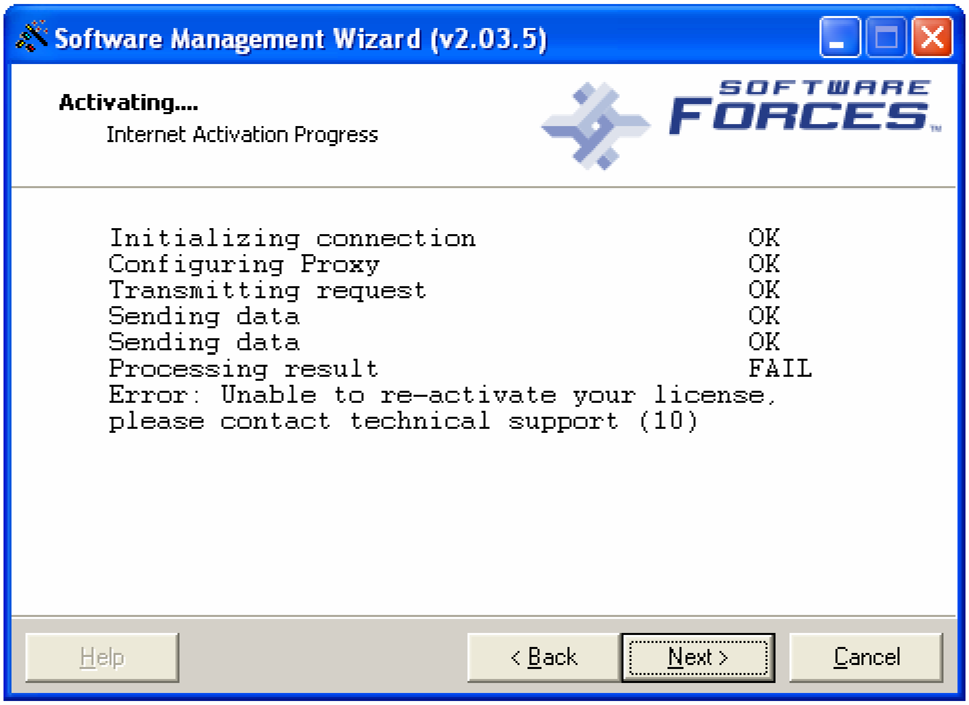
This is usually the case when tampering with the license on your computer has occurred.



Invalid E-mail address!
Please correct and try again.

**Figure 52 – Invalid E-mail address**

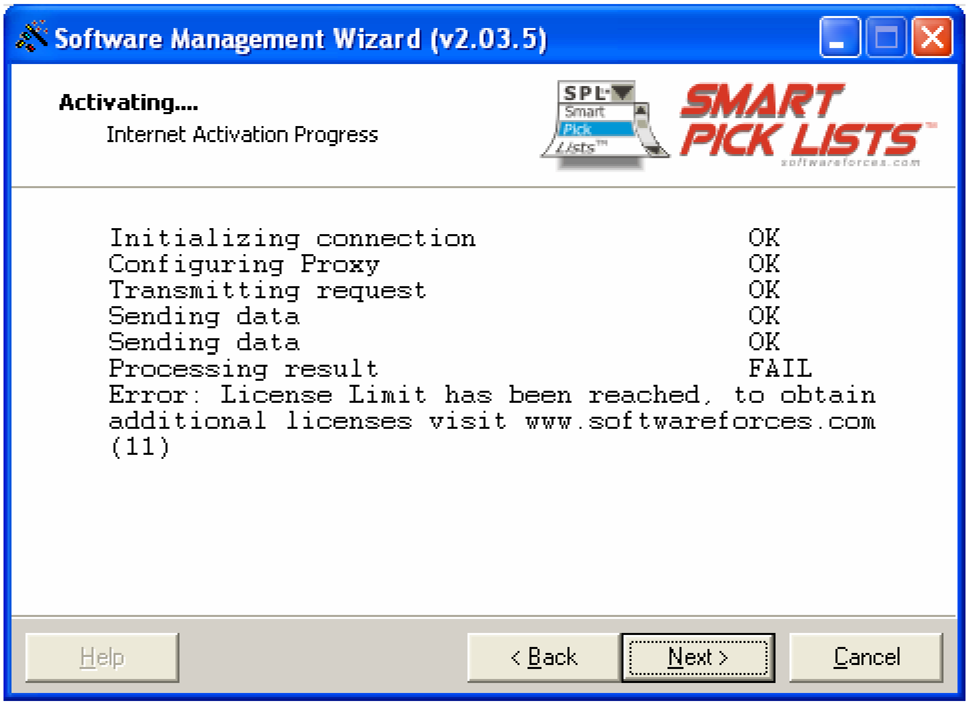
The error “Invalid E-mail address!” is displayed (shown in Figure 52, is shown if the E-mail address specified is either blank or not in the proper format.


Software Management Wizard (v2.03.5)
Activating....
Internet Activation Progress
SOFTWARE FORCES
Help
< Back
Next >
Cancel

```
Initializing connection                 OK
Configuring Proxy                       OK
Transmitting request                    OK
Sending data                            OK
Sending data                            OK
Processing result                       FAIL
Error: Unable to re-activate your license,
please contact technical support (10)
```

**Figure 53 – Error code 10**

The error “Unable to re-activate your license, please contact technical support” with code 10 (as shown in Figure 53) is displayed if the purchase key has been activated too many times on the same computer and the activation server has placed it in to tampering mode.



**Activating....**

Internet Activation Progress

```
Initializing connection                OK
Configuring Proxy                      OK
Transmitting request                   OK
Sending data                           OK
Sending data                           OK
Processing result                      FAIL
Error: License Limit has been reached, to obtain
additional licenses visit www.softwareforces.com
(11)
```

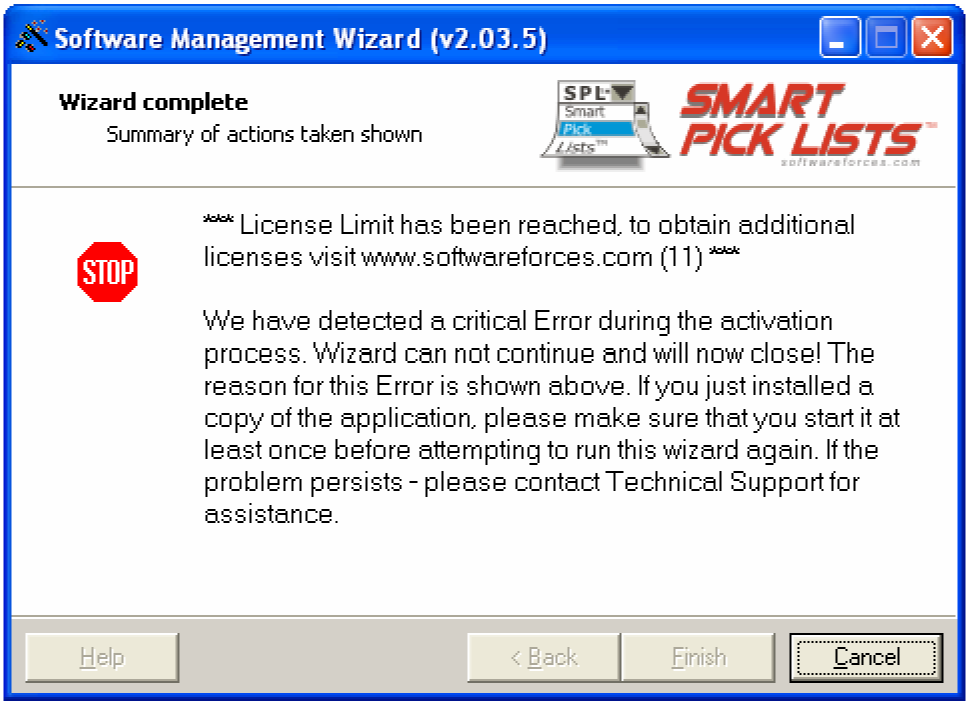
**Figure 54 – Error Code 11**

A purchase key can be licensed for more than one computer. This means that if you purchased 5 licenses, these licenses can be on a single purchase key, instead of having 5 separate purchase keys with one license each.

The error “License Limited has been reached, to obtain additional licenses visit www.softwareforces.com” with code 11 (as shown in Figure 54 and Figure 55) is displayed if the purchase key’s license limit has been reached.

This means that you have already activated the maximum allowed licenses on this purchase key.

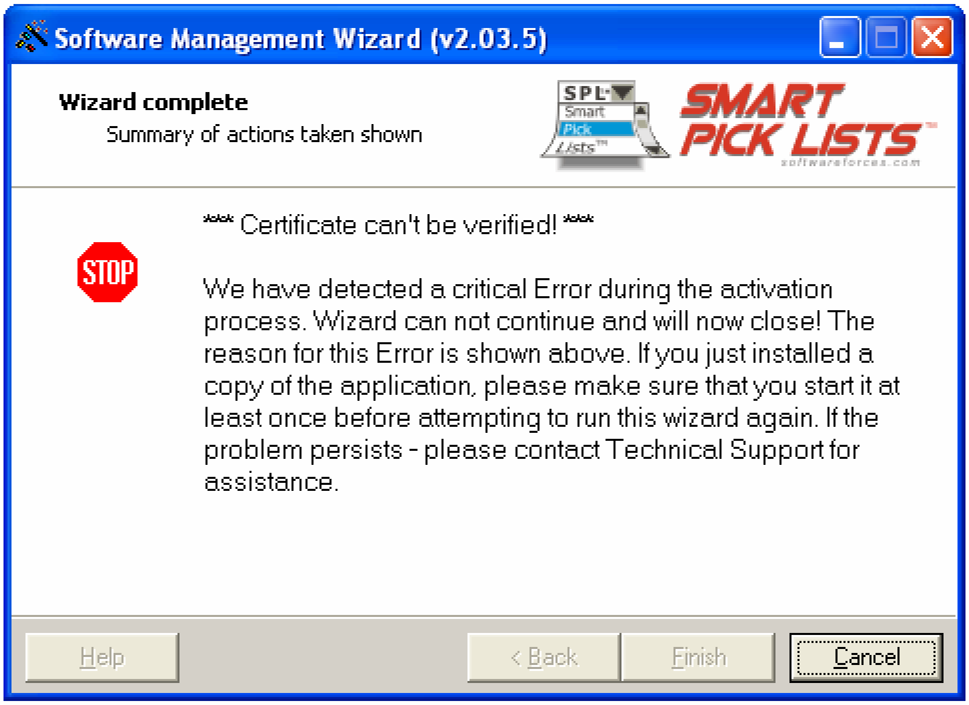
Remember that if all you are trying to do, is to move a license from one computer to another, you must first go through the move license procedure as described in section 6.2.2 on page 68, before you can activate it on another computer.



*** License Limit has been reached, to obtain additional licenses visit www.softwareforces.com (11) ***

We have detected a critical Error during the activation process. Wizard can not continue and will now close! The reason for this Error is shown above. If you just installed a copy of the application, please make sure that you start it at least once before attempting to run this wizard again. If the problem persists - please contact Technical Support for assistance.

**Figure 55 – Error Code 11**



*** Certificate can't be verified! ***

We have detected a critical Error during the activation process. Wizard can not continue and will now close! The reason for this Error is shown above. If you just installed a copy of the application, please make sure that you start it at least once before attempting to run this wizard again. If the problem persists - please contact Technical Support for assistance.

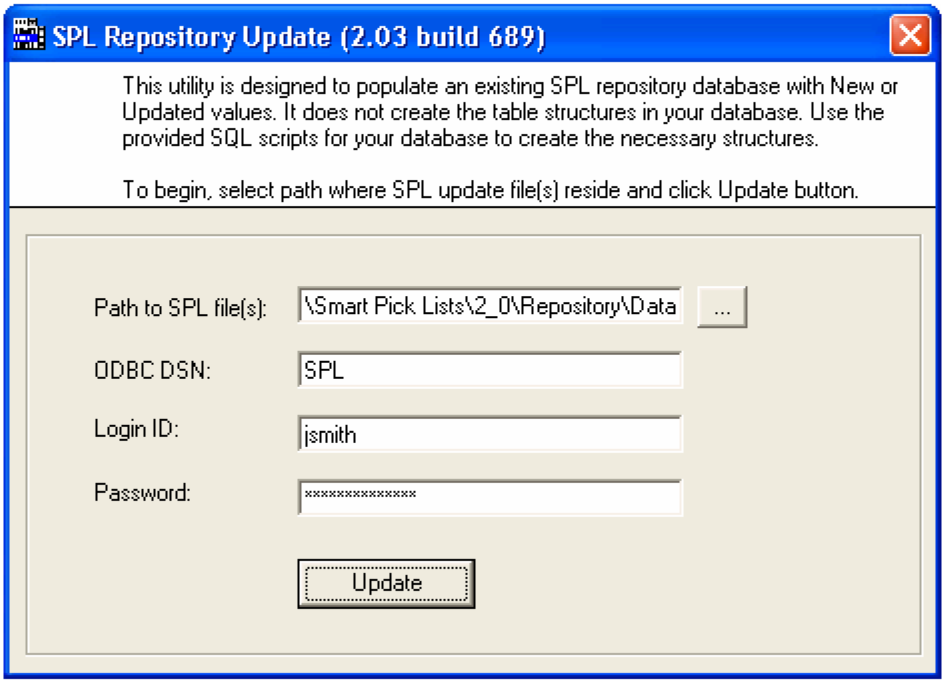
**Figure 56 – Error Certificate can't be verified**

The error "Certificate can't be verified!" (as shown in Figure 56), is displayed if the certificate sent to the activation server from your computer was invalid.

This is usually the case when tampering with the license on your computer has occurred.

# Chapter 7

## 7.0 SPL Repository Update



SPL Repository Update (2.03 build 689)

This utility is designed to populate an existing SPL repository database with New or Updated values. It does not create the table structures in your database. Use the provided SQL scripts for your database to create the necessary structures.

To begin, select path where SPL update file(s) reside and click Update button.

Path to SPL file(s): \Smart Pick Lists\2_0\Repository\Data

ODBC DSN: SPL

Login ID: jsmith

Password: **************

**Figure 57 – SPL Repository Update utility screen shot**

The SPL Repository is a set of tables residing in a database. This repository is used by both the SPL backend and SPL MMC.

SPL backend uses this repository to read necessary values in to its cache (such as the options and custom controls, information about drivers and connections), and uses the combination of the cached information and other contents of the repository to make actual connections to the defined databases so that it can retrieve the dynamic values for defined parameters. The definitions that are stored in the repository are managed by the SPL MMC.

SPL MMC uses this repository to read and write the connections, reports and parameter definitions, options and custom controls. These are all used by SPL backend.

Supported databases for this repository are MSDE, Oracle, and SQL Server. DB2, Informix, and Sybase support are expected in a future version.

### 7.1 Populate repository – on new SPL installation

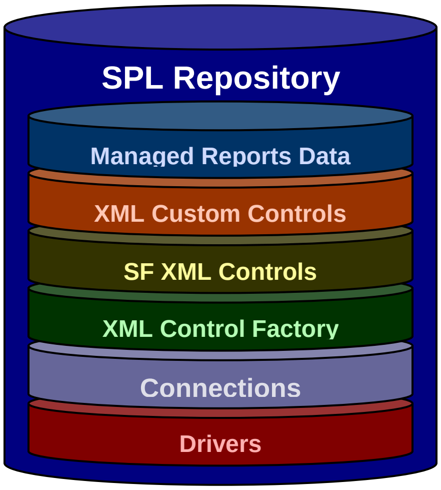
The first time SPL is installed, it is mandatory that you create the SPL table structures in one of the supported repository databases.

For instructions on how and where to install the "**SPL Repository Update**" utility, necessary "**SQL scripts**" (that create the table structures for the supported repository databases), and necessary "**repository data files**" refer to chapter 4, section 4.2.1 on page 19.

Once the table structures have been created, it is necessary to populate them with data.

The data that will be populated consists of:

- driver definitions – these will later be used by the SPL MMC when you define connections to databases
- SF XML Controls
- SF XML Control Factory & miscellaneous SF supporting code



The "**repository data files**" in the "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository\Data**" contain the necessary data to populate the repository with the above data.

If the necessary SQL script was executed on your database to create the table structures, and the ODBC DSN configuration to these tables was defined, then you are now ready to run the "**SPL Repository Update**" (as shown in Figure 57) to populate the tables structures. This utility should be run on the computer where the SPL backend is installed.

The "**SPL Repository Update**" utility requires the following fields to be specified:

- Path to SPL file(s)
- ODBC DSN
- Login ID
- Password

"Path to SPL file(s)" – is the location to the "repository data files" and by default is "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository\Data**". Here you can either type the path or click on the "…" button to search for the files. The data files are called:

- SPLCONN.dat
- SPLDRVR.dat
- SPLRPST.dat
- SPLRPST.1
- SPLRPST.2
- SPLRPST.3
- SPLRPST.4
- SPLSTYLES.dat

"ODBC DSN" – is the same DSN name that you created as per the instructions in chapter 4, section 4.2.1 on page 19.

If the database requires authentication to the "**SPL Repository**" table structures, then you will need to specify the "Login ID" and "Password" that should be used to authenticate with the database. The Login ID provided must have enough rights to both read and write.

After all the necessary fields has been provided, click on the "Update" button to have the "**SPL Repository Update**" utility make a connection to the DSN with the Login ID and Password and populate the table structures with the contents contained in the "**repository data files**".

## 7.2 Populate repository – update existing SPL installation

As we add new features or implement fixes to:

- driver definitions – these will later be used by the SPL MMC when you define connections to databases
- SF XML Controls
- SF XML Control Factory & miscellaneous SF supporting code

We will make those updates available in the form of "**repository data files**" that can be used to update an existing "**SPL Repository**". These repository data files should always be placed in "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository\Data**", and should always overwrite the existing files.

When you are ready to update the repository, run the "**SPL Repository Update**" (as shown in Figure 57) to update the contents of the existing tables structures.

The connections that you have defined, reports that you've added to be managed by SPL, and parameters and/or cascades that you have defined will not altered or deleted.

This utility should be run on the computer where the SPL backend is installed.

The "**SPL Repository Update**" utility requires the following fields to be specified:

- Path to SPL file(s)
- ODBC DSN
- Login ID
- Password

"Path to SPL file(s)" – is the location to the "repository data files" and by default is "**Program Files\Software Forces\Smart Pick Lists\2_0\Repository\Data**". Here you can either type the path or click on the "…" button to search for the files. The data files are called:

- SPLCONN.dat
- SPLDRVR.dat
- SPLRPST.dat
- SPLRPST.1
- SPLRPST.2
- SPLRPST.3
- SPLRPST.4
- SPLSTYLES.dat

"ODBC DSN" – is the same DSN name that you created as per the instructions in chapter 4, section 4.2.1 on page 19.

If the database requires authentication to the "**SPL Repository**" table structures, then you will need to specify the "Login ID" and "Password" that should be used to authenticate with the database. The Login ID provided must have enough rights to both read and write.

After all the necessary fields has been provided, click on the "Update" button to have the "**SPL Repository Update**" utility make a connection to the DSN with the Login ID and Password and populate the table structures with the contents contained in the "**repository data files**".

New with service pack 3 for SPL, the "**SPL Repository Update**" utility will now automatically upgrade the table structures if it finds that the table structures are that of the ones used by service pack 2. It will not however upgrade anything lower than service pack 2. Once "Update" is clicked, it will now first analyze the version and table structures of the SPL Repository and offer a migration option if service pack 2 is detected.

# Chapter 8

## 8.0 SPL MMC user interface

The SPL MMC user interface is a Microsoft Management Console (MMC) snap-in. MMC has shipped with every version of Windows. An example of another MMC snap-in you may already be familiar with is the "Services" snap-in.

MMC snap-ins typically maintain a similar user interface to minimize the learning curve when moving between the different snap-ins.

The SPL MMC is designed to be used by a report designer or appropriate administrator to:

- define default SPL backend behavior as well as the ePortfolio frontend that is integrated with SPL
- define connections to database servers and databases
- select reports from Crystal Enterprise to be managed by SPL
- select web style to use for managed report
- select and configure parameters to receive dynamic values
- define cascade hierarchy

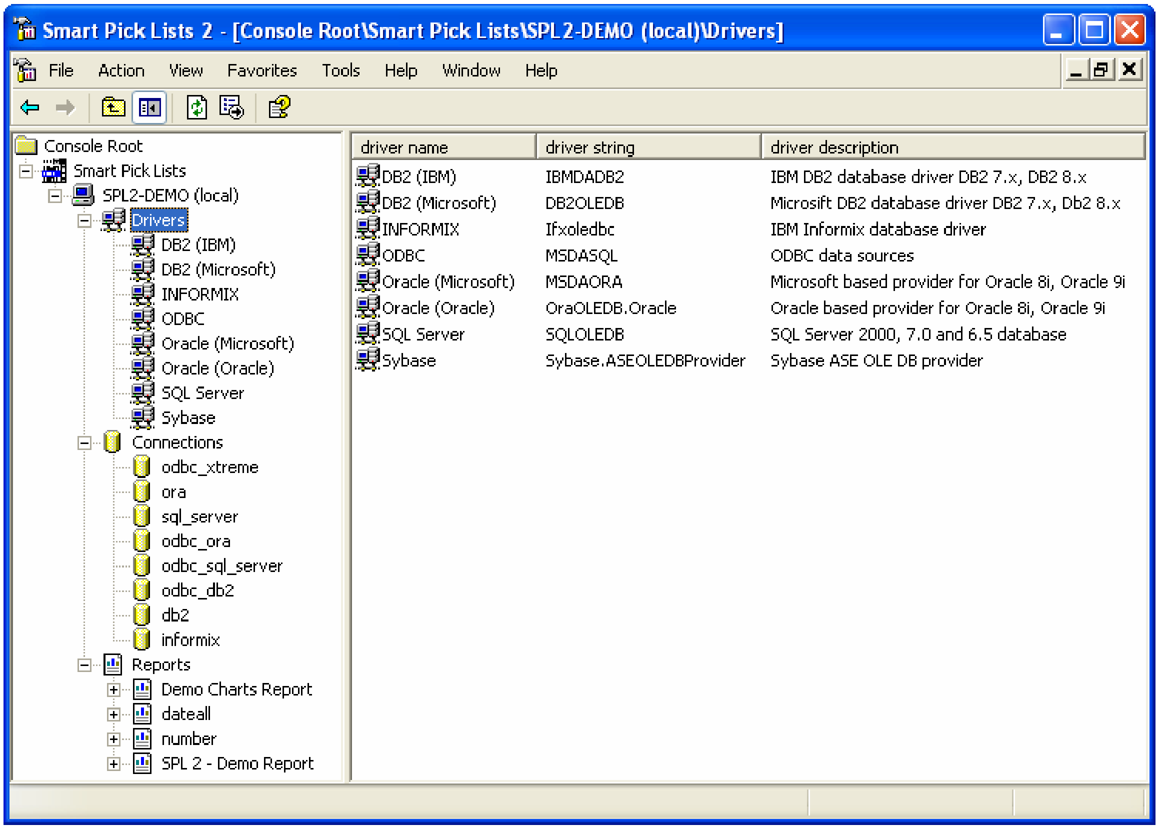
As with other areas designed throughout SPL, when designing the SPL MMC, we had several goals:

- design a frontend product to manage the SPL backend and ePortfolio frontend architecture in an environment that is familiar to most users – the MMC
- provide a single point of configuration and administration for both the SPL backend and ePortfolio frontend
- provide an environment that displays a quick view of configured drivers, connections, reports, and parameters (with or without associated cascades)

### 8.1 Starting the SPL MMC

The SPL MMC can be started using several methods:

- clicking on the short-cut created on desktop
- clicking on the short-cut created in "Start|Programs|Software Forces|Smart Pick Lists|2.0" called "Smart Pick Lists"
- running MMC, clicking on File|Add/Remove Snap-in, clicking on Add button (on the Add/Remove Snap-in dialog) and selecting Smart Pick Lists from the list of available snap-ins (on the Add Standalone Snap-in dialog). Then clicking on the Add button (on the Add Standalone Snap-in dialog) followed by a click on the Close button on the same dialog box, and then clicking on the OK button (on the Add/Remove Snap-in dialog)



| driver name | driver string | driver description |
|---|---|---|
| DB2 (IBM) | IBMDADB2 | IBM DB2 database driver DB2 7.x, DB2 8.x |
| DB2 (Microsoft) | DB2OLEDB | Microsift DB2 database driver DB2 7.x, Db2 8.x |
| INFORMIX | Ifxoledbc | IBM Informix database driver |
| ODBC | MSDASQL | ODBC data sources |
| Oracle (Microsoft) | MSDAORA | Microsoft based provider for Oracle 8i, Oracle 9i |
| Oracle (Oracle) | OraOLEDB.Oracle | Oracle based provider for Oracle 8i, Oracle 9i |
| SQL Server | SQLOLEDB | SQL Server 2000, 7.0 and 6.5 database |
| Sybase | Sybase.ASEOLEDBProvider | Sybase ASE OLE DB provider |

The SPL MMC user interface consists of a “Tools|Options” menu (used for global configuration of SPL), a “Help|About” menu (used to display license information), and two main window panes. The left window pane contains a tree control and the right window pane contains a list control.

The left window pane’s tree control is divided in to 4 areas:

- SPL backend server (the WCS that is being managed by the MMC)*
- Drivers
- Connections
- Reports

* The ability to control multiple SPL backend servers running on multiple WCS servers through a single MMC is currently not supported. Additionally, the ability to run the SPL MMC snap-in via MMC on a computer that does not have the SPL backend installed is currently not supported. Support for these features is planned for a future release.

The right window pane contains a list control that is context sensitive to display values of a selected item in the tree.

To keep with the MMC user interface guidelines, the majority of actions are available by selecting a specific node in the tree and clicking the right mouse button to reveal a pop-up menu displaying available action(s) applicable for the selected node. For example, to view and/or edit values of most nodes, right click on the node and then select Properties from the pop-up menu. This will open a property page dialog that will allow you to view and/or edit values.

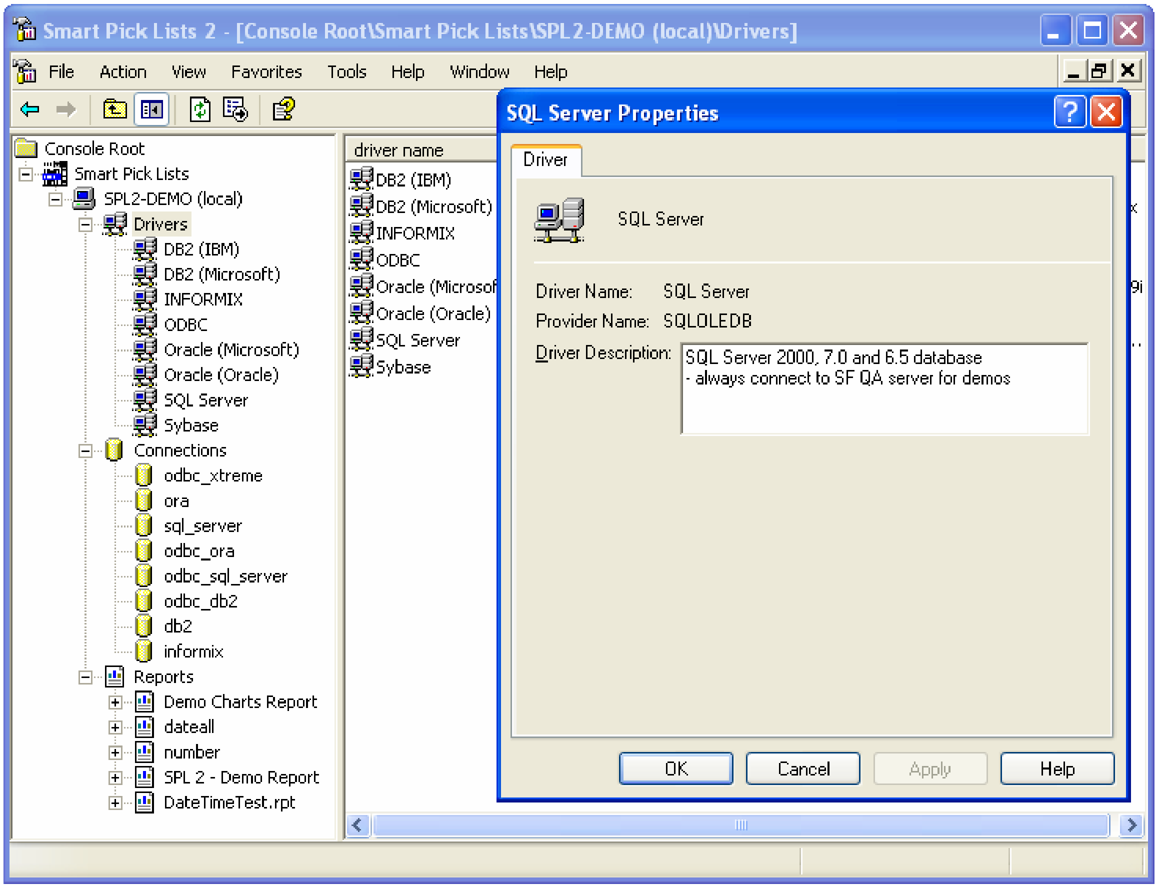
For a detailed explanation on using and navigating around MMC, please refer to the on-line help for MMC.

### 8.2 Drivers

The Drivers node (when expanded) will display a list of available drivers that the SPL backend can use to connect to databases, to retrieve the dynamic values for parameters.

However, the actual connection configuration to the database is specified in the Connections node (such as database server name, database name, and authentication information).

Unlike other areas in the SPL MMC, the Drivers node does not allow you add or remove drivers. However, you can specify a comment or description for a specific driver by editing the specific driver's properties.



Driver Name: SQL Server

Provider Name: SQLOLEDB

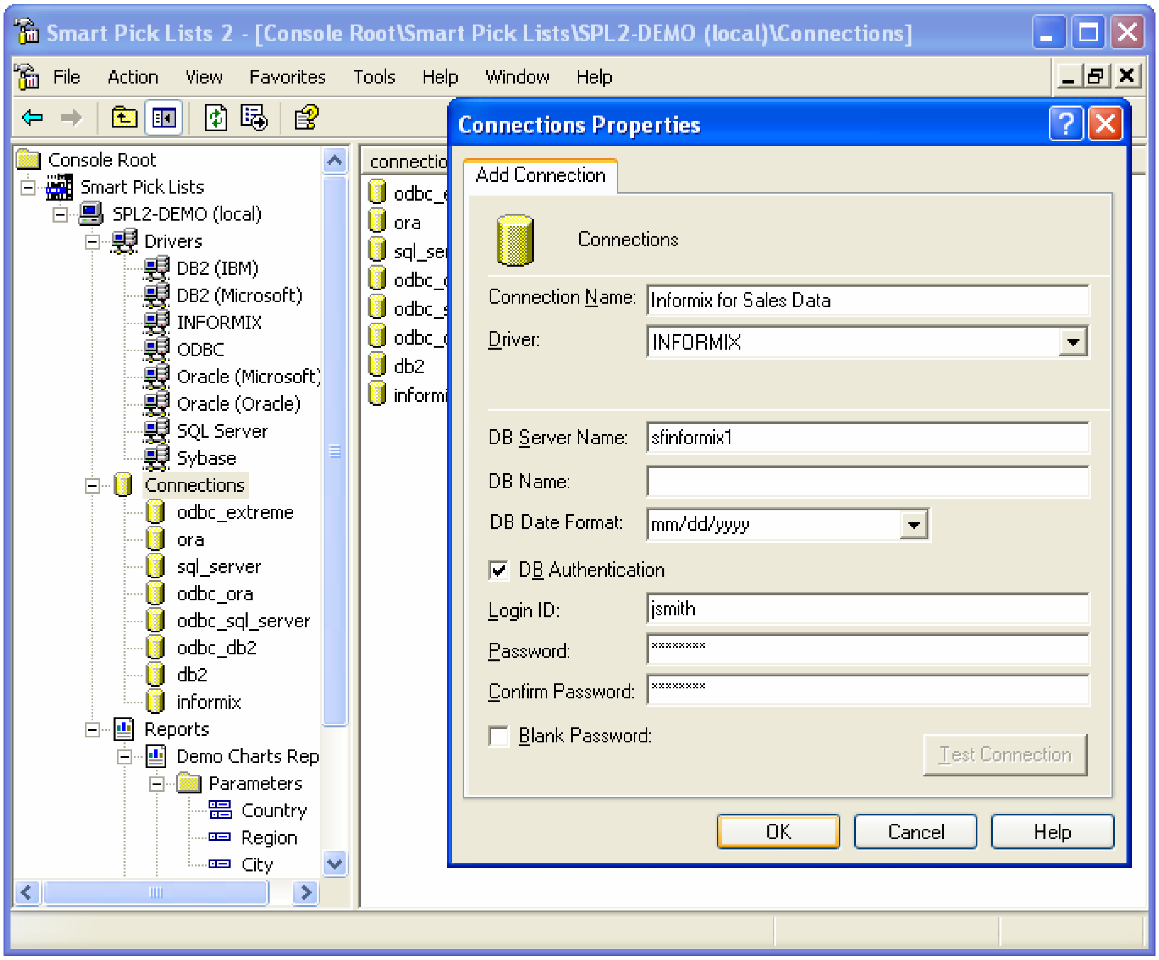
Driver Description: SQL Server 2000, 7.0 and 6.5 database
- always connect to SF QA server for demos

As of this writing, the available drivers that ship with SPL are:

- DB2 (using IBM provider)
- DB2 (using Microsoft provider)
- Informix
- Oracle (using Microsoft provider)
- Oracle (using Oracle provider)
- SQL Server
- Sybase
- ODBC

These drivers will allow you to connect to the majority of databases. Should you require SPL support for a custom driver, please contact our sales department for SPL custom driver pricing.

## 8.3 Connections



Connection Name: Informix for Sales Data

Driver: INFORMIX

DB Server Name: sfinformix1

DB Name:

DB Date Format: mm/dd/yyyy

☑ DB Authentication

Login ID: jsmith

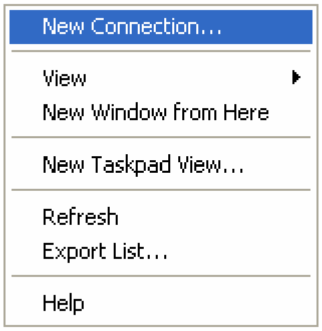
Password: ********

Confirm Password: ********

☐ Blank Password:

The Connections node (when expanded) will display a list of configured connections that the SPL backend will use to connect to a specific database using the configured driver for the connection.

On a new SPL installation, the list of connections will be empty. To define a new connection, highlight the top level Connections node and right click on it to reveal a pop-up menu. From this pop-up menu, select "New Connection…".

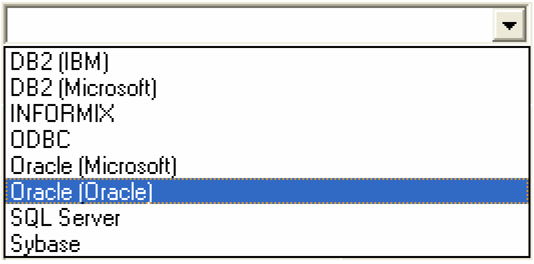


The following options can be configured on the Connection property page:

- Connection Name
- Driver
- ODBC Driver Type (new as of service pack 3 for SPL 2)
- DB Server Name
- DB Name
- DB Date Format (new as of service pack 3 for SPL 2)
- DB Authentication
- Login ID
- Password
- Confirm Password
- Blank Password
- Test Connection

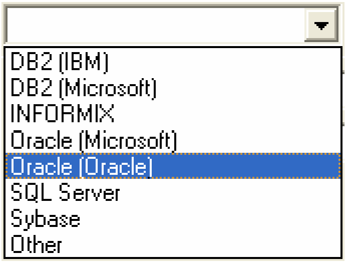
"Connection Name" is a descriptive name that specify for the connection you are creating. This name can be anything you wish.

"Driver" drop-down box displays the available drivers that SPL can use to make this connection. From this drop-down box, select the appropriate driver that SPL should use when connecting to the database.



Some drivers are listed twice, with a provider name displayed in parenthesis next to driver name (i.e. "DB2 (IBM)" and "DB2 (Microsoft)"). The name in parenthesis determines the provider string that SPL will use while making a connection to the database. For example, if you were making a connection to DB2 and have found that using the IBM provider produced better results then using the Microsoft provider, then you'll want to select "DB2 (IBM)" for your driver. Results may also vary depending on the database client installed.

"ODBC Driver Type" is only available if ODBC was selected as the driver, otherwise it is hidden for all other drivers.

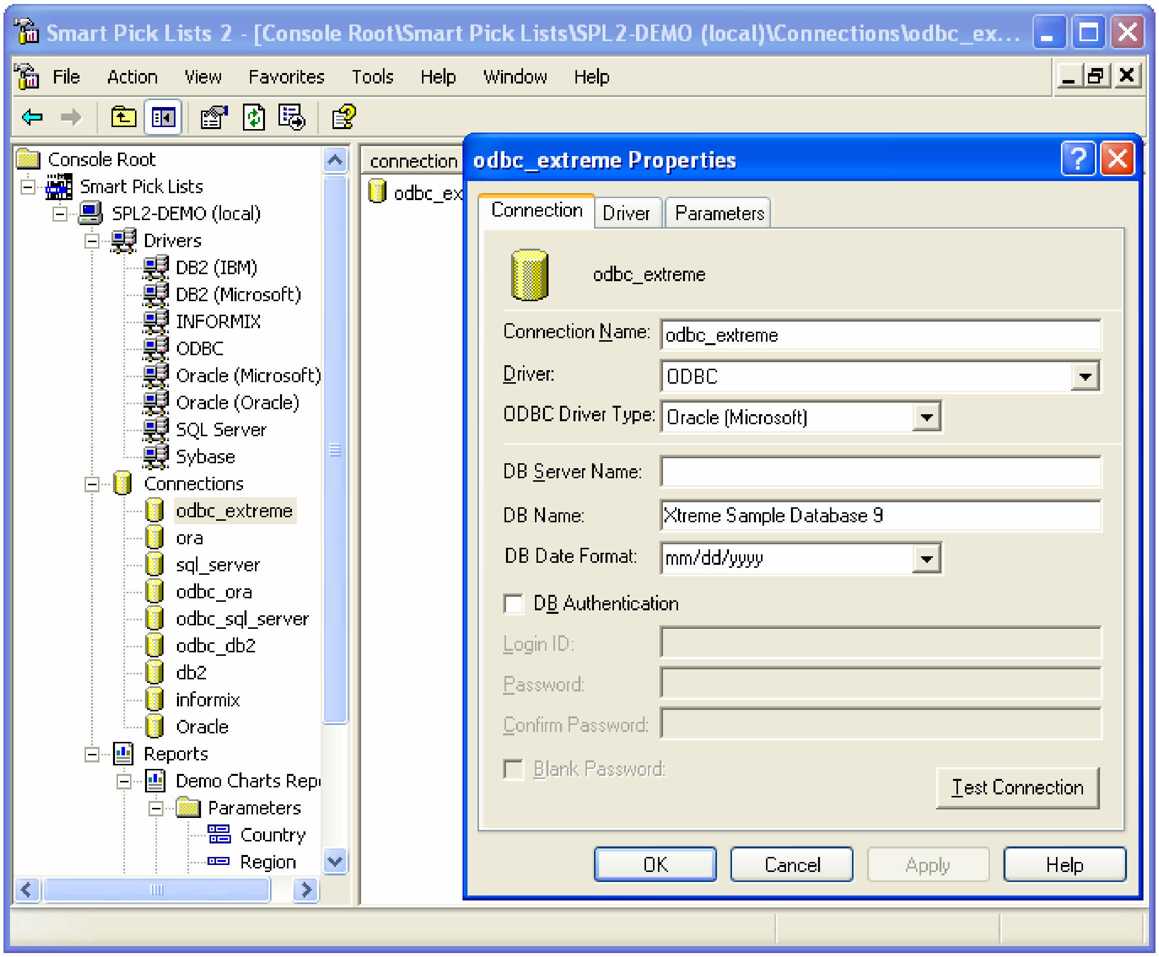


DB2 (IBM)
DB2 (Microsoft)
INFORMIX
Oracle (Microsoft)
Oracle (Oracle)
SQL Server
Sybase
Other

Prior to service pack 3 for SPL 2, any time ODBC was selected as the driver, SPL 2 communicated with ODBC connections very generically.

Unfortunately even with ODBC, the same generic communication method will produce different results based on the database client you have installed. To improve compatibility and performance, we now allow you to specify the ODBC driver type (based on the driver you configured for your ODBC DSN), which will then communicate in an optimized method specific to the database type.

Therefore, if your ODBC connection is one of the listed types, select it. Otherwise, select "Other" and we will communicate using our generic method.



Connection Name: odbc_extreme

Driver: ODBC

ODBC Driver Type: Oracle (Microsoft)

DB Server Name:

DB Name: Xtreme Sample Database 9

DB Date Format: mm/dd/yyyy

□ DB Authentication

Login ID:

Password:

Confirm Password:

□ Blank Password:

"DB Server Name" is the name of the database server or IP address.

"DB Name" is the database name of the database server to use. In a future version of SPL, we plan on supporting the ability to browse a list of available databases on the database server.

**NOTE**:

> For ODBC connections – leave "DB Server Name" blank and for "DB Name" specify the name of the table.
>
> For Oracle connections – leave "DB Server Name" blank and for "DB Name" specify the name of your database server.

"DB Date Format" drop-down box is used to specify the date format used by your database server. This is important for use with parameters that are date ranges and/or parameters that are child cascades which are also dates. By specifying the format here, SPL will both display the date in this format on the web frontend, as well as send the date in the SQL select statement in this format to the database server for use with cascading parameter children. Important to note is that the value displayed as default in the drop-down is not a value detected from your database server – the value is simply the first in the list.

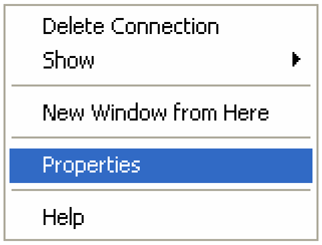
"DB Authentication" is a check box that toggles whether or not this connection requires authentication to the database associated with this connection. If this check box is enabled, then you will need to specify the "Login ID" and "Password" that should be used to authenticate with the database. If a password is required, for confirmation you will also need to specify the identical password in the "Confirm Password" edit box. If no "Password" is required for authentication, then selecting the "Blank Password" check box will disable both "Password" and "Confirm Password" edit boxes and this connection.

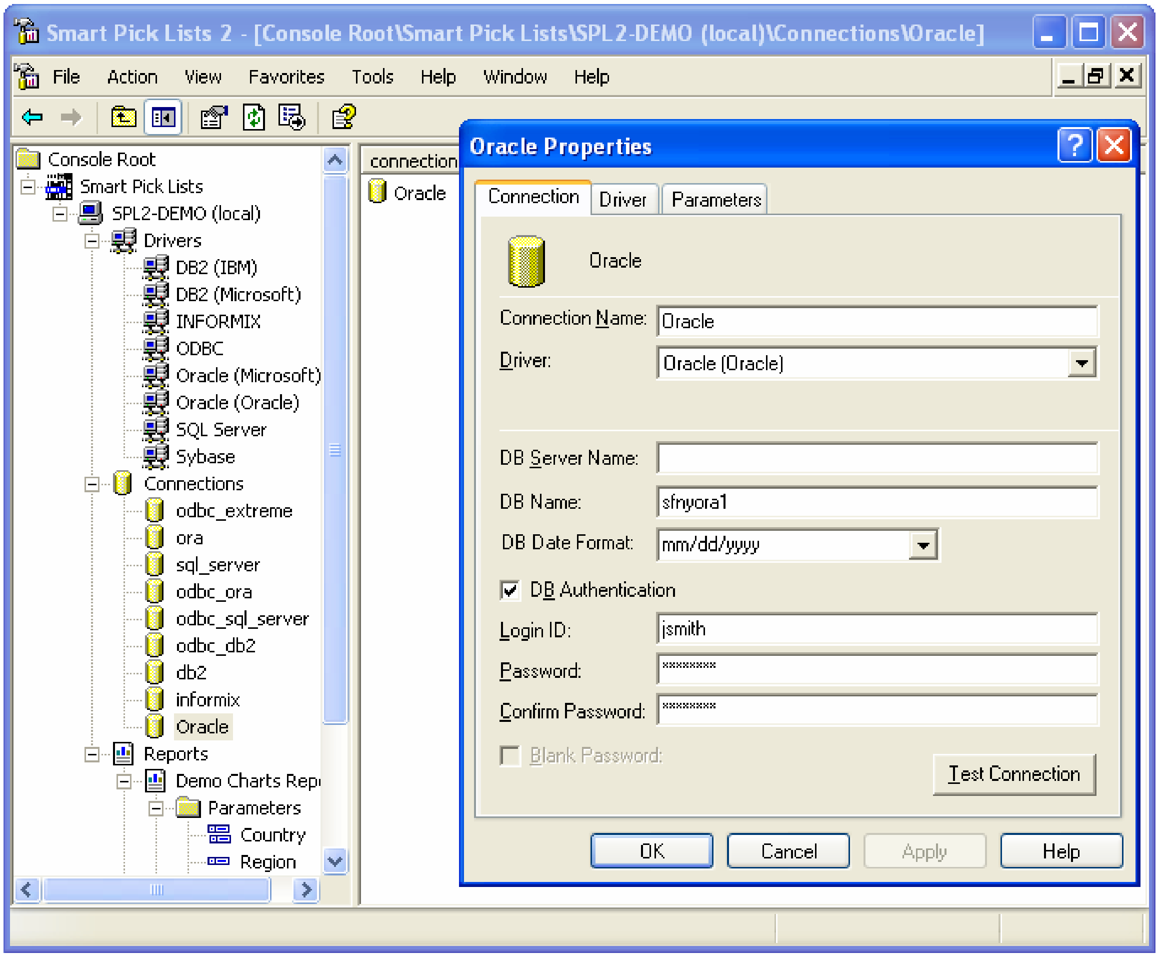
"Test Connection" button will only work after you have applied your settings and will attempt to make a connection to the database, based on the information provided.

A single connection you define can be used with any number of parameters across multiple reports.

After creating and saving the connection, you can open the properties of an existing connection which contains two additional tabs:

- Driver
- Parameters





Connection Name: Oracle

Driver: Oracle (Oracle)

DB Server Name:

DB Name: sfnyora1

DB Date Format: mm/dd/yyyy

☑ DB Authentication

Login ID: jsmith

Password: ********

Confirm Password: ********

☐ Blank Password:

The “Driver” tab displays additional information about the driver (same information that is available on the property page of the driver).

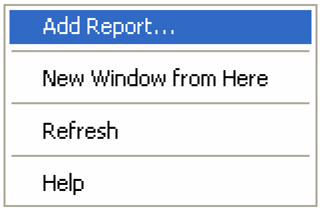
The “Parameters” tab displays a list of all parameters across all reports that have been configured to use this connection.

## 8.4 Reports

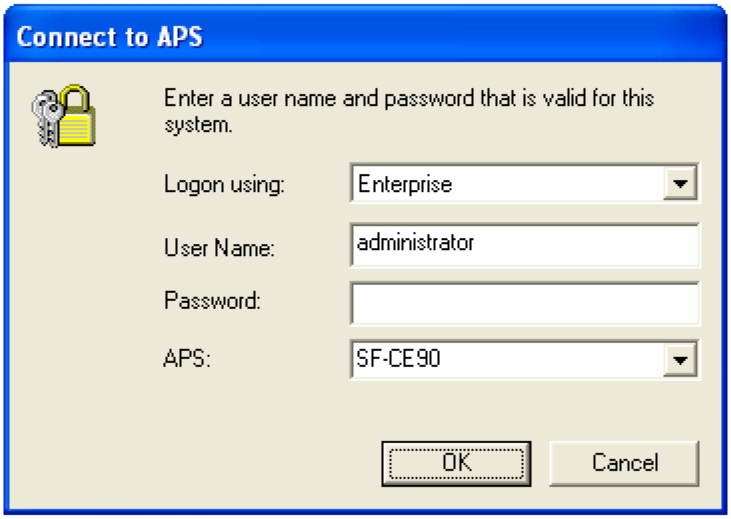
The Reports node (when expanded) will display a list of reports that have been added from Crystal Enterprise to be managed by SPL. When a report is added, its parameters are automatically read and a sub-node is created for each parameter. Only those parameters that you want managed by SPL need to be configured. Any parameters that are not configured will continue to work statically.

On a new SPL installation, the list of reports will be empty.

### 8.4.1 Add Report



To add a new report, highlight the top level Reports node and right click on it to reveal a pop-up menu. From this pop-up menu, select "Add Report…". This will display a Crystal Enterprise APS authentication dialog.



Enter a user name and password that is valid for this system.

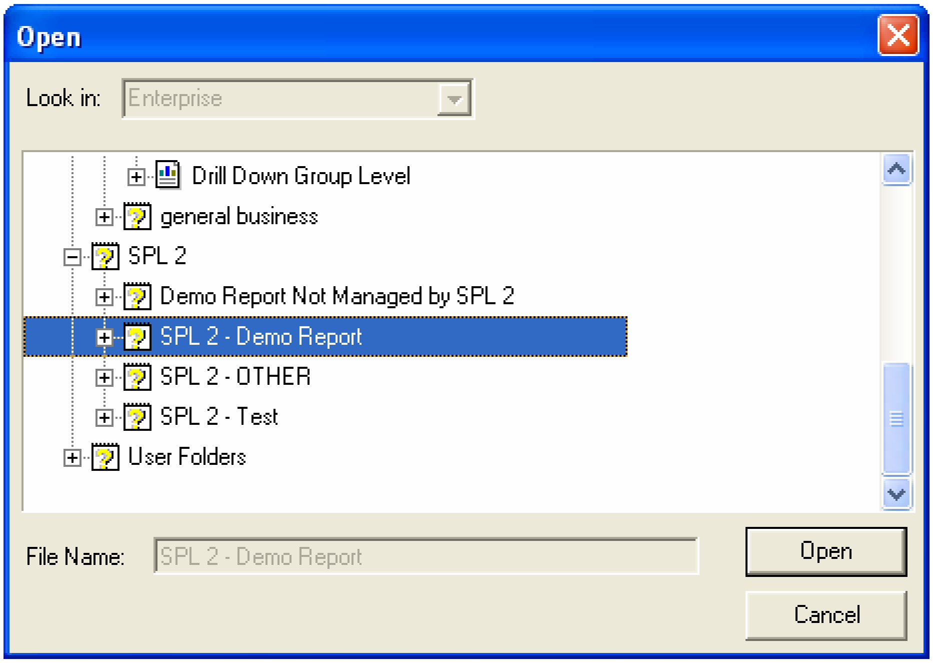
Logon using: Enterprise

User Name: administrator

Password:

APS: SF-CE90

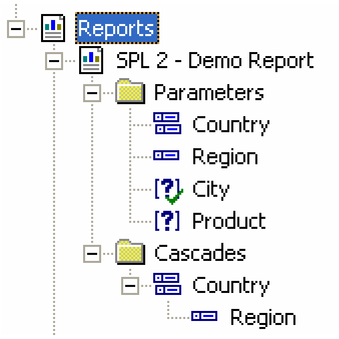
Once you authenticate to the APS, from which you would like to add a report, you will receive a dialog that displays the virtual web folders in Crystal Enterprise.



Navigate to the web folder that contains the report you wish to add, highlight it and click on the "Open" button to have SPL create a sub-node under the Reports node for this report, and create sub-nodes under this report for its parameters.

Note that SPL was designed not to make a single modification to a report that you add to SPL. This allows you complete freedom to continue making updates to the report in Crystal Reports, without having to come back to SPL and reconfigure SPL configurations for the report. The exception of course is if you make certain changes to a SPL configured parameter (such as removing the actual parameter from a report or renaming it), in which case it is a simple matter of highlighting the report in which such a change took place, right click on this report and select "Refresh" from the pop-up menu.

After a report has been added to SPL, expand the newly added report node to reveal the report's parameters.



Icon legend:

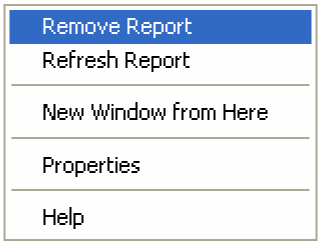
[?] parameter not added to SPL for management (parameter is static)

[?]✓ parameter added to SPL for management (parameter is dynamic), but not part of a cascade

▤ dynamic parameter, and top most parent of a cascade

▭ dynamic parameter, and child of a cascade

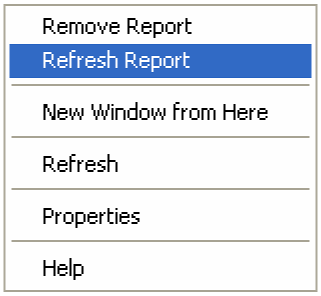
### 8.4.2 Remove Report



Should you no longer wish to have a report managed by SPL, you can remove it from SPL by right clicking on the report which will reveal a pop-up menu. From this pop-up menu select "Remove Report…". This will automatically remove any configured parameter settings and any cascade associations.

### 8.4.3 Refresh Report (new as of service pack 1 for SPL 2)

To refresh an existing report (if you added, removed, modified, or renamed a parameter in the report), highlight the report node of the right you wish to refresh and right click on it to reveal a pop-up menu. From this pop-up menu, select "Refresh Report". This will display a Crystal Enterprise APS authentication dialog.



Once you authenticate to the APS from which you would like to refresh a report, you will receive a dialog that confirming that you wish to refresh.

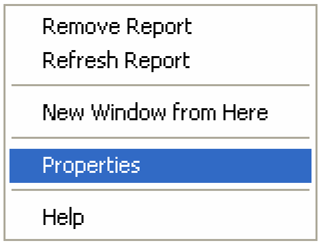
Note that while every effort is made to maintain the configuration you have applied to a parameter, there are cases where some or all, of the configuration for a parameter will be lost:

- If you rename a parameter in the report, on a refresh we will remove the orphaned* parameter. If an orphaned parameter participated in a cascade, the cascade will be discarded and you will have to recreate the cascade.
- If you delete a parameter in the report, on a refresh we will remove the orphaned* parameter. If an orphaned parameter participated in a cascade, the cascade will be discarded and you will have to recreate the cascade.

* A parameter is considered to be orphaned, when it no longer exists in the report but still exists in the SPL repository.

### 8.4.4 Web Styles (new as of service pack 2 for SPL 2)

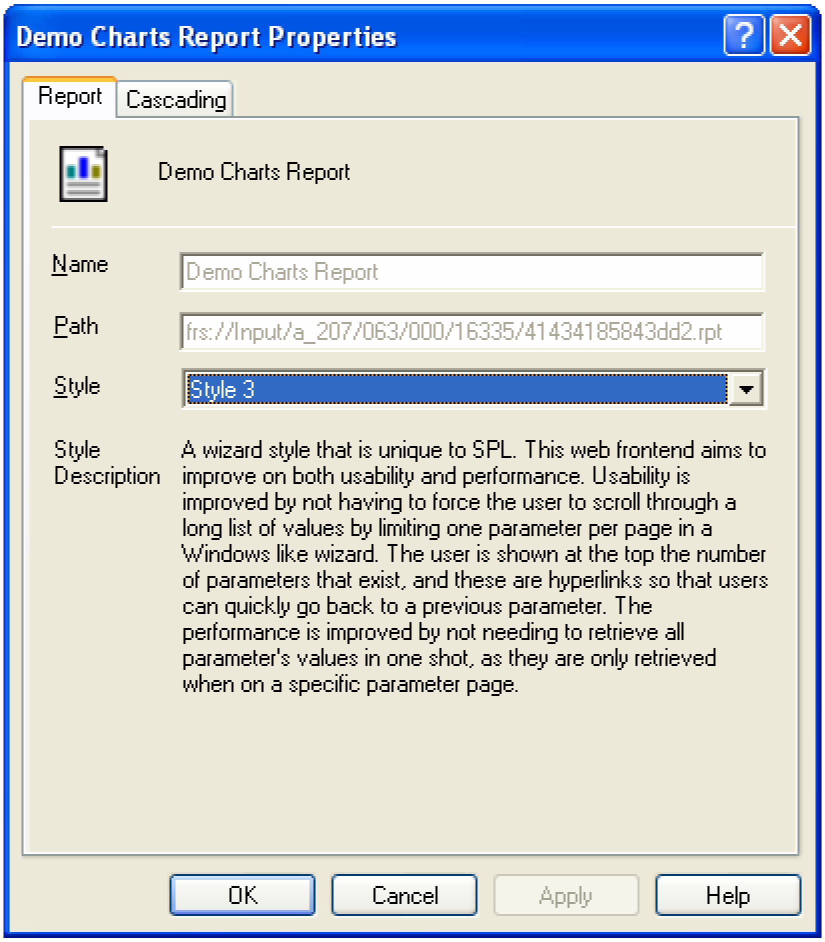
Web styles can be configured per report. This allows for the possibility of each report using a different web style. An example where you may wish to use different web styles for different reports may be based on your departments. For your less experienced users, you may wish to present the parameter selection page for a set of reports designed for those users via a wizard style, while for your more experienced users you may wish to present either the default ePortfolio look and feel style (use style 1 for an identical ePortfolio look and feel) or a variant of it.



To configure a web style for a report managed by SPL, right click on the report to reveal a pop-up menu. From this pop-up menu select “Properties…”.

The following options can be configured on the Parameter property page:

- Style
- Style Description



Name: Demo Charts Report

Path: frs://Input/a_207/063/000/16335/41434185843dd2.rpt

Style: Style 3

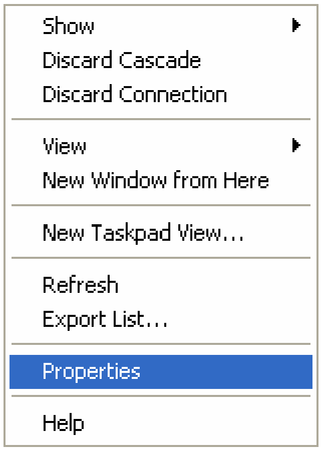
Style Description: A wizard style that is unique to SPL. This web frontend aims to improve on both usability and performance. Usability is improved by not having to force the user to scroll through a long list of values by limiting one parameter per page in a Windows like wizard. The user is shown at the top the number of parameters that exist, and these are hyperlinks so that users can quickly go back to a previous parameter. The performance is improved by not needing to retrieve all parameter's values in one shot, as they are only retrieved when on a specific parameter page.

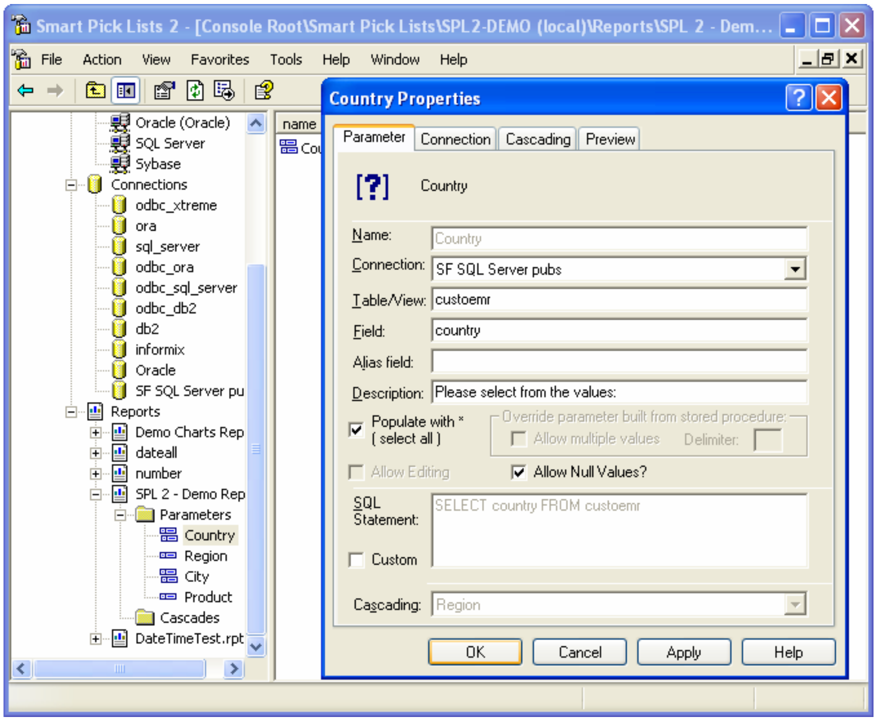
“Style” is a drop-down box which displays a list of available web styles to select from. As of service pack 2 for SPL 2, there are six (6) web styles to choose from. Select the web style you wish to use for this report when displaying its parameter selection screen in ePortfolio.

"Style Description" displays a multi-line description of the selected style.

## 8.5 Dynamic Parameters

To configure a parameter to be managed by SPL and become a dynamic parameter, right click on the parameter to reveal a pop-up menu. From this pop-up menu select "Properties…".





Name: Country

Connection: SF SQL Server pubs

Table/View: custoemr

Field: country

Alias field:

Description: Please select from the values:

☑ Populate with * ( select all )

Override parameter built from stored procedure: ☐ Allow multiple values Delimiter:

☐ Allow Editing ☑ Allow Null Values?

SQL Statement: SELECT country FROM custoemr

☐ Custom

Cascading: Region

The following options can be configured on the Parameter property page:

- Name
- Connection
- Table
- Field
- Alias field
- Description
- Populate with * ( select all )
- Allow Editing
- Allow Null Values
- Allow multiple values
- Delimiter
- SQL Statement
- Cascading

"Name" is the name of the parameter and is not editable here.

"Connection" is a drop-down box which displays a list of previously defined connections. Select the database server / database connection that should be used to retrieve values for this parameter.

"Table" is either a table or view name to retrieve values. This is used in conjunction with the connection you select (which specifies the database server / database to use).

Currently not implemented and expected in a future version is support for stored procedures. Additionally, in a future version of SPL, we plan on supporting the ability to browse a list of available tables, views, and stored procedures available on the database server.

"Field" is the name of the field in the table or view that should be used to retrieve values. Unique to SPL is the ability to use compound fields to return back as a single value for the parameter. Compound fields must reside in the same table and need to be enclosed in square brackets.

For example, specifying **[First Name] + ' ' + [Last Name]** on the field will return a concatenation. If a compound is used, then the alias must be specified. Be sure to use proper SQL syntax here as is required for your database (for example, concatenation syntax may differ from database to database).

"Alias field" is used in conjunction with a "Field" when a compound is used. The name of the field that will hold the new value can be anything. In the above compound field, the Alias field name could be **First_Last_Name**.

"Description" is used to allow you to specify comments or a description that should be associated with this parameter or any of the settings configured for it.

“Populate with * ( select all )” is used to control whether or not one of the values to be selected, will be a *. If this check box is turned on, then when the ePortfolio frontend displays the appropriate control for the parameter in the browser, the first value will be a *. Selecting this * means to select all choices.

“Allow Editing” is used to toggle the display of an edit control in the web frontend. When a parameter is created in Crystal Reports, you can toggle this property on/off on the parameter only if you have at least one default value. Otherwise, with no default values, you can no longer toggle this property off. But since you are now using the report with SPL, there is no longer a need to enter even a single value in the Crystal Reports parameter default value(s). Therefore, to override this behavior, we allow you to toggle it on/off for the purpose of displaying or not displaying an edit control in the web frontend.

“Allow Null Values” check box controls whether Null Values will be returned when retrieving values for a parameter from the database.

“Allow multiple values” (displayed in the “Override parameter built from stored procedure:” group) is available only if the parameter was built from a stored procedure. In Crystal Reports, a parameter built from a stored procedure can not be set to support multiple values. But by turning this option on in SPL, you will direct the web frontend to build a control that supports multiple values. You will also need to specify a delimiter to be used (see below) when building the string that will be sent to the stored procedure. Your stored procedure must be written to support these multiple values by parsing the string for the delimiter.

“Delimiter” (displayed in the “Override parameter built from stored procedure:” group) is available only if the parameter was build from a stored procedure. This is used in conjunction with the above “Allow multiple values”, and defines the delimiter (separator) that will be used when building the string that will be sent to the stored procedure.

“SQL Statement” is automatically and dynamically built and is displayed here as a preview for you to see what SPL will use to retrieve values. With the “Custom” check box turned on, you can change the SQL statement that the SPL backend will use to retrieve values.

“Cascading” is discussed in the “Dynamic Parameters with Cascade” section 8.5.1 on page 103.

On the parameter property page are three additional tabs:

- Connection
- Cascading
- Preview

The “Connection” tab displays additional information about the connection such as the connection name and the driver used with this connection.

The “Cascading” tab displays a tree of all cascades configured for the report.

The “Preview” tab is used to allow you to test the SQL statement (whether automatically created, or the custom one you define). On this tab, you will see the SQL statement that will be used and you can select a limited number of rows to be retrieved from the drop-down. Then clicking on the “Preview” button will make a connection to the database (based on connection you selected to be used with this parameter), and execute the SQL statement. If it is able to retrieve data, that data will be displayed in the listbox. Otherwise if there was an error, the error will be displayed in the listbox.

### 8.5.1 Dynamic Parameters with Cascade

Cascade is the concept similar to a parent and child relationship. Where depending on the value selected in a parent parameter determines the values presented to be selected for a child parameter. So the values you select in the parent parameter work like a constraint for child or dependant parameter’s available values to choose from.

For example, if you had a report with the following parameters (and in the order as below) all configured for dynamic values and cascades:

- Country
- Region
- City
- Employee

You would start with the Country parameter that will list all available countries as the values to be selected. Assuming you select only United States as your choice, you then display in the Region parameter a list of all regions for United States (because Country now acts as a parent to Region and the constraint for Region is the value(s) selected in Country).

Next in the Region parameter (which now lists only regions for United States) you select New York. Since the City parameter receives its constraint from the value(s) selected in Region, the list of available cities will be those of New York. Assuming you select Brooklyn, Manhattan, and Queens.

Now the final parameter which is Employee will display a list of available employees that are based on the constraint of Brooklyn, Manhattan, and Queens. So only employees from those cities will be available as values to be selected.

While SPL allows you to configure and display cascades for any report and any set of parameters, your report does need to be built properly to support this feature. Otherwise, you may need to modify your report(s) for this feature to function properly.

Configuring a parameter to support cascade is almost identical to the procedure for configuring a parameter without a cascade and a single report can have multiple parent cascades.

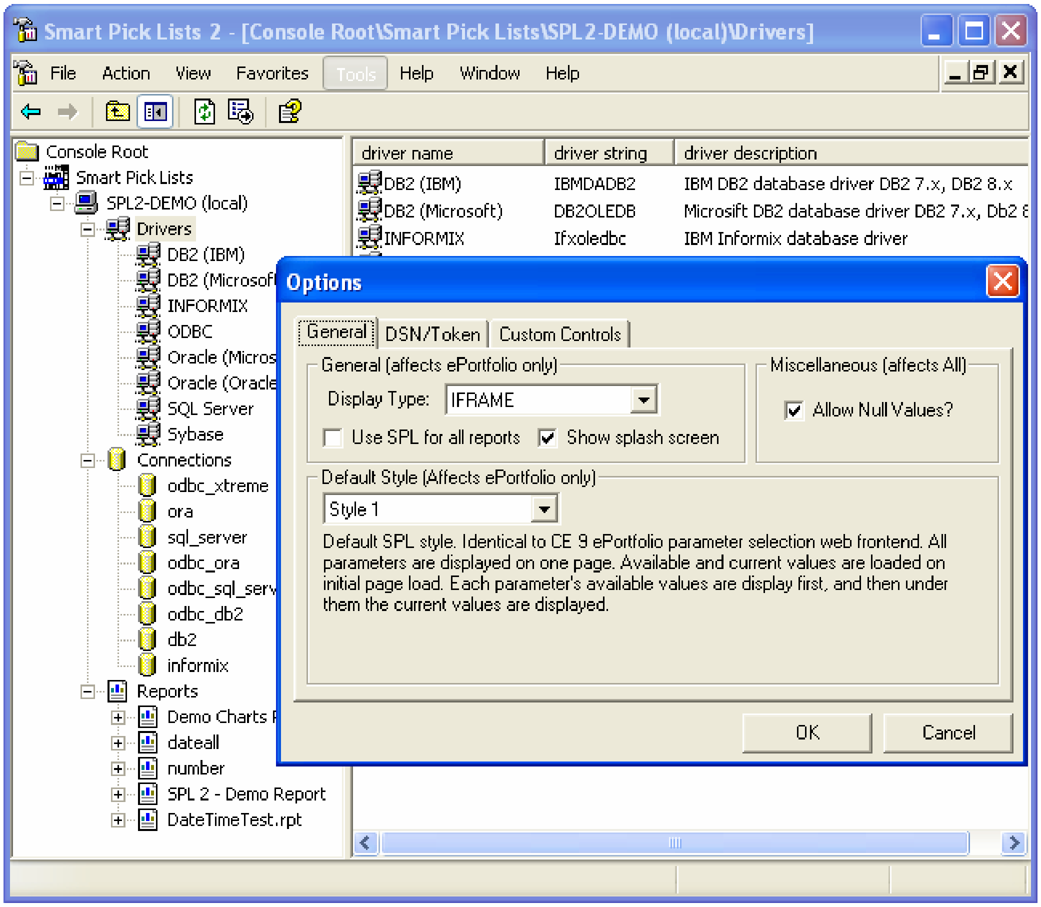
The easiest method to create a cascade is start first by configuring all parameters to be dynamic. Then select the parameter that will used as the parent of the cascade (i.e. Country) and right click on it, which will reveal a pop-up menu. From this pop-up menu select "Properties…". At the bottom of the parameter properties dialog is a drop-down box which specifies the next child parameter to associate in the cascade chain. Once you select the child parameter, you can save the setting and modify the properties of the child parameter to specify the next child in the cascade chain.

There are two caveats to remember when enabling and working with cascade parameters:

- connection of child parameters participating in cascade must be the same as the parent parameter (for cascades, we do not support working with a different database and tables, which may be addressed in a future release)
- custom SQL Statement for child parameters is limited to only additional "where clause" (but leave out the "where" statement itself, SPL will automatically add the "where" statement if you add additional constraints here) statements that can be specified (due to constraints, we must maintain dynamically for all preceding parameters

## 8.6 Options (configuration)

The SPL MMC, SPL backend, and SPL ePortfolio frontend configuration is available from the "Tools" menu, "Options" choice. Making changes to the options here will have a global affect.



| driver name | driver string | driver description |
|---|---|---|
| DB2 (IBM) | IBMDADB2 | IBM DB2 database driver DB2 7.x, DB2 8.x |
| DB2 (Microsoft) | DB2OLEDB | Microsift DB2 database driver DB2 7.x, Db2 8 |
| INFORMIX | Ifxoledbc | IBM Informix database driver |

General (affects ePortfolio only)

Display Type: IFRAME

☐ Use SPL for all reports ☑ Show splash screen

Miscellaneous (affects All)

☑ Allow Null Values?

Default Style (Affects ePortfolio only)

Style 1

Default SPL style. Identical to CE 9 ePortfolio parameter selection web frontend. All parameters are displayed on one page. Available and current values are loaded on initial page load. Each parameter's available values are display first, and then under them the current values are displayed.

When "Tools | Options" is dialog is opened, it contains:

- General tab
- DSN/Token tab
- Custom Controls tab

### 8.6.1 General

The "General" tab contains the following four items:

- Display Type
- Use SPL ePortfolio for all reports
- Show splash screen
- Allow Null Values
- Default Style

"Display Type" is how the SPL backend will render controls that will be used by the SPL ePortfolio frontend. Changing the value for "Display Type", will affect all ePortfolio users. The two available choices are:

- IFRAME
- POP-UP

The default selection is IFRAME because SPL frontend is designed with compatibility in mind. Therefore, for each browser that loads the SPL frontend, there is a detection to confirm whether or not IFRAME is supported. If it is not, it will automatically render the controls just for that browser to be POP-UP based.

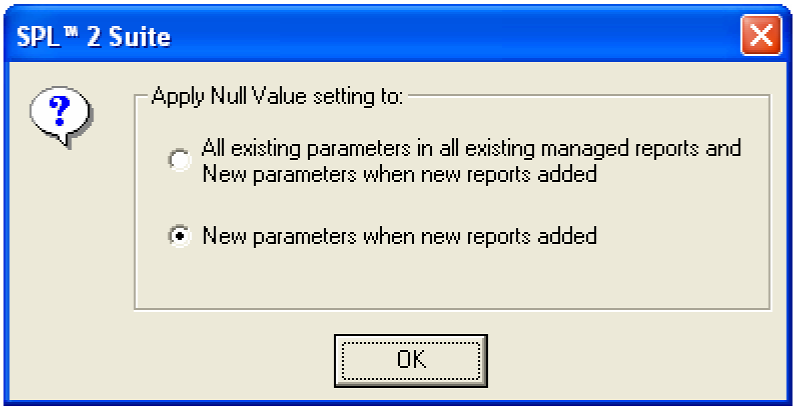
However, if you do not wish to support IFRAME for even those browsers that can, setting "Display Type" to POP-UP will force all browsers to use that instead. For a more detailed explanation on IFRAME and POP-UP, please see chapter 5, section 5.1 on page 25.

"Use SPL ePortfolio for all reports" check box controls how reports that have not been added to SPL to be managed, should be displayed in ePortfolio.

"Show splash screen" check box controls whether to display a intermediate screen while the parameters page is being built. By default this is enabled. Having this enabled is useful if there a many parameters and/or lots of values to be retrieved for parameters -- it may take a little while for the parameters page to be built. While it is being built, the user will see a blank white window during this duration, which typically results in the user not knowing if anything is happening. With the splash screen enabled, the user

If this check box is off, then reports that have not been added to SPL to be managed will render using the standard ePortfolio frontend (without the benefit of IFRAME or POP-UP). However, if this check box is on, then all reports will be displayed using the SPL ePortfolio integrated frontend, and will therefore inherit the SPL controls (including the any custom controls you define).

"Allow Null Values" check box controls whether Null Values will be returned when retrieving values for a parameter. This setting controls the default behavior of parameter properties set on newly added reports. This setting can be overridden per parameter. New as of service pack 3 for SPL 2 – toggling this value displays a dialog box allowing you to specify if your choice affects existing reports and new reports to be added, or will only affect new reports to be added.

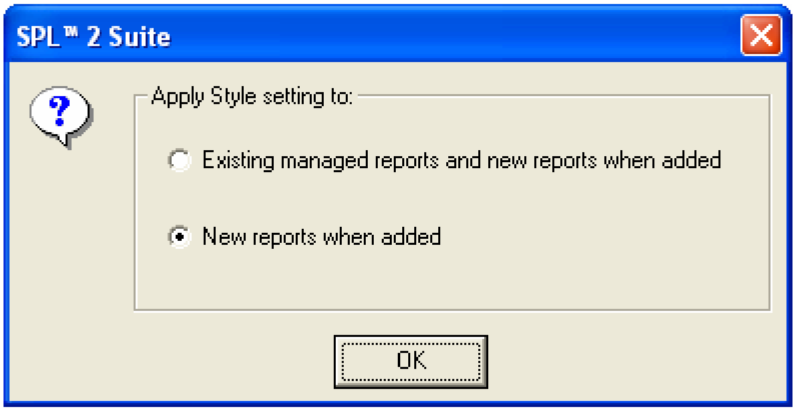


Apply Null Value setting to:

○ All existing parameters in all existing managed reports and New parameters when new reports added

● New parameters when new reports added

“Default Style” drop-down specifies the default web style to use on the ePortfolio integrated web frontend. This setting applies to newly added reports and can be overridden per report (via the report properties). New as of service pack 3 for SPL 2 – toggling this value displays a dialog box allowing you to specify if your choice affects existing reports and new reports to be added, or will only affect new reports to be added.

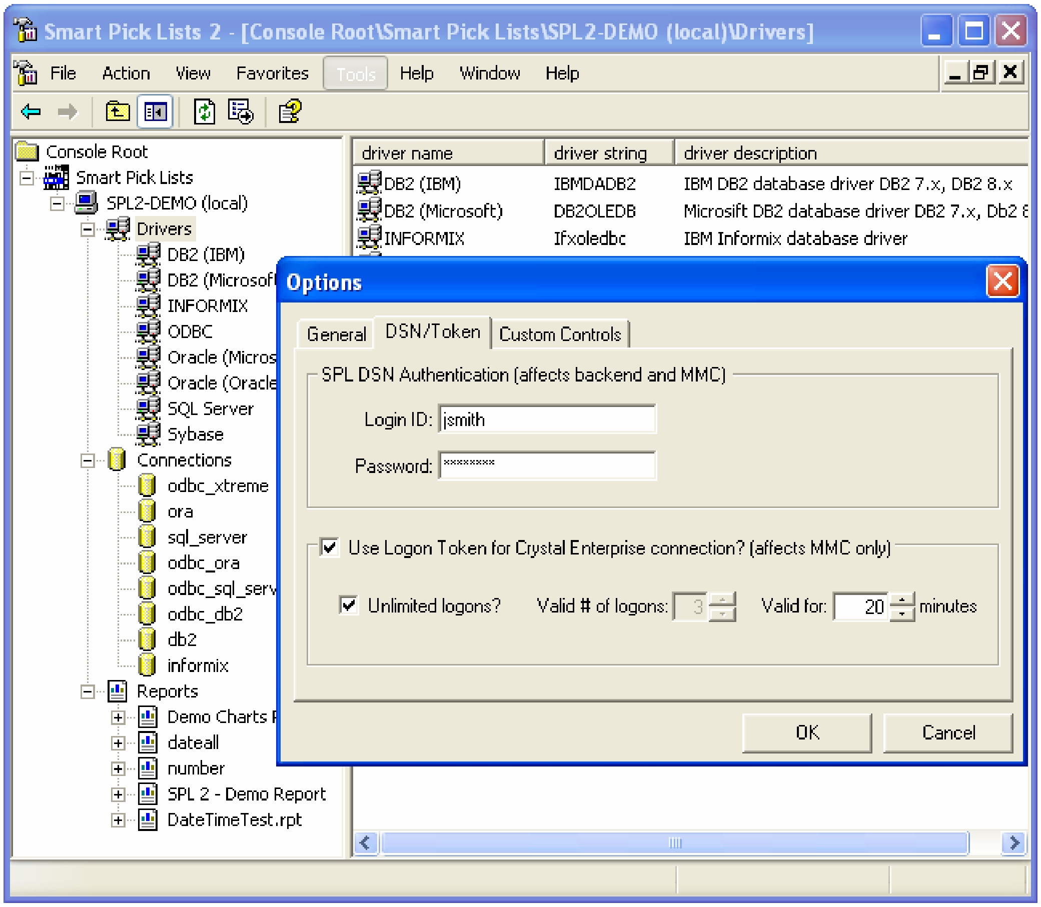


Apply Style setting to:

○ Existing managed reports and new reports when added

● New reports when added

### 8.6.2 DSN/Token



| driver name | driver string | driver description |
|---|---|---|
| DB2 (IBM) | IBMDADB2 | IBM DB2 database driver DB2 7.x, DB2 8.x |
| DB2 (Microsoft) | DB2OLEDB | Microsift DB2 database driver DB2 7.x, Db2 8 |
| INFORMIX | Ifxoledbc | IBM Informix database driver |

SPL DSN Authentication (affects backend and MMC)

Login ID: jsmith

Password: ********

☑ Use Logon Token for Crystal Enterprise connection? (affects MMC only)

☑ Unlimited logons? Valid # of logons: 3 Valid for: 20 minutes

The “DSN/Token” tab contains the following four items:

- Login ID
- Password
- Use Logon Token for Crystal Enterprise connection
- Valid # of logons
- Unlimited logons
- Valid for

The SPL backend and SPL MMC require the SPL repository to function properly. All values defined in the SPL MMC (connections, reports, parameter settings, global settings, etc.) are stored in the repository. The SPL repository can use any SQL compliant database and ships with table creation and drop SQL scripts for MSDE, Oracle, and SQL Server.

For the SPL backend and SPL MMC to communicate with this database, SPL requires a ODBC System DSN to be created called “SPL”.

"Login ID" is the account name that SPL backend should use to authenticate to the SPL ODBC DSN.

"Password" is the account password that SPL backend should use to authenticate to the SPL ODBC DSN.

"Use Logon Token for Crystal Enterprise connection" enables the SPL MMC snap-in to create a Crystal Enterprise Logon Token for the connection that you establish. By enabling this option, and the associated values, you can eliminate being prompted to logon to Crystal Enterprise, each time you add, refresh, or remove a report from SPL management.

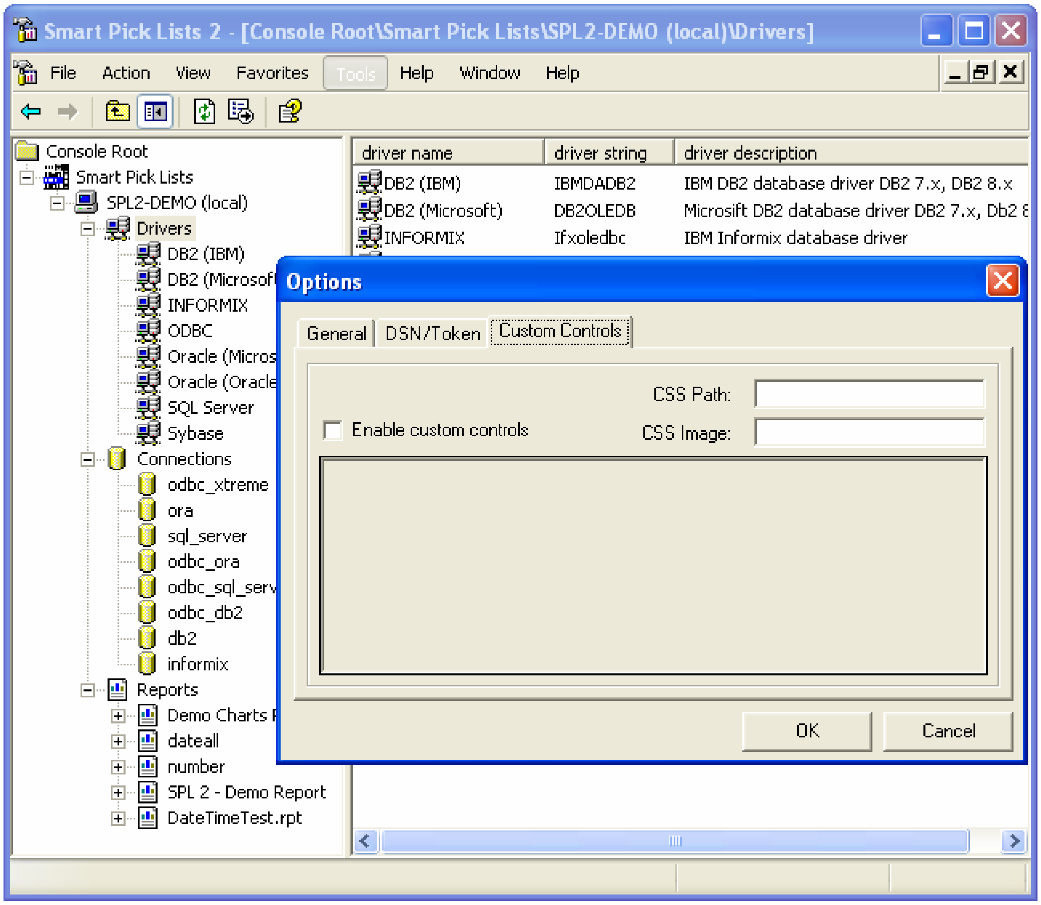
"Valid # of logons" is used to control how many logons the logon token will be valid.

"Unlimited logons" is used to specify that the logon token will  be valid for an unlimited number of logons, but is associated to the "Valid for" (length of time) that the logon token is valid.

"Valid for" is used to specify the length of time that the logon token is valid.

Note that the settings for "Use Logon Token for Crystal Enterprise connect", "Valid # of logons", "Unlimited logons", and "Valid for" affect only the SPL MMC snap-in behavior.

### 8.6.3 Custom Controls



| driver name | driver string | driver description |
|---|---|---|
| DB2 (IBM) | IBMDADB2 | IBM DB2 database driver DB2 7.x, DB2 8.x |
| DB2 (Microsoft) | DB2OLEDB | Micrsift DB2 database driver DB2 7.x, Db2 8 |
| INFORMIX | Ifxoledbc | IBM Informix database driver |

The "Custom Controls" tab is available should you wish to define or override the default SF Controls that are used by ePortfolio integrated SPL frontend using the Custom Controls XML API. This tab contains three items:

- Enable custom controls
- CSS Path
- CSS Image
- Paste Custom Controls XML here

"Enable custom controls" check box enables the other related custom control items on the dialog, and informs SPL backend that while rendering the controls for ePortfolio frontend, it should incorporate the custom controls.

"CSS Path" allows you to specify the location on the Web Component (WC) server where to find a CSS style sheet if your custom frontend and/or custom controls require a different style sheet then the one used by the standard ePortfolio. Leaving this field blank will use the default ePortfolio style sheet.

“CSS Image” allows you to specify the location on the Web Component (WC) server where to find image files if your custom frontend and/or custom controls use them. Leaving this field blank will use the default ePortfolio image path.

“Paste Custom Controls XML here” text box is where you paste the XML you’ve defined using the Custom Controls XML API. Since this text box is a free form text box, it is strongly recommended that you use your favorite XML design tool and then paste the contents in this text box.

# Chapter 9

## 9.0 SPL backend Smart API

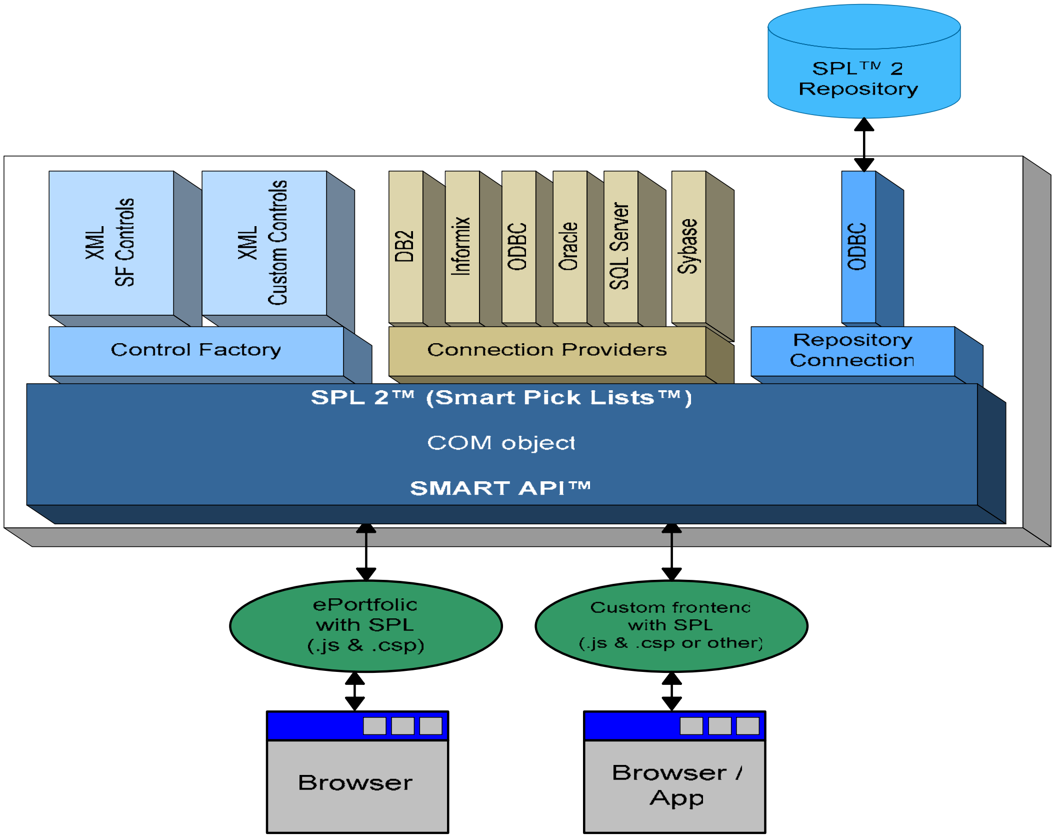
The heart, soul, and brain of SPL is driven by the SPL backend COM object's Smart API™.

This API is used by the SPL MMC and SPL ePortfolio integrated frontend (as well as any custom frontend you design).

The API objects provide complete encapsulation of the entire logic that includes:

- Connecting to the SPL repository database to obtain the parameter properties
- Invoking appropriate logic to build the sets of controls for the specific parameter types
- Managing the database connections
- Parsing XML configuration files and the XML Controls
- Managing all interactions between JavaScript objects, SPL MMC and SPL repository objects

The SPL backend COM Smart API™ is documented in a separate help file called "SPL2.chm" . For details on the COM Smart API, please refer to that help file.

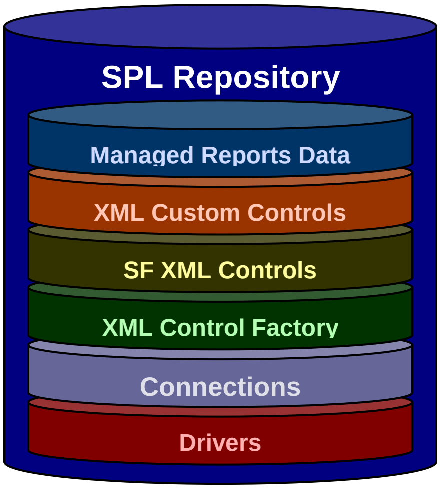


# Chapter 10

## 10.0 XML Custom Controls API

The controls that are rendered in a frontend by default are Software Forces (SF) XML Controls that we have defined based on the XML Control API and are populated in to the “SPL Repository”.

While it is intentional that you can not modify the SF XML Controls, you can build your own XML Custom Controls, and even override the SF XML Controls.



Using your favorite XML editor and the XML Custom Controls API (as described in the table below), you can define your own custom control(s).

For example, if you wanted to change the default SF XML Control used for an “Input” box to be that of a “Checkbox”, you could define the control below (specifying all relevant values for the attributes you wanted this control to use):

```
<control tag="INPUT" endtag="false" type="CHECKBOX" name="" value=""
checked="false" disabled="false" readonly="false" size="" maxlength="" src=""
id="" class="" lang="" dir="" title="" style="" align="" alt="" usemap=""
ismap="" tabindex="" accesskey="" onfocus="" onblur="" onselect=""
onchange="" onclick="" ondblclick="" onmousedown="" onmouseup=""
onmouseover="" onmousemove="" onmouseout="" onkeypress=""
onkeydown="" onkeyup="" accept=""/>
```

By setting the “<paramcontrols ptype>” of your custom control to the same name of the “<paramcontrols ptype>” used by the SF control, you will override the default SF XML Control defined. Setting the “<paramcontrols ptype>”  to any other name, means that the control will be treated as an additional control.

| PType | Control Group Type Description |
|---|---|
| dropdowndiscrete_edit | 1 textfield with 1 single listbox |
| dropdowndiscrete | 1 single listbox |
| dropdownrange_edit | 1 textfield with 1 range drop-down listbox |
| dropdownrange | 1 range drop-down listbox |
| selectboxdiscrete_edit | 1 textfield with 1 single listbox |
| selectboxdiscrete | 1 single listbox |
| selectboxrange_edit | 1 textfield with 1 range listbox |
| selectboxrange | 1 range listbox |
| selectboxboth_edit | 2 textfields and 2 listboxes with checkboxes |
| selectboxboth | 2 listboxes with checkboxes |
| dropdowndiscrete@number_edit | 1 textfield with listbox |
| dropdowndiscrete@number | 2 listboxes with 2 textfields and checkboxes |
| dropdownrange@number_edit | 2 listboxes with 1 textfields and checkboxes |
| dropdownrange@number | 2 listboxes and checkboxes |
| selectboxdiscrete@number_edit | 1 textfield and 1 listbox |
| selectboxdiscrete@number | 1 listbox |
| selectboxrange@number_edit | 2 listboxes and 2 textfields + checkboxes group |
| selectboxrange@number | 2 listboxes + checkboxes group |
| selectboxboth@number_edit | 3 textfields + 3 listboxes + checkboxes group |
| selectboxboth@number | 3 listboxes + checkboxes group |
| boolean_edit | 2 radio buttons |
| dropdowndiscrete@date_edit | 1 listbox + 1 textfield + calendar control |
| dropdowndiscrete@date | 1 listbox |
| dropdownrange@date_edit | 2 listboxes + 2 calendar + checkbox group |
| dropdownrange@date | 2 listboxes + checkbox group |
| selectboxdiscrete@date_edit | 1 listbox + 1 textfield + 1 calendar |
| selectboxdiscrete@date | 1 listbox |
| selectboxrange@date_edit | 2 listboxes + 2 calendars + checkbox group |
| selectboxrange@date | 2 listboxes + checkbox group |
| selectboxboth@date_edit | 3 textfields + 3 calendars + 3 listboxes |
| selectboxboth@date | 3 listboxes + checkbox group |
| dropdowndiscrete@time_edit | 3 textfields + 1 listbox |
| dropdowndiscrete@time | 1 listbox |
| dropdownrange@time_edit | 2 listboxes + 6 textfields + checkbox group |
| dropdownrange@time | 2 listboxes + checkbox group |
| selectboxdiscrete@time_edit | 3 textfields + 1 listbox |
| selectboxdiscrete@time | 1 listbox |
| selectboxrange@time_edit | 2 listboxes + 6 textfields + checkbox group |
| selectboxrange@time | 2 listboxes + checkbox group |
| selectboxboth@time_edit | 9 textfields + 3 listboxes + checkbox group |
| selectboxboth@time | 3 listboxes + checkbox group |
| dropdowndiscrete@datetime_edit | 4 textfields + 1 calendar + 1 listbox |
| dropdowndiscrete@datetime | 1 listbox |
| dropdownrange@datetime_edit | 8 textfields + 2 calendars + 2 listboxes |
| dropdownrange@datetime | 2 listboxes + checkbox group |
| selectboxdiscrete@datetime_edit | 4 textfields + 1 calendar + 1 listbox |
| selectboxdiscrete@datetime | 1 listbox |
| selectboxrange@datetime_edit | 8 textfields + 2 calendars + 2 listboxes |
| selectboxrange@datetime | 2 listboxes + checkbox group |
| selectboxboth@datetime_edit | 2 listboxes + 3 calendars + 12 textfields |
| selectboxboth@datetime | 3 listboxes |
| dropdowndiscrete@date2_edit | 4 textfields + 1 calendar + 1 listbox |
| dropdowndiscrete@date2 | 1 listbox |
| dropdownrange@date2_edit | 2 textfields + 2 calendars + 2 listboxes |
| dropdownrange@date2 | 2 listboxes + checkbox group |
| selectboxdiscrete@date2_edit | 1 textfield + 1 calendar + 1 listbox |

| | |
|---|---|
| selectboxdiscrete@date2 | 1 listbox |
| selectboxrange@date2_edit | 2 textfields + 2 calendars + 2 listboxes + checkboxes |
| selectboxrange@date2 | 2 listboxes + checkbox group |
| selectboxboth@date2_edit | 3 textfields + 3 calendars + 3 listboxes + checkboxes |
| selectboxboth@date2 | 3 listboxes + checkbox group |
| cascading_child | iframe with default source for single listbox |
| multi_cascading_child | iframe with default source for multi listbox |
| pop_up_child | 1 single-value listbox |
| multi_pop_up_child | 1 multi-value listbox |

| XML Control Tags | | | |
|---|---|---|---|
| Control Tag | Type | Attributes | HTML output |
| INPUT | TEXT<br>HIDDEN<br>BUTTON<br>CHECKBOX | endtag<br>type<br>name<br>value<br>checked<br>disabled<br>readonly<br>size<br>maxlength<br>src<br>id<br>class<br>lang<br>dir<br>title<br>style<br>align<br>alt<br>usemap<br>ismap<br>tabindex<br>accesskey<br>onfocus<br>onblur<br>onselect<br>onchange<br>onclick<br>ondblclick<br>onmousedown<br>onmouseup<br>onmouseover<br>onkeypress<br>onkeydown<br>onkeyup<br>accept | <Input type=*type* *[Attributes follow]*> |
| LABEL | | endtag<br>for<br>id<br>value<br>class<br>lang<br>dir<br>title | <Label *[attributes follow]*></Label> |

| | | | |
|---|---|---|---|
| | | style<br>accesskey<br>onfocus<br>onblur<br>onclick<br>ondblclick<br>onmousedown<br>onmouseup<br>onmouseover<br>onkeypress<br>onkeydown<br>onkeyup | |
| P | | endtag<br>id<br>class<br>lang<br>dir<br>align<br>title<br>style<br>clear<br>onclick<br>ondblclick<br>onmousedown<br>onmouseup<br>onmouseover<br>onkeypress<br>onkeydown<br>onkeyup | <P *[Attributes follow]*> |
| BR | | endtag<br>id<br>class<br>title<br>style<br>clear | <BR *[Attributes follow]*> |
| SELECT | | endtag<br>Name<br>Size<br>Multiple<br>Id<br>Class<br>lang<br>dir<br>title<br>style<br>disabled<br>tabindex<br>onfocus<br>onblur<br>onchange<br>onclick<br>ondblclick<br>onmousedown<br>onmouseup<br>onmouseover<br>onmousemove<br>onmouseout | <select *[Attributes follow]*></*select*> |

| | | | |
|---|---|---|---|
| | | onkeypress<br>onkeydown<br>onkeyup | |
| IMAGE | | Longdesc<br>Height<br>Width<br>Name<br>Border<br>Hspace<br>Vspace<br>Src<br>Id<br>Class<br>Lang<br>Dir<br>Title<br>Style<br>Align<br>alt<br>usemap<br>ismap<br>tabindex<br>onclick<br>ondblclick<br>onmousedown<br>onmouseup<br>onmouseover<br>onmousemove<br>onmouseout<br>onkeypress<br>onkeydown<br>onkeyup | <IMG *[attributes follow]*> |
| DIV | | Align | <div> |
| IFRAME | | Longdesc<br>Id<br>Name<br>Class<br>Lang<br>Title<br>Style<br>Src<br>Width<br>Height<br>Scrolling<br>Frameborder<br>Marginwidth<br>Marginheight<br>Align | <IFrame> |

An example of how to use the above XML Control API follows in the XML snippet below. The snippet presents a guideline on how the controls should be written in to an XML file and later pasted in to the “Custom Controls” section of the MMC.

It is expected that developer building the custom control possess a good working knowledge of XML in order to perform this task successfully.

## 10.1 XML Custom Control sample snippet

The outermost encapsulating tag in the controls repository is the “controlsfactory” In designing the custom control nodes, all of the entries should be placed between this tag pair.

In the example provided below, there is a definition of the control functional representation that is called “sometype”. The control type “sometype” is here defined within its encapsulating pair “paramcontrols”.

```
<? xml version="1.0" encoding="UTF-8"?>
<controlsfactory>
    <paramcontrols ptype="sometype">
            <control tag="LABEL" endtag="true" for="" id="label" value=""
class="" lang="" dir="" title="" style="" accesskey="" onfocus="" onblur=""
onclick="" ondblclick="" onmousedown="" onmouseup="" onmouseover=""
onmousemove="" onmouseout="" onkeypress="" onkeydown=""
onkeyup=""/>
            <control tag="INPUT" endtag="false" type="CHECKBOX"
name="" value="" checked="false" disabled="false" readonly="false" size=""
maxlength="" src="" id="" class="" lang="" dir="" title="" style="" align=""
alt="" usemap="" ismap="" tabindex="" accesskey="" onfocus="" onblur=""
onselect="" onchange="" onclick="" ondblclick="" onmousedown=""
onmouseup="" onmouseover="" onmousemove="" onmouseout=""
onkeypress="" onkeydown="" onkeyup="" accept=""/>
            <control tag="SELECT" endtag="true" name="" size="1"
multiple="true" id="" class="" lang="" dir="" title="" style=""
disabled="false" tabindex="" onfocus="" onblur="" onchange="" onclick=""
ondblclick="" onmousedown="" onmouseup="" onmouseover=""
onmousemove="" onmouseout="" onkeypress="" onkeydown=""
onkeyup="">
                    <celement tag="OPTION" endtag="true" selected="false"
disabled="false" label="" value="" id="" class="" lang="" dir="" title=""
style="" onclick="" ondblclick="" onmousedown="" onmouseup=""
onmouseover="" onmousemove="" onmouseout="" onkeypress=""
onkeydown="" onkeyup=""/>
            </control>
            <control tag="TEXTAREA" endtag="true" name="" rows="10"
cols="20" id="" class="" lang="" dir="" title="" style="" disabled="false"
readonly="false" tabindex="" accesskey="" onfocus="" onblur="" onselect=""
onchange="" onclick="" ondblclick="" onmousedown="" onmouseup=""
onmouseover="" onmousemove="" onmouseout="" onkeypress=""
onkeydown="" onkeyup=""/>
    </paramcontrols>
</controlsfactory>
```

Let’s consider building a very popular and practical control, an HTML text field. This control is represented through the following XML code:

```
<control tag="INPUT" endtag="false" type="CHECKBOX" name=""
value="" checked="false" disabled="false" readonly="false" size=""
maxlength="" src="" id="" class="" lang="" dir="" title="" style="" align=""
alt="" usemap="" ismap="" tabindex="" accesskey="" onfocus="" onblur=""
onselect="" onchange="" onclick="" ondblclick="" onmousedown=""
onmouseup="" onmouseover="" onmousemove="" onmouseout=""
onkeypress="" onkeydown="" onkeyup="" accept=""/>
```

It is fundamental to define a particular control within the tag “control”, and then the HTML definition of this control will fall into place by being positioned within this particular control pair.

The first attribute is “tag”, and defines the tag name that will be built for this HTML representation.

Another important attribute is “endtag”, which is designed to carry the Boolean value of either true or false. The “endtag” attribute will indicate if the closing tag is necessary to be built within the HTML realm (as per W3 specification, some tags are containers and some are stand-alone).

In general, the approach is to completely adhere to the W3 specifications for the HTML tags, so that the succession of attributes will basically show all of the attributes that can be associated with this particular HTML control representation.

There is also a complete set of attributes in this XML snippet that define the events normally known to be fired on this particular control. These events are JavaScript functions handled events such as:

- onfocus=""
- onblur=""
- onselect=""
- onchange=""
- onclick=""
- ondblclick=""
- onmousedown=""
- onmouseup=""
- onmouseover=""
- onmousemove=""
- onmouseout=""
- onkeypress=""
- onkeydown=""
- onkeyup=""

When the custom controls are designed, it is expected that the developer will design and develop a JavaScript API of client-side functions that will render and service the controls that are built from the custom controls XML.

Since the HTML is dynamically assembled in the backend with the help of the SPL 2 COM object's internals the reference to these JavaScript functions will be carried on to the client, through inclusions that are delivered to the browsers.

The JavaScript function names are referenced in the values of this XML snippet that represents this particular control in HTML.

For example, consider associating a JavaScript function with the event of "on focus" to call the displayInstructions(this). The code for this will be presented in the following manner:

- onfocus=" displayInstructions(this);"

This would be exactly as a developer would have defined a function that is associated with handling an HTML control server-side. It is assumed that the function "displayInstructions(this)" is pre-defined in a JavaScript file that is used as an inclusion in this dynamically constructed HTML page. The complete path reference is provided in the "<script>" section that is relevant to the code location on the server.

It is also certainly possible that a certain control may have a child control that is encapsulated by the outer parent control, such as that of a HTML "<select>" control with its child "<option>" elements. The XML DTD is absolutely flexible to accommodate this if required:

```
<control tag="SELECT" endtag="true" name="" size="1" multiple="true"
id="" class="" lang="" dir="" title="" style="" disabled="false"
tabindex="" onfocus="" onblur="" onchange="" onclick="" ondblclick=""
onmousedown="" onmouseup="" onmouseover="" onmousemove=""
onmouseout="" onkeypress="" onkeydown="" onkeyup="">
<celement tag="OPTION" endtag="true" selected="false" disabled="false"
label="" value="" id="" class="" lang="" dir="" title="" style=""
onclick="" ondblclick="" onmousedown="" onmouseup=""
onmouseover="" onmousemove="" onmouseout="" onkeypress=""
onkeydown="" onkeyup=""/>
</control>
```

The child control/element is defined as the "<celement>" to accommodate the necessity to build "<option>" elements in this case.

When the developer wishes to build any HTML control that assumes the use of child elements, they are expected only to add this XML representation to their control display definition. Note that this is for consistency, as the attribute "tag" will bear the name of the tag, which represents the child element.

Finally, the closing “</control>” tag identifies the end of the encapsulating tag pair.

It is important to realize, that to save on the processing time and the HTML build speed, the attributes in the XML definition have their values left empty intentionally. This means that they will not be built, as they are not intended for use.

Therefore, only those attributes that were assigned values, will be built. For example:

> style="" attribute is left blank, this attribute will not be built, but style="selector” will reference the css class called “selector”, thus assuming that for rendering this control, this look-and-feel will be used.

Most of the JavasScript API and XML controls (that are transformed into the HTML layer with the handlers), are stored in the “SPL Repository”.