



AGILEXRM REFERENCE ARCHITECTURE

Table of Contents

1.	Introduction	4
1.1	Disclaimer of warranty	4
1.2	AgileXRM components	5
1.3	Access from PES to AgileXRM Process Engine Database	7
1.4	Access from PES to XRM Server when process activities are executed	7
1.4.1	Runtime Security Configuration	9
1.5	Access from PES to XRM Server to synchronize users	13
1.6	Access from AgileMonitor to PES, PES Database and XRM Server to retrieve data.	14
1.7	Access from Process Manager PES and XRM Server to retrieve data	17
1.7.1	Process Manager Security Configuration	17
1.7.2	Access from Process Manager to PES	19
1.8	Access from SharePoint to PES and XRM Server to show webparts	22
1.8.1	SharePoint External Connector	26
1.9	Access from Envision to PES and XRM Server to retrieve metadata and set process template configuration	32
1.10	Access to PES from XRM Server to start processes and set activity information	34
2.	AgileLightForms Security	38
3.	Connections from Client Browser to AgileLightForms Server	39
3.1	Form design	39
3.2	Form usage	41
3.2.1	CRM Forms	41
3.2.2	External Forms	43
4.	Connections to CRM Server from AgileLightForms Server	45
4.1	Form Storage	45
4.2	Connections with credentials	45
4.3	Design-time connections without credentials	46
4.4	Run-time connections without credentials	47
4.4.1	Connections with Process Template Permissions	47
4.4.2	Connections with System Permissions	48
4.4.3	Connections with Form User Permissions	48
4.4.4	Connections with Process Template Owner Permissions	48
4.4.5	Connections with Process Initiator Permissions	49
5.	Connections to AgilePoint Server from AgileLightForms Server	49
5.1	AgileLightForms Connector	49
5.2	Connections with credentials	50
5.3	Connections without credentials	50
6.	AgileDialogs Security (only for CRM 2011)	50
6.1	Access from AgileDialogs Engine to PES	51
6.1.1	Web Service Access	51
6.1.2	WCF using HTTP Binding	51

6.2	Access from PES to AgileDialogs Engine.	51
6.3	Access from AgileDialogs Engine to XRM Server.	51

AgileXRM Reference Architecture

1. Introduction

This document serves as a description of all elements in an AgileXRM deployment and how they interact from the point of view of security.

1.1 Disclaimer of warranty

AgilePoint Inc. makes no representations or warranties, either express or implied, by or with respect to anything in this document, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect, special or consequential damages.

Copyright © 2012, AgilePoint Inc. All rights reserved.

GOVERNMENT RIGHTS LEGEND: Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the applicable AgilePoint Inc. license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (Oct 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14, as applicable.

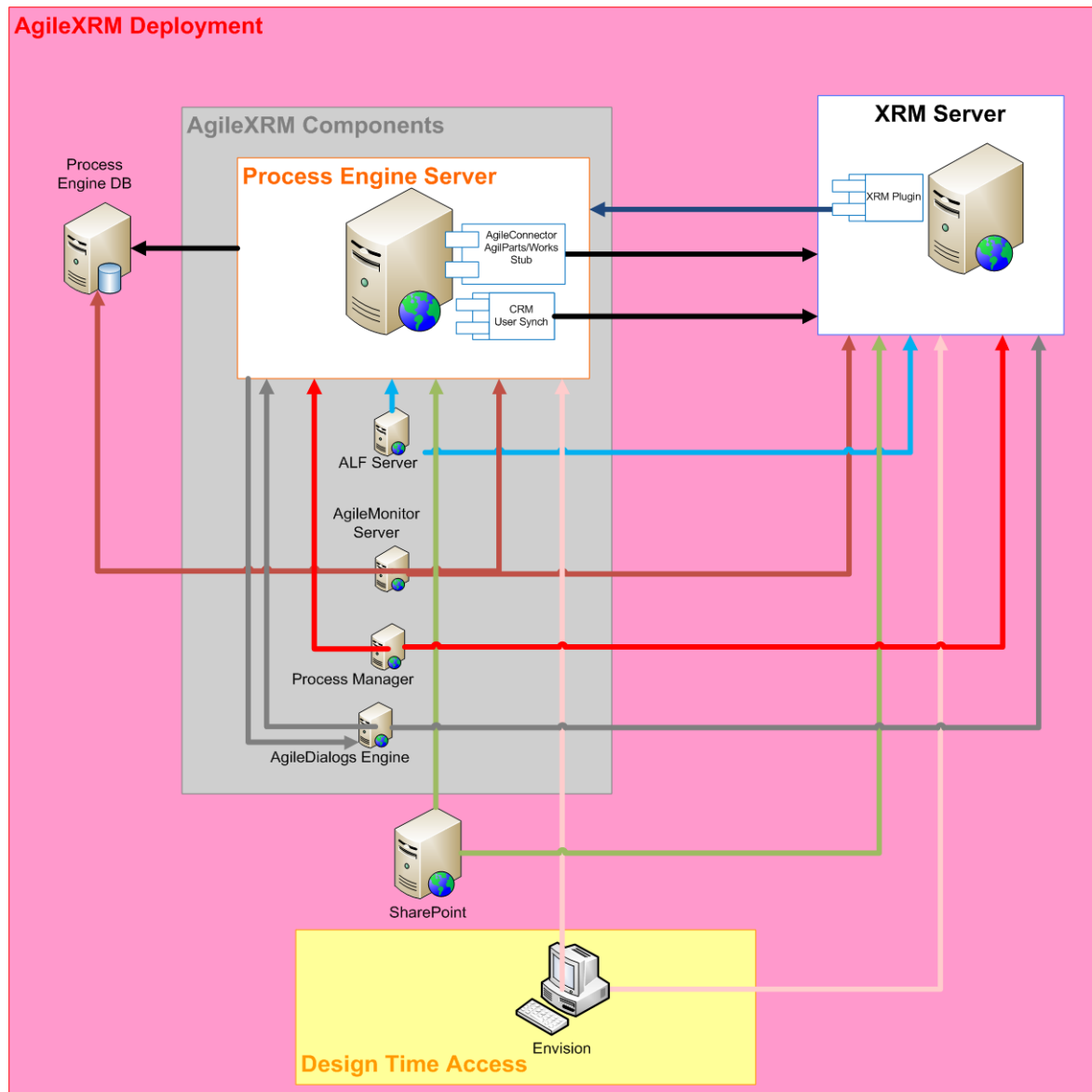
'AgilePoint Inc.' and all its products are trademarks of AgilePoint Inc.. References to other companies and their products use trademarks owned by the respective companies and are for reference purposes only.

1.2 AgileXRM components

These are the pieces involved in AgileXRM deployment

- AgileXRM Process Engine Database: SQL Server database.
- AgileXRM Process Engine Server (PES): IIS application
- AgileLightForms Server (ALF Server): IIS Application with Silverlight and ASP.NET
- AgileXRM CRM plug-in: plug-in deployed to CRM.
- AgileMonitor Server: IIS Application With Silverlight and ASP.NET
- Process Manager: IIS Application With Silverlight and ASP.NET
- SharePoint (Optional)
- Process Designer (Envision): Add-on to Visio
- Microsoft Dynamics CRM Server (XRM Server).
- AgileDialogs (only for CRM 2011) : IIS Application with Silverlight and ASP.NET

Each of these components can be set in the same machine or in different physical or virtual machines.



These are the scenarios where security configuration is involved:

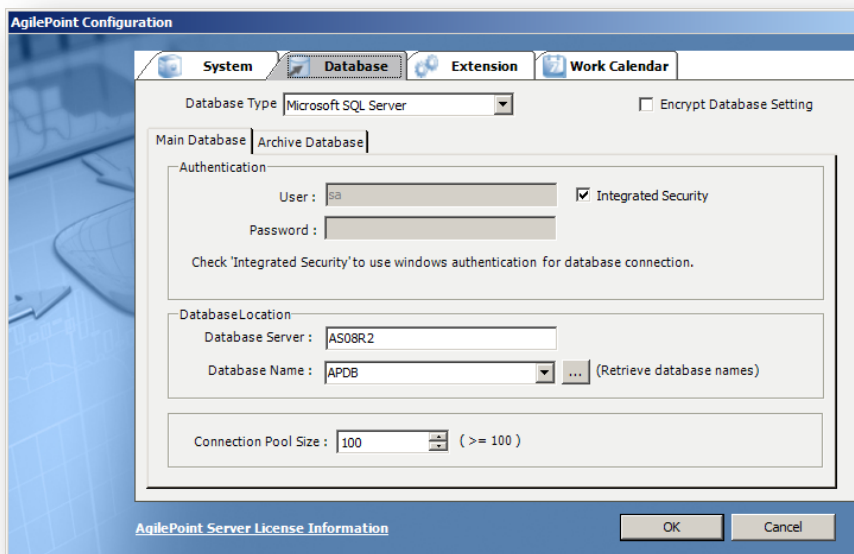
- Access from PES to AgileXRM Process Engine Database.
- Access from PES to XRM Server when process activities are executed.
- Access from PES to XRM Server to synchronize users.
- Access from AgileMonitor to PES, PES Database and XRM Server to retrieve data.
- Access from Process Manager PES and XRM Server to retrieve data.
- Access from SharePoint to PES and XRM Server to show *webparts*.
- Access from Envision to PES and XRM Server to retrieve and set process template configuration.
- Access to PES from XRM Server to start processes and set activity information.
- Access from ALF to PES to start processes and complete activities.
- Access from ALF to XRM Server to read and write data.
- Access from AgileDialogs Engine to PES.

- Access from PES to AgileDialogs Engine.
- Access from AgileDialogs Engine to XRM Server.
- User Access.

1.3 Access from PES to AgileXRM Process Engine Database

PES stores runtime information in a database. This database contains information about running processes, activity instances, assigned tasks, ...

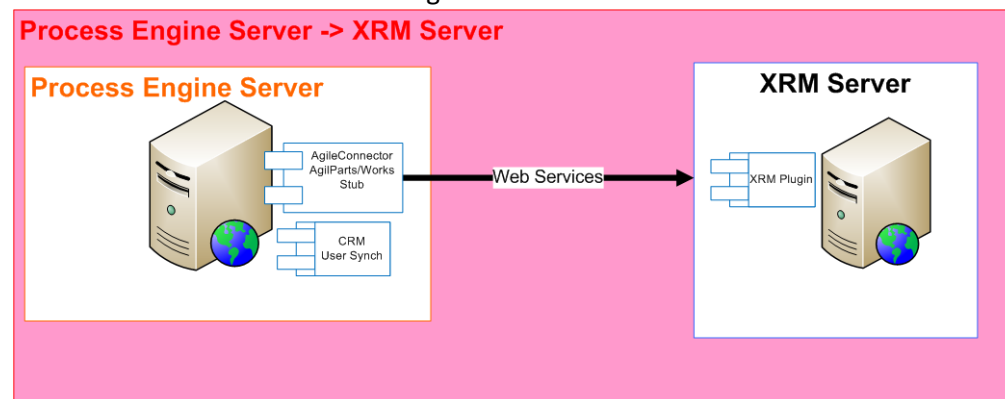
The configuration of this access is set in AgilePoint Server Configuration. This tool can be opened in the server where PES is installed.



In “Database” tab authentication can be set to integrated security (in this case PES will access to database using the credentials of the application pool where PES is running) or a specific user and password can be set.

1.4 Access from PES to XRM Server when process activities are executed

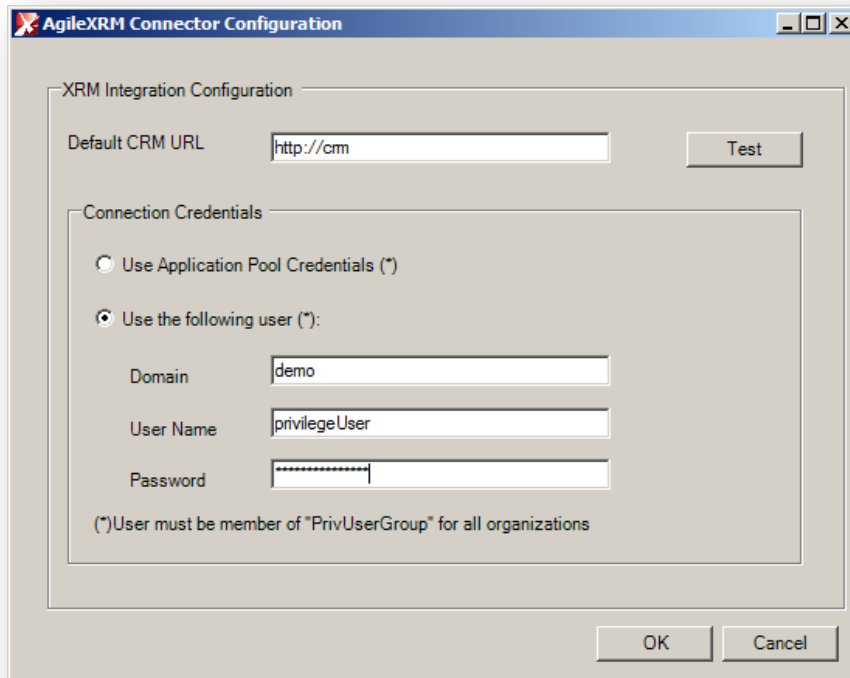
PES interacts with XRM Server using web services.



CRM provides mechanisms to impersonate users when a web service call is done to allow make this calls on behalf of a specific user.

This is achieved using a specific Active Directory group (**PrivUsersGroup**) that contains the users that are allowed to impersonate other users.

In the case of Web Service calls from PES to XRM Server, there are two options to set which user to use as the user allowed to impersonate. This can be set in CrmConnector configuration:



In Connection Credentials, the connector can be configured to use either Application Pool User credentials (that is, when connecting to CRM web service calls will use this user windows credentials) or a specific user (this user will be used to set Web Service call credentials, in this sample, *privilegeUser* will be used to connect to CRM).

This user must belong to *PrivUserGroup* in order to allow CRM impersonation.

This user must be a CRM licensed user and must have a set of minimum permissions:

- Read permissions in system user entity at organization level.
- Read permissions in *AgileProcessTemplate* entity at organization level.
- Read permissions in customizations. In customizations tab in role permissions editor set read permissions for Entity, Attribute and relationship.

Depending on process runtime security configuration this user may need more permission. If a process template is configured to run as system, actions in CRM will be executed as this user, so this user must have more permissions in this scenario (See Runtime Security Configuration).

1.4.1 Runtime Security Configuration

There are 2 aspects of security that can be configured at process template level:

1. Which user will act in CRM from PES when process activities are executed. For instance, if an activity in a process executes a query in CRM depending on security configuration this query could retrieve different records based on CRM visibility.
2. Which roles have permission to do what in process instances. For instance which roles can initiate a process, cancel a process,... By default **users that belong to Administrator Role in PES have all permissions.**

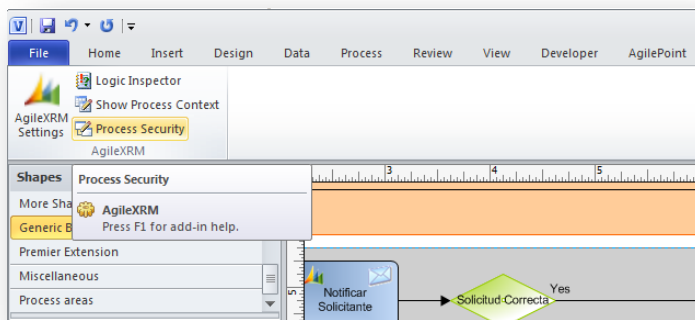
Now we are going to focus on point 1.

There are several custom entities added by AgileXRM to an organization in CRM to manage security:

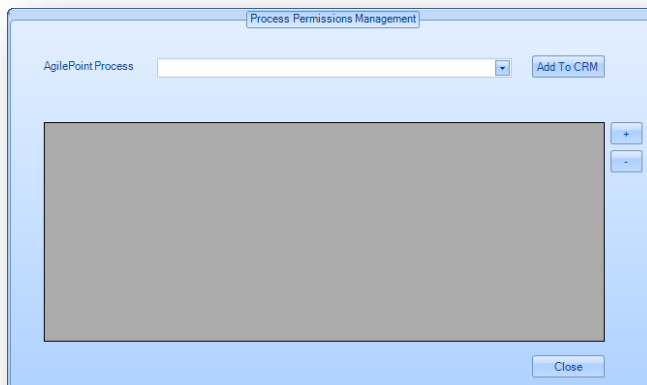
- *AgilePointProcessTemplate* (*ascentn_agilepointprocesstemplate*).
- *AgilePointTemplatePermission* (*ascentn_agilepointtemplatepermission*).

The first stores information about the process templates that has been deployed to AgileXRM and the second has permissions for these templates (these are related to point 2).

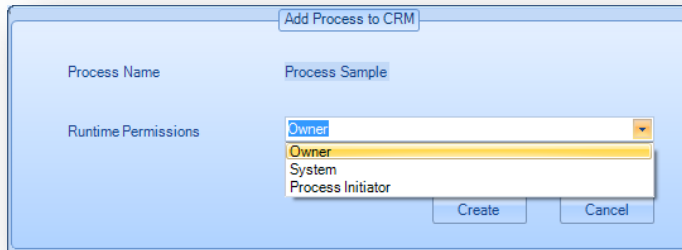
CRM administrator can add these entities to Settings or other section to allow administration from CRM. Records of these entities can be managed from Envision in run time permission management Process Security button in AgileXRM Ribbon):



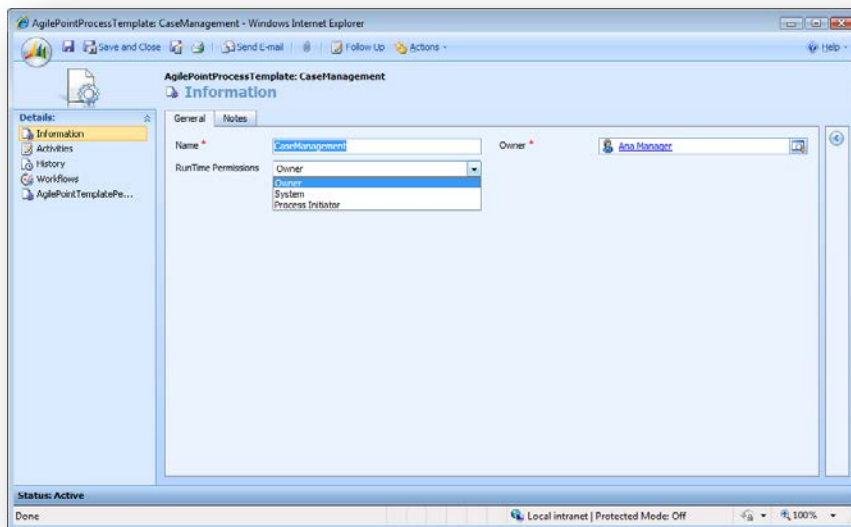
When a new process is created, if the user has permissions to create *AgilePointProcessTemplate*, a new record can be created using this interface:



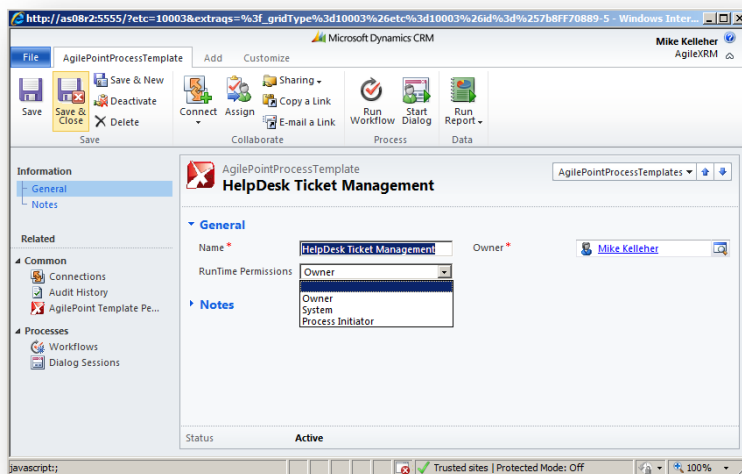
Using Add to CRM button a new record is created in CRM:



When Create is clicked a new *AgilePointProcessTemplate* record is created. This is the form for *AgilePointProcessTemplate* in CRM:

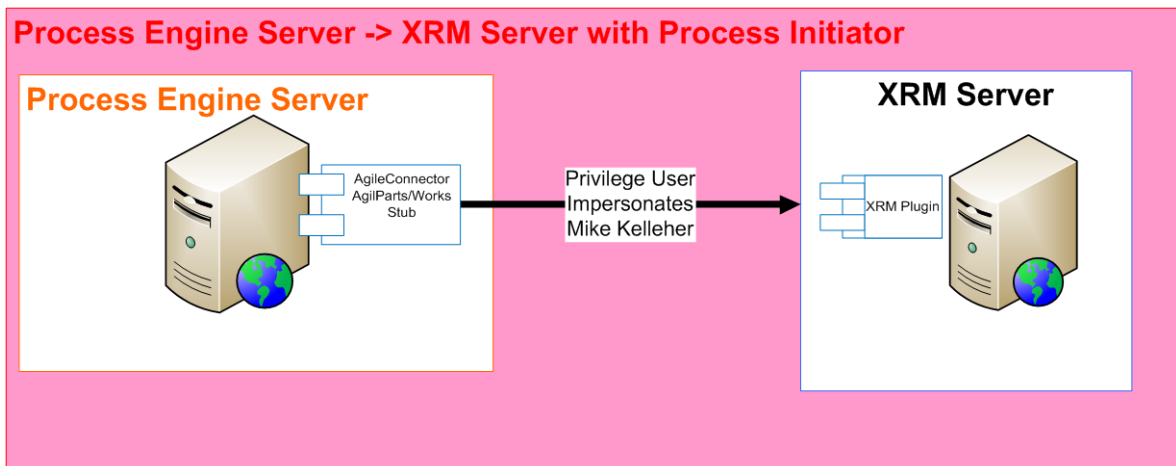


This is the same form in CRM 2011:



- The field Name is the name of the process template, this is set in Envision **and must be equal** to this value. When a process template is renamed in Envision, security configuration must be updated.
- The field Run Time Permissions specifies how PES will interact with CRM. There are 3 options to set permissions:
 - Owner
 - System
 - Process Initiator

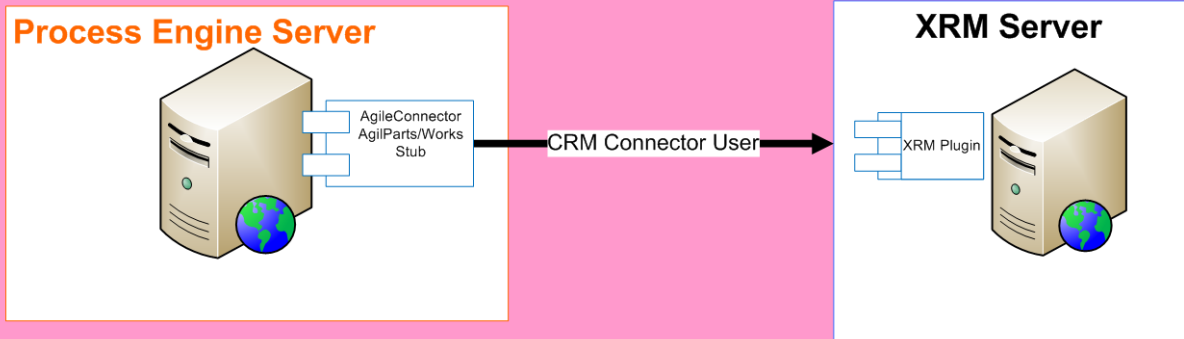
If the value is **Owner** when PES connects to CRM the user that is set of Owner of this record. In this sample, *Mike Kelleher* is the owner of the record *CaseManagement* so, all calls from PES to CRM while executing Case Management process instances, will be done impersonating Ana. If there is a query activity to retrieve Accounts, this query will only return Accounts that Ana can see in CRM. If the process executes an action that Ana cannot perform in CRM, for example update a record that Ana does not have permissions to write to, the process will throw an exception. CRM administrator can change the value of the owner in order to increase or decrease process execution permissions.



Note 1: In order to start processes this owner user must have read/write permissions for AgilePointProcess entity (ascent_agilepointprocess). When a process is started a record of this entity is created to store the relation between the main entity (the Account, Opportunity, ...) and the process instance in PES.

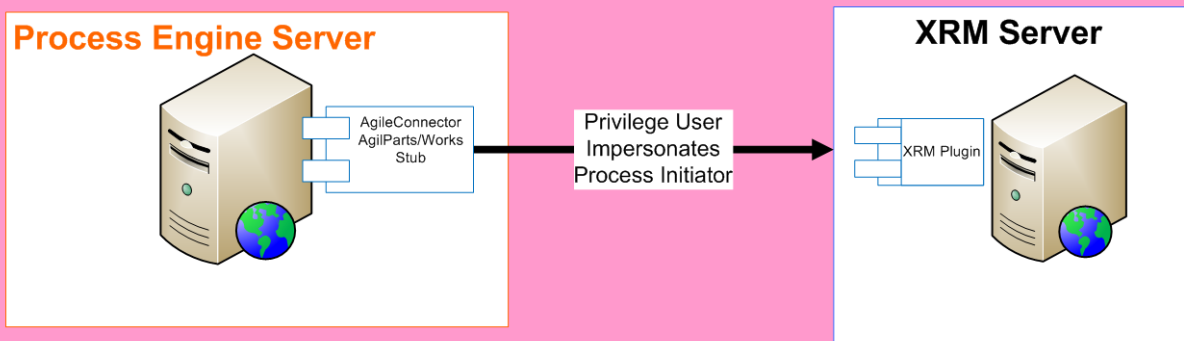
If value is **System**, all interactions between PES and CRM will be done on behalf of the user that is configured in CrmConnector, that means that in this case there is no impersonation using CRM mechanism. In this case, this user should have more permission depending on what the process is doing. If the process creates an account record, the user must have permission in CRM. **Note 1** applies in this case too, the user configured in CrmConnector must have read/write permissions in *AgilePointProcess*.

Process Engine Server -> XRM Server with System



When value is **Process Initiator** interactions from PES to CRM will impersonate to the user that started the process. Note 1 applies to these users too.

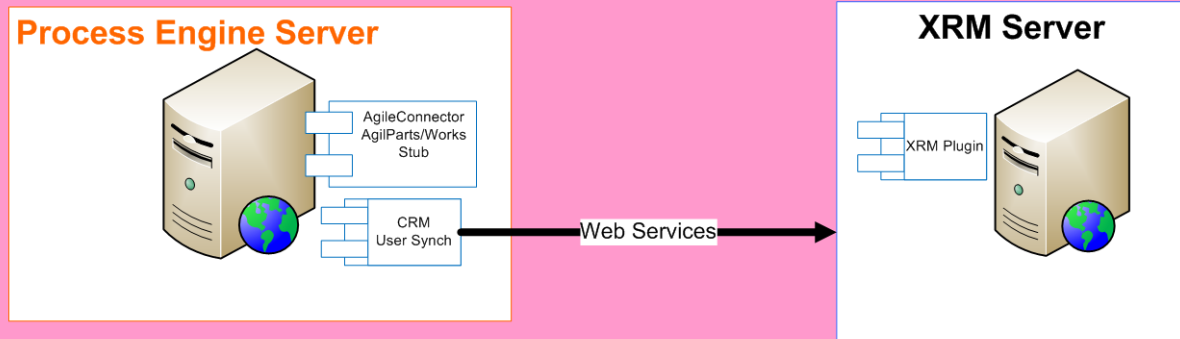
Process Engine Server -> XRM Server with Process Initiator



1.5 Access from PES to XRM Server to synchronize users

AgileXRM provides a specific connector to import users from CRM to PES users repository. This connector uses web services to retrieve information from CRM.

Process Engine Server -> XRM Server User Synch Connector



This connector connects to XRM Server using Web Services with the credentials of the user configured in CRM Connector in PES.

The connector is located in AgilePoint server bin directory in Ascentn.Crm.UserSynch.dll.

1.6 Access from AgileMonitor to PES, PES Database and XRM Server to retrieve data.

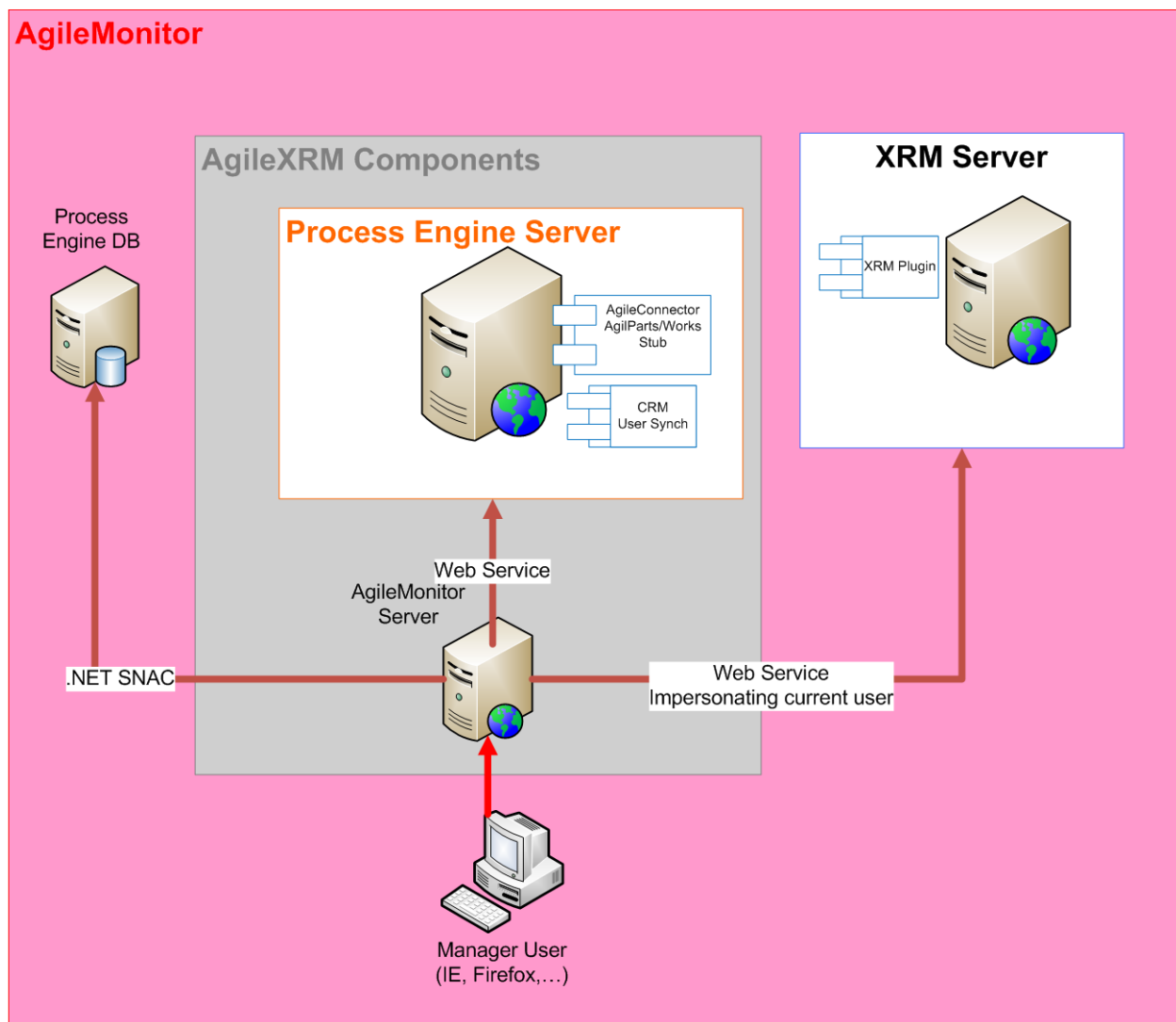
AgileMonitor is an ASP.Net application that uses a Silverlight based UI.

This applications mixes information from PES and XRM server to create dynamic graphic reports using the process drawing as canvas to show information.

In order to do that AgileMonitor server needs to access to XRM Server data through services (CRM Service and Metadata Service), PES using Web Services and PES DB using .NET SQL Server Native Client. AgileMonitor connects to CRM impersonating the user that has accessed to AgileMonitor, that's why the user will see only data related to records that CRM allow him to see. The credentials used to connect to the web service are the credentials of the Application Pool that executes AgileMonitor. The user configured in this *AppPool* must belong to *PrivUserGroup*.

This application pool user must have permissions to access to PES server ad PES DB too.

The connection string to access to PES DB is stored AgileMonitor *web.config*.

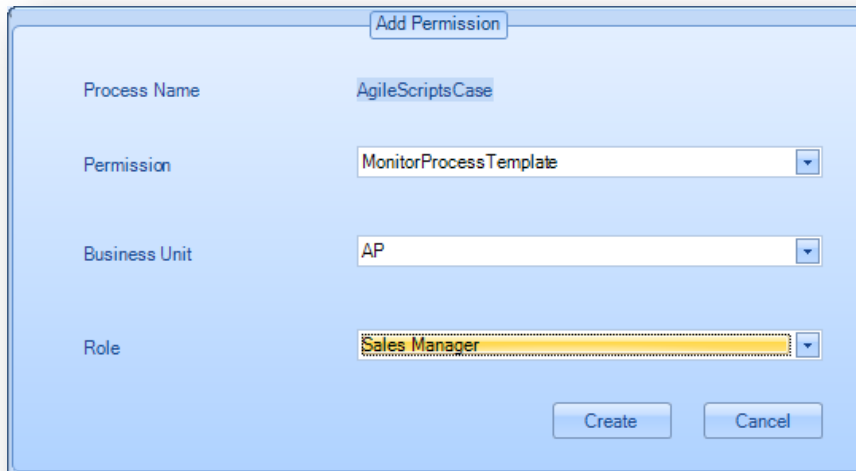


The user can monitor processes using a web explorer.

In order to monitor instances of a specific process the user must belong to a CRM role that has permission.

These permissions are stored in an entity called *AgilePointTemplatePermission*. This entity is related to *AgilePointProcessTemplate* entity, there are many *AgilePointTemplatePermissions* for each *AgilePointProcessTemplate*.

The permission needed to monitor a process is called *MonitorProcessTemplate* and can be either assigned from Envision:

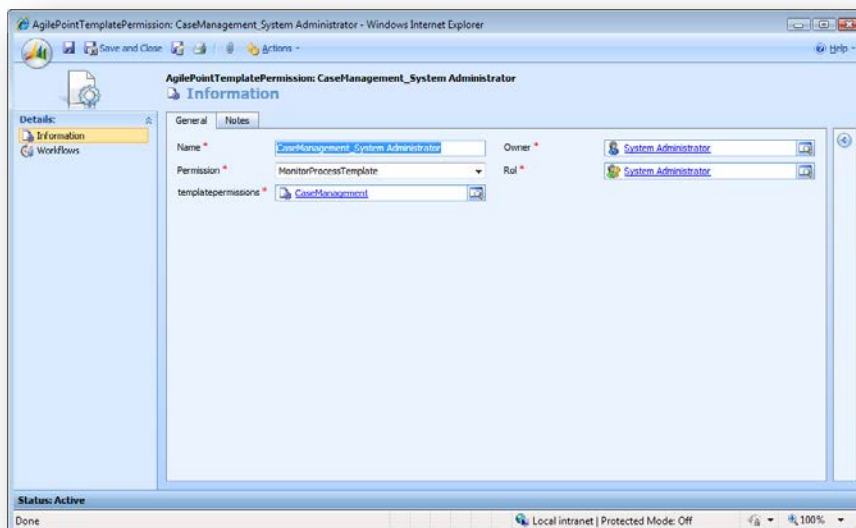


The 'Add Permission' dialog box is shown with the following fields:

- Process Name: AgileScriptsCase
- Permission: MonitorProcessTemplate
- Business Unit: AP
- Role: Sales Manager

Buttons: Create, Cancel

Or from *AgilePointTemplatePermission* form in CRM:

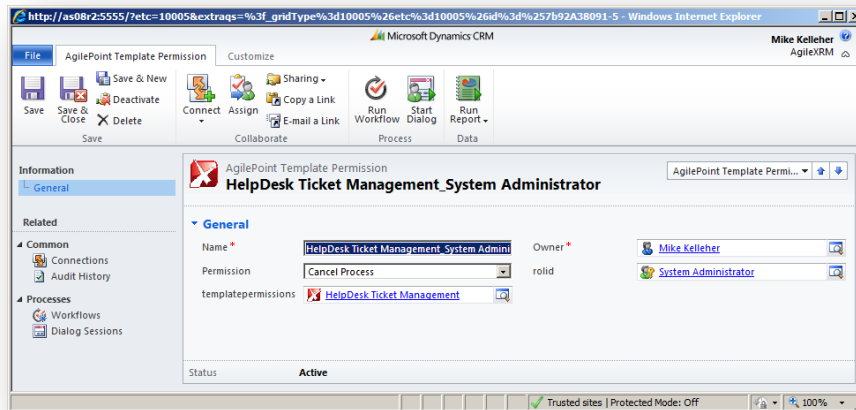


The screenshot shows the 'AgilePointTemplatePermission: CaseManagement_System Administrator' form in a Windows Internet Explorer browser. The form is titled 'Information' and has tabs for 'General' and 'Notes'. The 'General' tab is active, showing the following fields:

- Name: CaseManagement_System Administrator
- Owner: System Administrator
- Permission: MonitorProcessTemplate
- Role: System Administrator
- templatepermissions: CaseManagement

The status bar at the bottom indicates 'Status: Active' and 'Local intranet | Protected Mode: Off'.

This is the same form in CRM 2011:

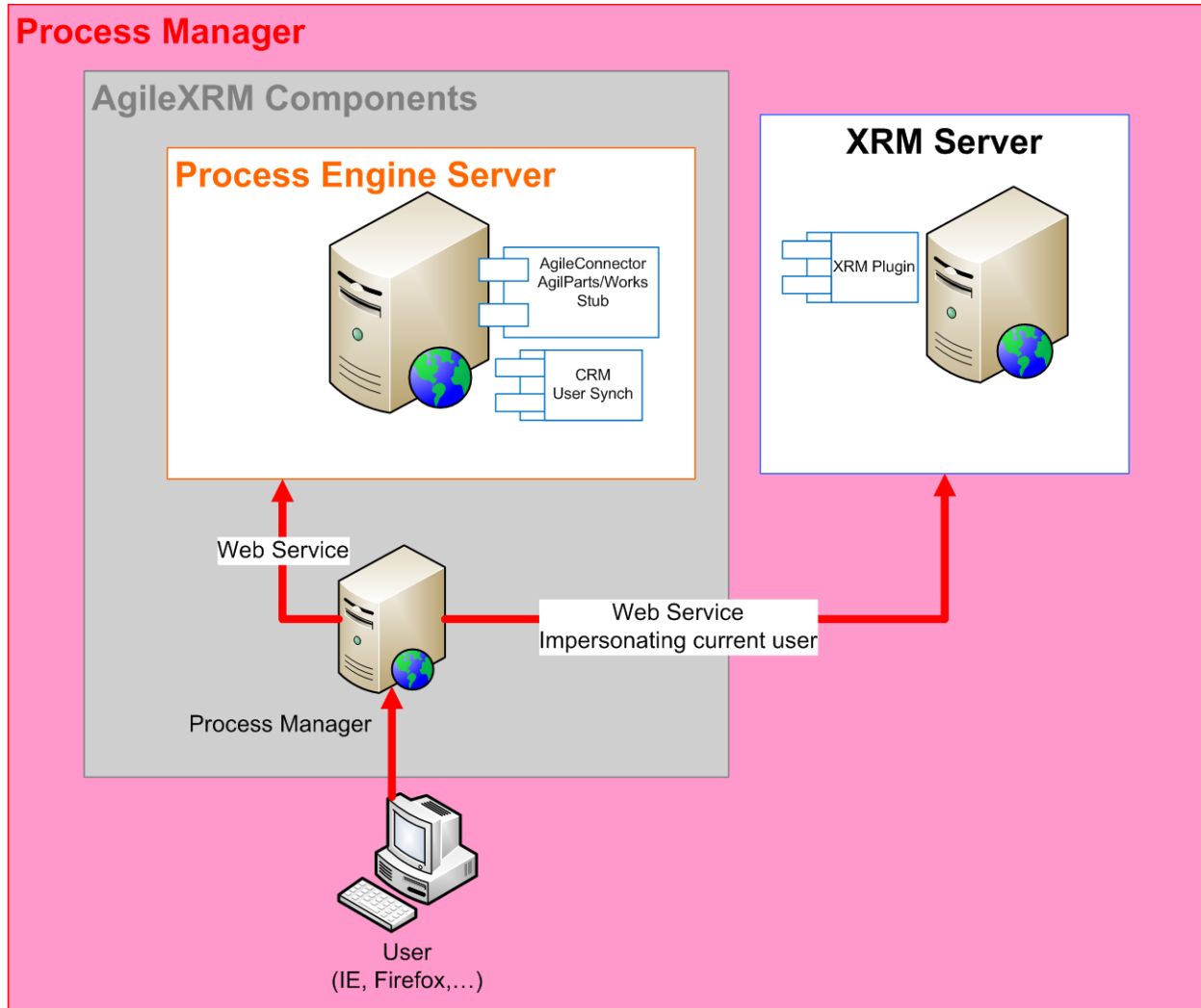


1.7 Access from Process Manager PES and XRM Server to retrieve data

Process Manager is an ASP.NET application with Silverlight UI that accesses to PES and XRM Server to retrieve information about a specific process instance.

Process Manager connects to XRM server through Web Services impersonating the user that has accessed to the web application using *PrivUserGroup* mechanism. The Application Pool user configured in IIS must belong to *PrivUserGroup*.

The user that is viewing the process must have at least permission to view it (this permission is set using *AgilePointTemplatePermission*) and permissions to read *AgilePointProcess* records.



1.7.1 Process Manager Security Configuration

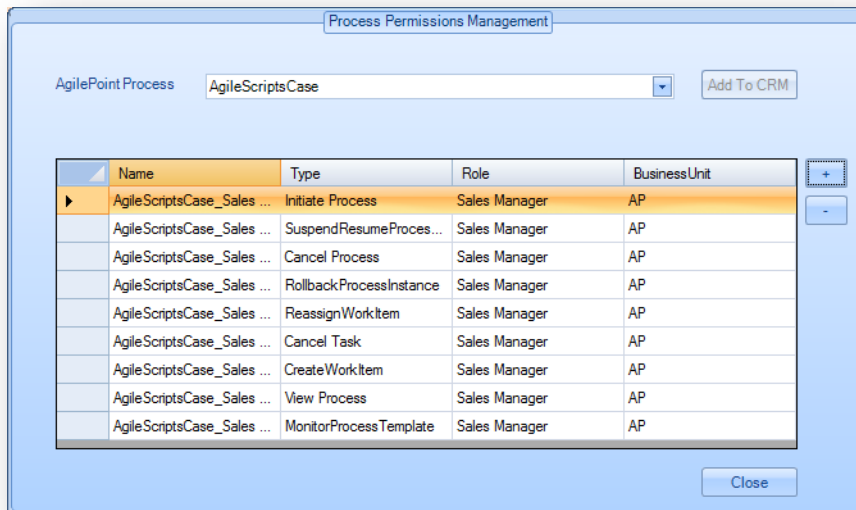
There are several actions that a user can perform in a process instance using Process Manager:

- View
- Cancel Process
- Suspend/Resume Process
- Migrate Process
- Change Flow
- Reassign Task

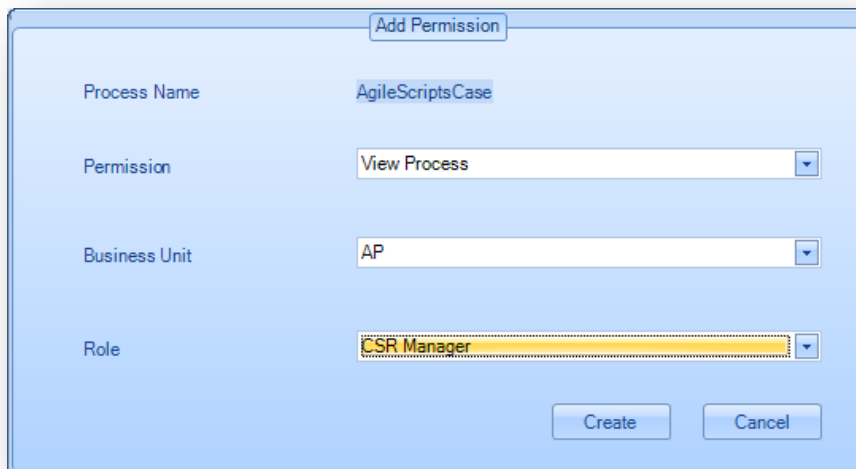
- Create Linked WorkItem

Each action has a corresponding permission that is stored in *AgilePointTemplatePermission* entity. For each process template (stored in *AgilePointProcessTemplate* entity) there are multiple permissions. These permissions can be created from Envision or from CRM forms.

In Envision this is the window to set permissions:

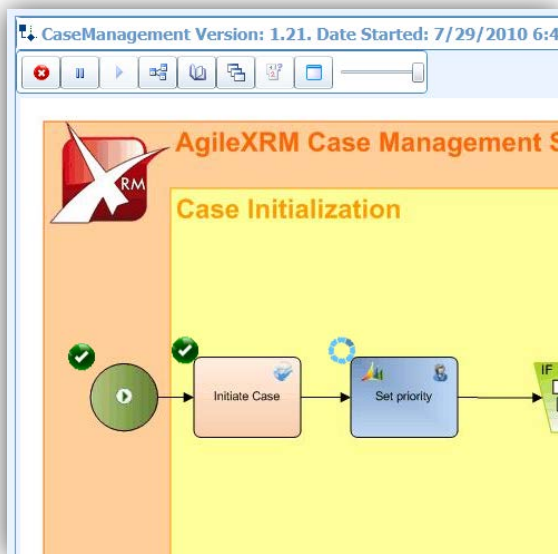


The button + is used to add permissions, when this button is clicked this window is opened:



This window allows giving a specific permission to a role in CRM.

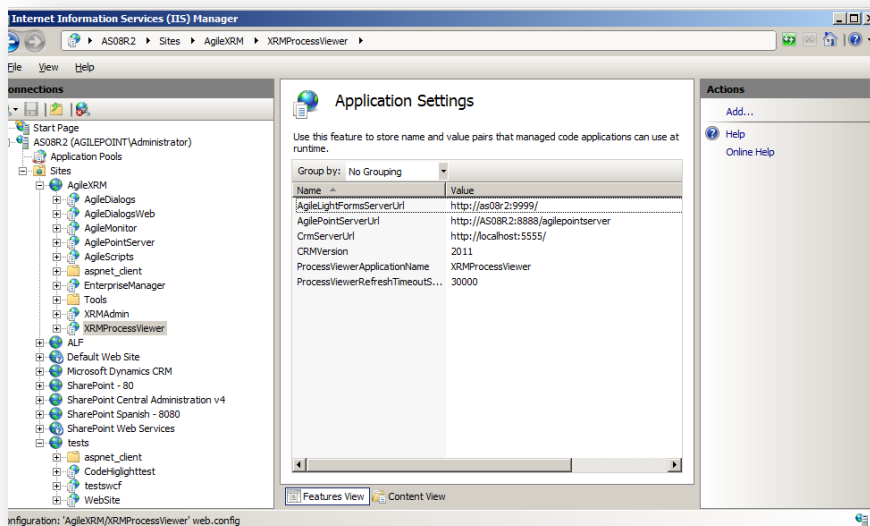
Based on these permissions and the roles that the user belongs to, Process Manager allows the user execute corresponding actions.



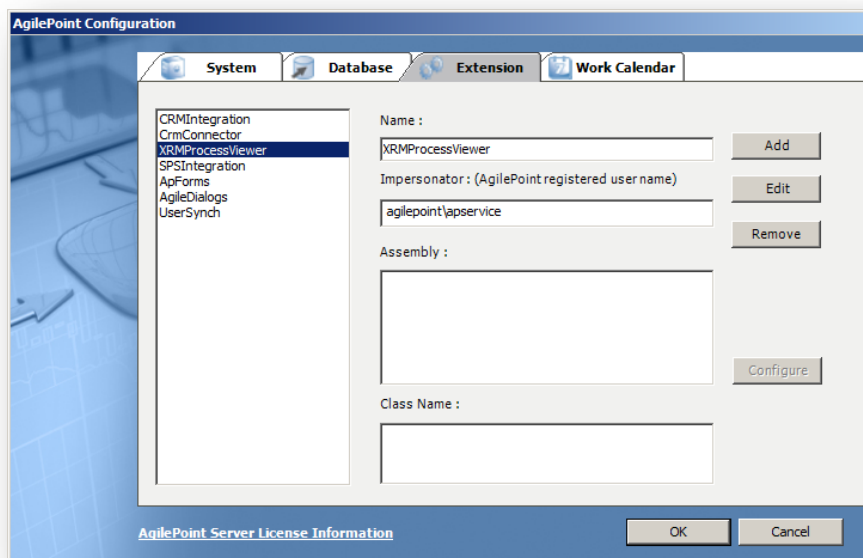
In this sample the user can perform all actions in this process instance (cancel process, suspend, change flow,...). All actions appear in the toolbox on top of Process Manager.

1.7.2 Access from Process Manager to PES

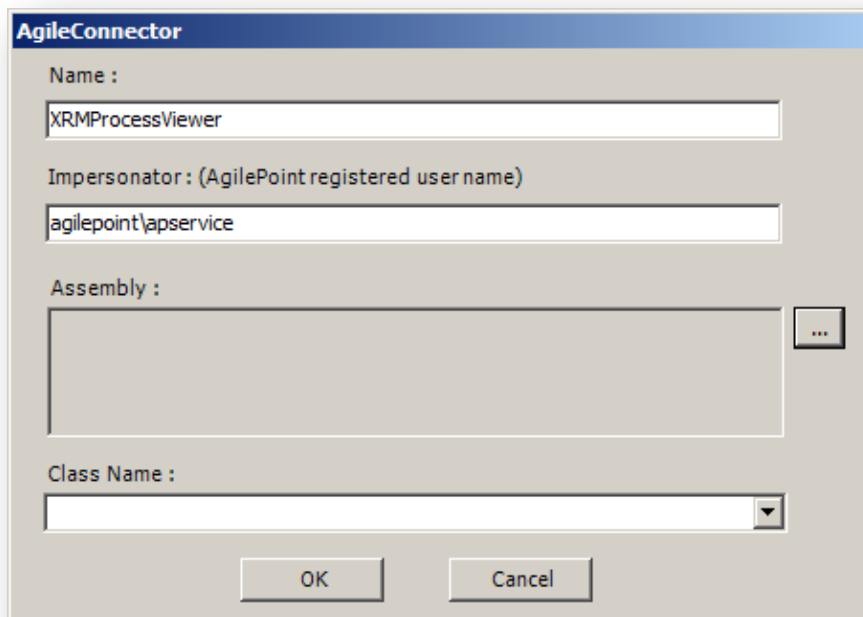
When Process Manager connects to PES uses the application name configured in the web application configuration:



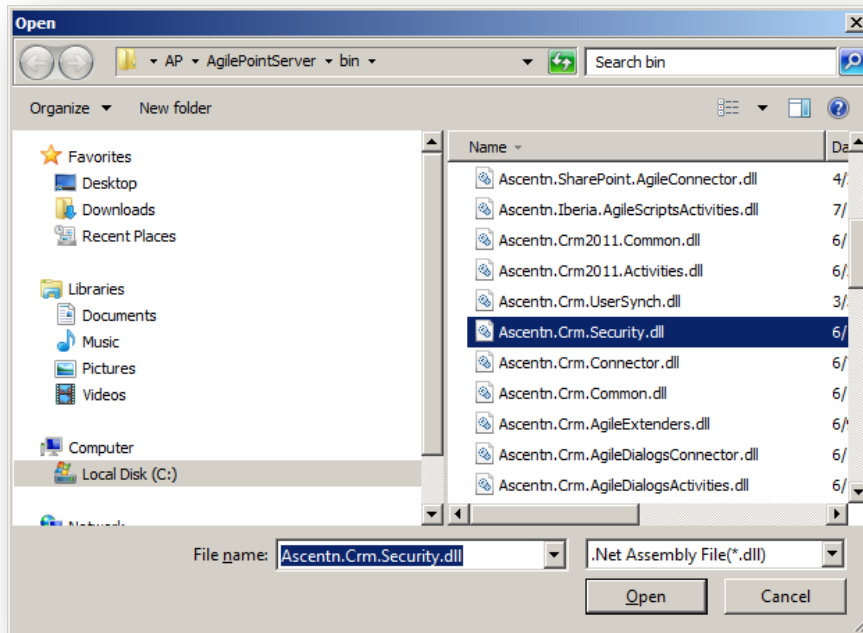
In this sample *XRMProcessViewer*. This application name must be configured in AgilePoint Configuration as an extension:



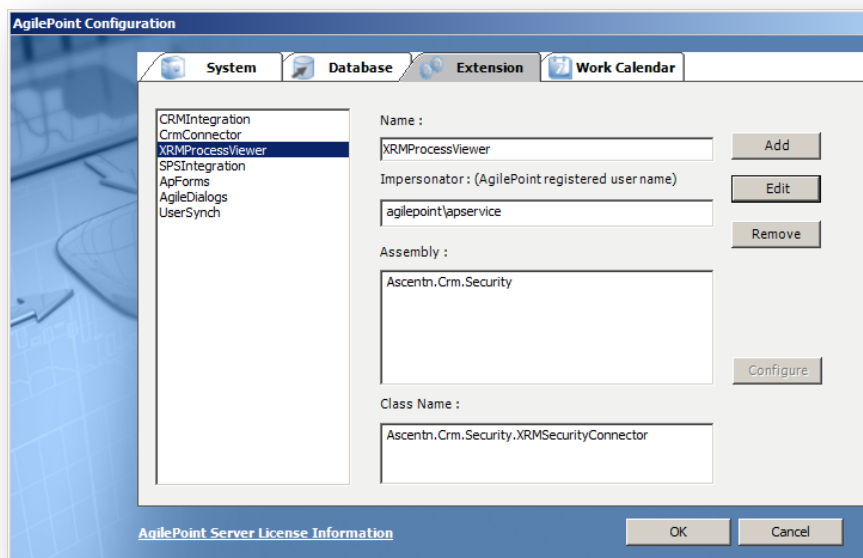
This extension must be configured to apply AgileXRM security. To apply this security click **Edit** :



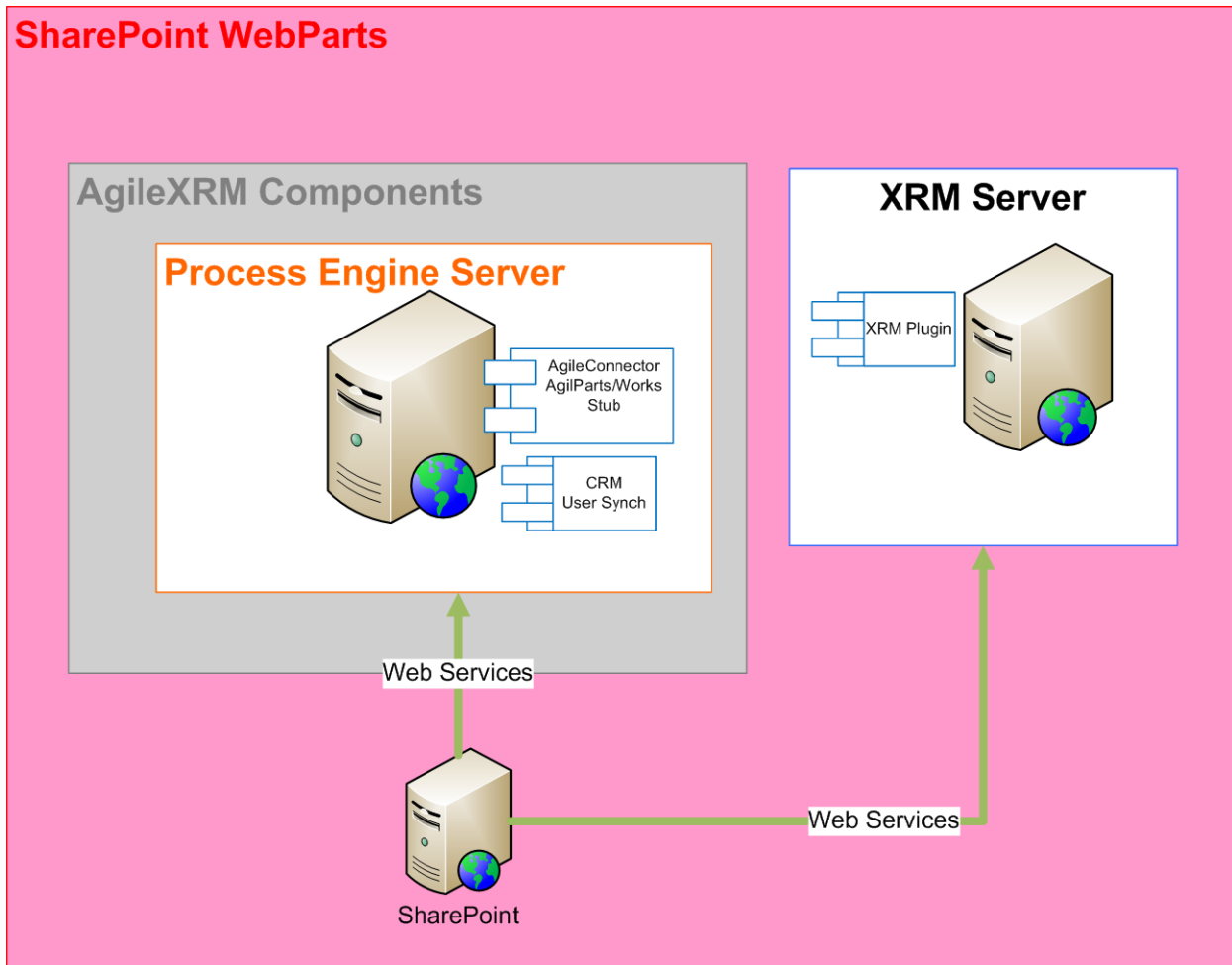
Click ellipsis button and select *Ascentn.Crm.Security.dll*:



Click OK to save this configuration:



1.8 Access from SharePoint to PES and XRM Server to show webparts

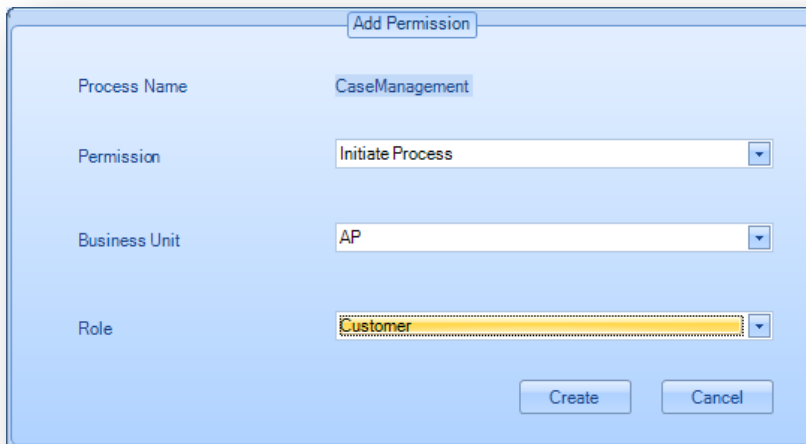


AgileXRM provides a set of SharePoint *webparts* that are intended to allow users (either CRM users or External Users) to interact with processes.

These *webparts* are configurable to show the information needed in each moment.

There are 3 *webparts*:

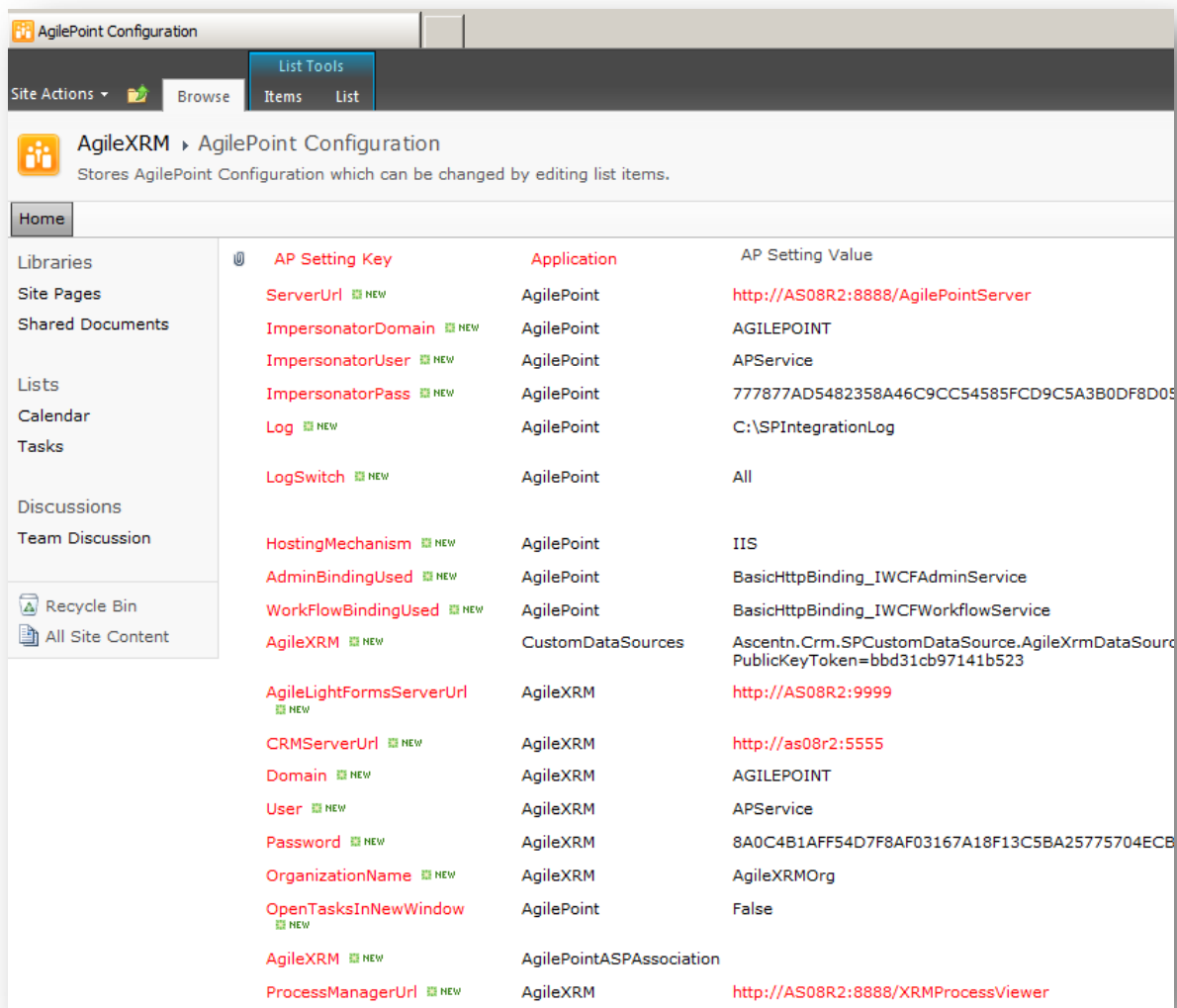
- **Process Initialization:** This is intended to show a link to the processes that the user can initiate. Permissions to initiate processes can be set in Envision at process level. This permission is assigned to CRM Roles. If the current user belongs to a role with this permission then this user can initiate the process. External Users can belong to one role because AgileXRM adds a relation between Contact and role entities, so if the external user belongs to a role with this permission the external user will be allowed to start the process.



In this sample, all contacts that belong to Customer role will be allowed to start a Case Management process.

- Process Instances: This *webpart* can show information about process instances, for example, those initiated by the current user.
- Tasks: This is intended to show the tasks that the user must interact with. Actions that the user can execute in this tasks depend on user privileges

Those WebParts get the parameters to connect to PES and XRM from a SharePoint List called *AgilePoint Configuration*. This SharePoint list is created automatically when SharePoint Solutions that contain the AgileXRM SharePoint Integration are deployed and activate to the SharePoint Site.

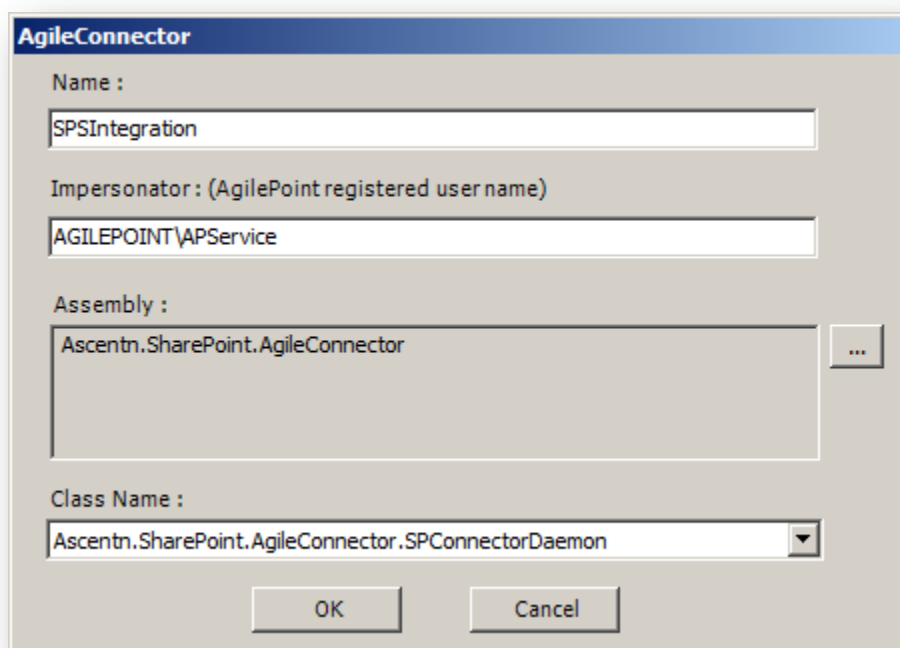
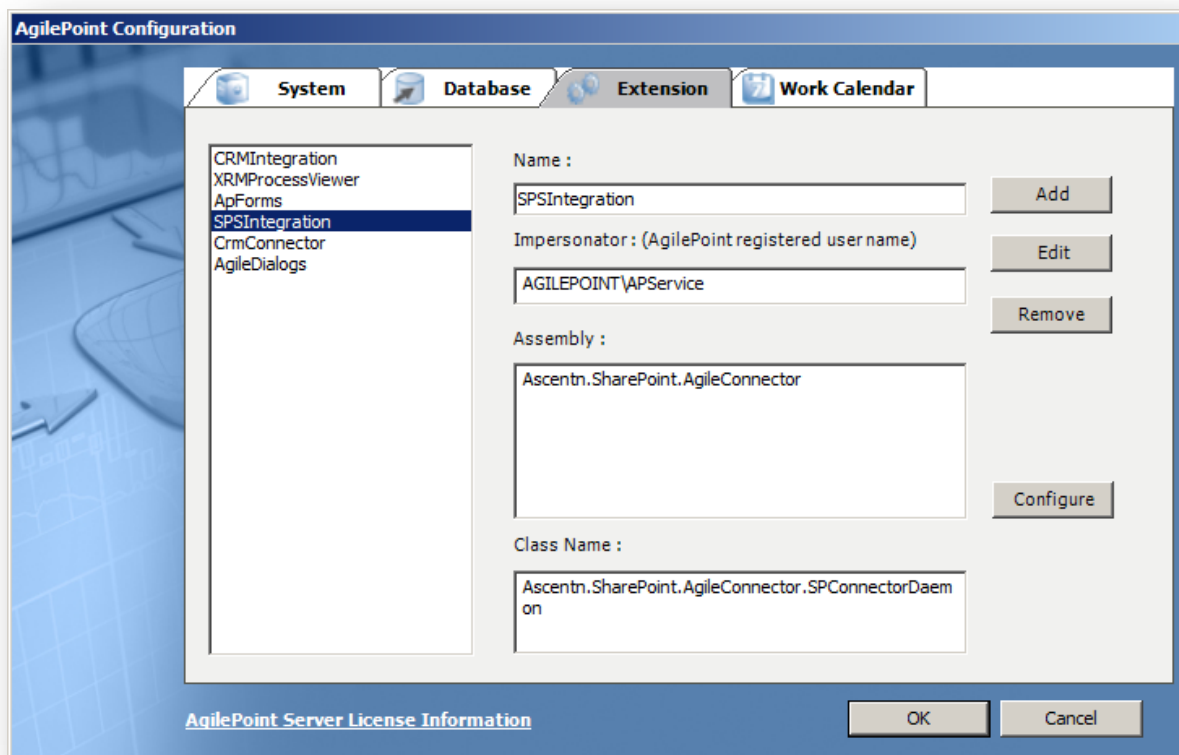


The screenshot shows the 'AgilePoint Configuration' interface. It has a top navigation bar with 'List Tools' and 'List' buttons. Below the navigation bar, there's a breadcrumb 'AgileXRM > AgilePoint Configuration' and a description: 'Stores AgilePoint Configuration which can be changed by editing list items.' A left sidebar contains navigation links like 'Libraries', 'Site Pages', 'Shared Documents', 'Lists', 'Calendar', 'Tasks', 'Discussions', 'Team Discussion', 'Recycle Bin', and 'All Site Content'. The main content area displays a table of configuration items.

AP Setting Key	Application	AP Setting Value
ServerUrl <small>NEW</small>	AgilePoint	http://AS08R2:8888/AgilePointServer
ImpersonatorDomain <small>NEW</small>	AgilePoint	AGILEPOINT
ImpersonatorUser <small>NEW</small>	AgilePoint	APService
ImpersonatorPass <small>NEW</small>	AgilePoint	777877AD5482358A46C9CC54585FCD9C5A3B0DF8D05
Log <small>NEW</small>	AgilePoint	C:\SPIntegrationLog
LogSwitch <small>NEW</small>	AgilePoint	All
HostingMechanism <small>NEW</small>	AgilePoint	IIS
AdminBindingUsed <small>NEW</small>	AgilePoint	BasicHttpBinding_IWCFAdminService
WorkFlowBindingUsed <small>NEW</small>	AgilePoint	BasicHttpBinding_IWCFWorkflowService
AgileXRM <small>NEW</small>	CustomDataSources	Ascentn.Crm.SPCustomDataSource.AgileXrmDataSource PublicKeyToken=bdd31cb97141b523
AgileLightFormsServerUrl <small>NEW</small>	AgileXRM	http://AS08R2:9999
CRMServerUrl <small>NEW</small>	AgileXRM	http://as08r2:5555
Domain <small>NEW</small>	AgileXRM	AGILEPOINT
User <small>NEW</small>	AgileXRM	APService
Password <small>NEW</small>	AgileXRM	8A0C4B1AFF54D7F8AF03167A18F13C5BA25775704ECB
OrganizationName <small>NEW</small>	AgileXRM	AgileXRMOrg
OpenTasksInNewWindow <small>NEW</small>	AgilePoint	False
AgileXRM <small>NEW</small>	AgilePointASPAssociation	
ProcessManagerUrl <small>NEW</small>	AgileXRM	http://AS08R2:8888/XRMProcessViewer

When SharePoint connects to XRM, it uses the values of *CRMServerUrl*, *Domain*, *User*, *Password* and *OrganizationName*. Note that a SharePoint site collection can only connect to a one XRM Organization. To connect to other XRM Organization, a new SharePoint site collection must be created. The new site collection can reside in same SharePoint web application or in a new SharePoint web application. The user that is configured to connect to XRM must exist on XRM Organization that SharePoint is connecting to.

When SharePoint connects to PES, it uses the values of *ServerUrl*, *ImpersonatorDomain*, *ImpersonatorUser*, *ImpersonatorPass*. SharePoint also uses those credentials to impersonate the current user on PES. This means that the ImpersonatorUser must be the impersonator for application SPSIntegration on AgilePoint Server Configuration (this user has to be an AgilePoint registered user as well):



1.8.1 SharePoint External Connector

AgileXRM provides a module named External Connector. With this module we can provide access to users that are not in the Active Directory of the organization, such as clients or suppliers, to participate in company processes.

This module is basically an ASP .NET Membership Provider that gets the user credentials from Contact entity from XRM. This Contact entity has custom fields to store user login (user name) and user password to login to SharePoint.

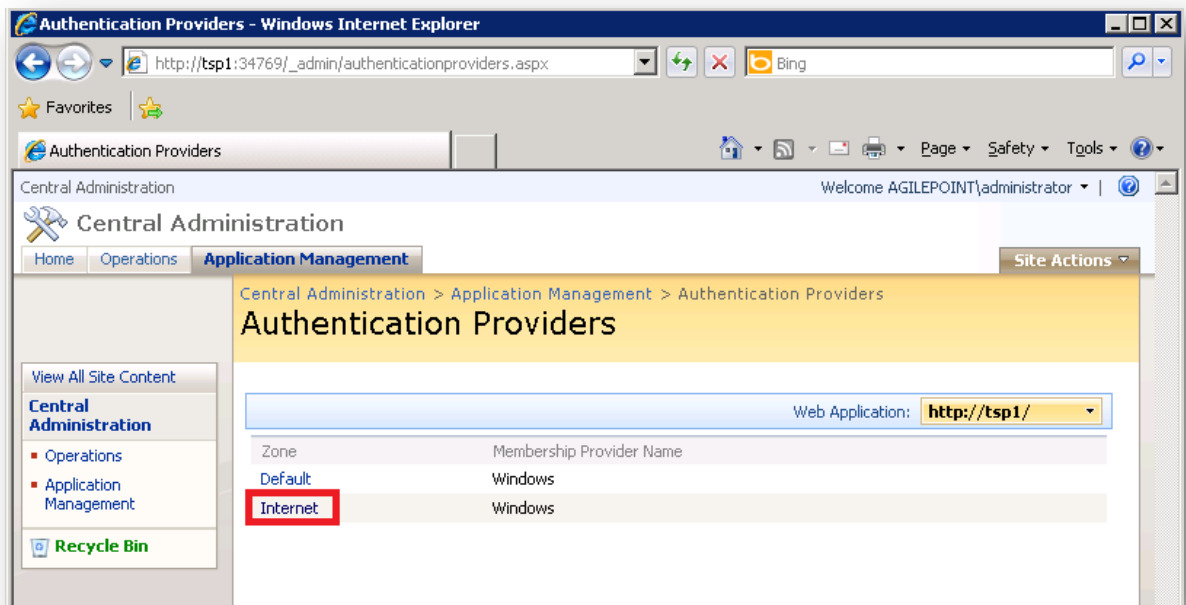
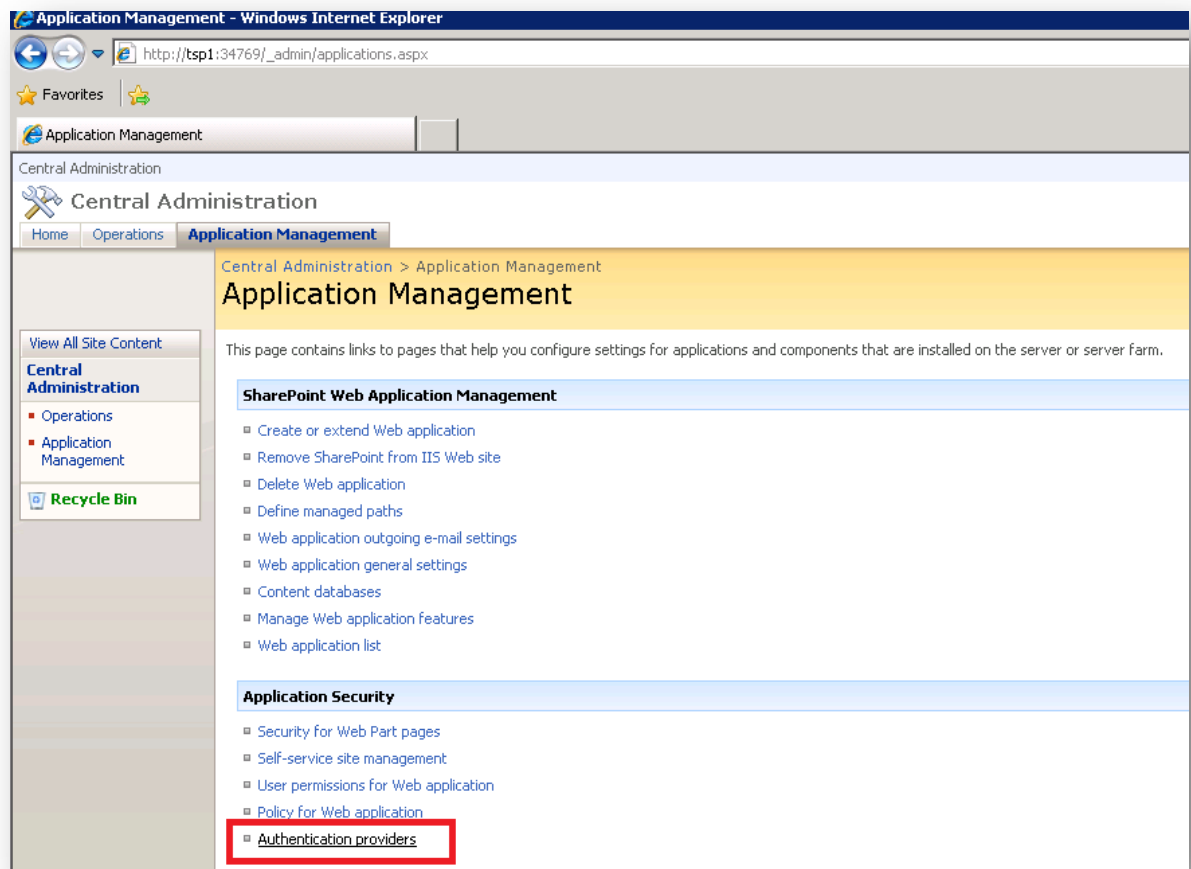
The configuration for External Connector Module in SharePoint 2010 differs from SharePoint 2007 due to new security model used in SharePoint 2010. This new security model is based on Windows Identity Framework and the use of claims to authenticate users.

1.8.1.1 External Connector for SharePoint 2007

To enable SharePoint to use AgileXRM External Connector Membership Provider, manual modifications of web.config of SharePoint Central Administration and SharePoint Web Application that will be enabled to authenticate users by AgileXRM Membership Provider are needed. The snippet below should be added inside *system.web* node:

```
<membership defaultProvider="AgileXrmMembershipProvider">
  <providers>
    <remove name="AgileXrmMembershipProvider" />
    <add connectionStringName="AgileXrmProvider"
      passwordAttemptWindow="10"
      enablePasswordRetrieval="false"
      enablePasswordReset="true"
      requiresQuestionAndAnswer="true"
      applicationName="/"
      requiresUniqueEmail="false"
      passwordFormat="Hashed"
      description="This is AgileXRM Membership provider"
      name="AgileXrmMembershipProvider"
      type="Ascentn.Crm.SPMembershipProvider.AgileXrmMembershipProvider, Ascent
n.Crm.SPMembershipProvider, Version=1.0.0.0, Culture=neutral, PublicKeyToken=bdb31cb9
7141b523" />
  </providers>
</membership>
```

Once web.config was modified, next step is to configure SharePoint Web Application to use the new Membership Provider. This is done by SharePoint Central Administration. The screens below show how to configure an existing Web Application that extends other one to have Internet zone:



The screenshot shows the 'Edit Authentication' page in the Central Administration of a SharePoint 2010 site. The page is viewed in Internet Explorer and shows configuration options for the 'AgileXrmMembershipProvider'.

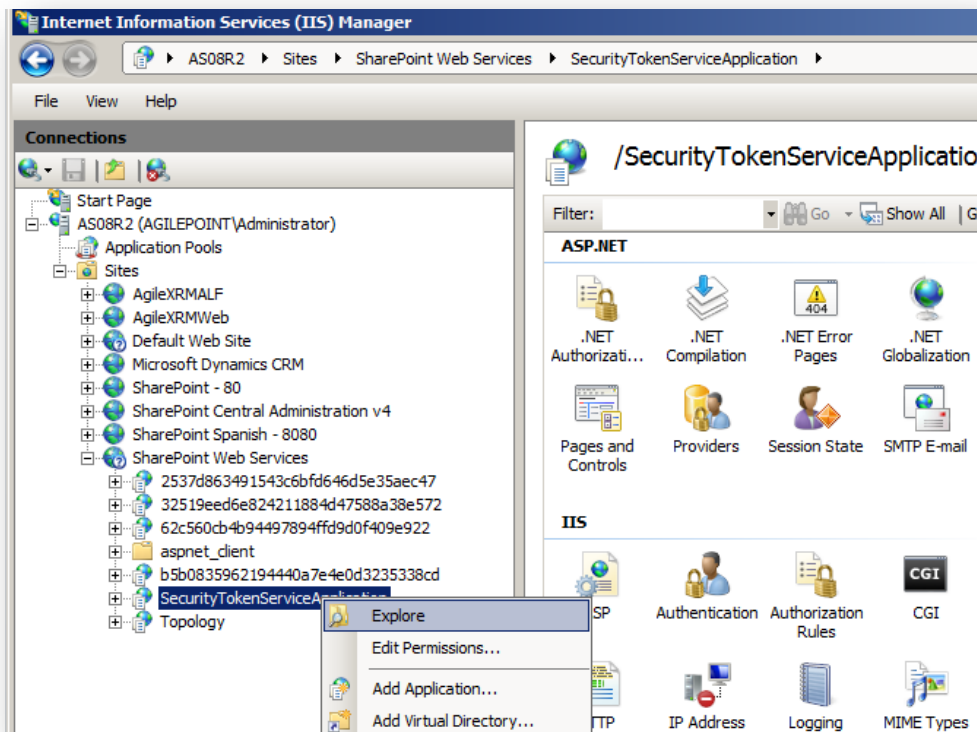
Web Application	Web Application: http://tsp1/
Zone These authentication settings are bound to the following zone.	Zone Internet
Authentication Type Choose the type of authentication you want to use for this zone. Learn about configuring authentication.	Authentication Type <input type="radio"/> Windows <input checked="" type="radio"/> Forms <input type="radio"/> Web single sign on
Anonymous Access You can enable anonymous access for sites on this server or disallow anonymous access for all sites. Enabling anonymous access allows site administrators to turn anonymous access on. Disabling anonymous access blocks anonymous users in the web.config file for this zone.	<input type="checkbox"/> Enable anonymous access
Membership Provider Name Enter the name of the membership provider. The membership provider must be correctly configured in the web.config file for the IIS Web site that hosts SharePoint content on each Web server. It must also be added to the web.config file for IIS site that hosts Central Administration.	Membership provider name: <input type="text" value="AgileXrmMembershipProvider"/>
Role Manager Name Enter the name of the role manager (optional). The role manager must be correctly configured in the web.config file for this zone.	Role manager name: <input type="text"/>
Client Integration Disabling client integration will remove features which launch client applications. See authentication providers (under Forms) for details.	Enable Client Integration? <input type="radio"/> Yes <input checked="" type="radio"/> No

Note: AgileXRM Membership Provider will get the configuration to connect to XRM and validate the user from AgilePoint Configuration List. The AgilePoint Configuration List must be configured properly before use AgileXRM Membership Provider for External Connector.

1.8.1.2 External Connector for SharePoint 2010

SharePoint 2010 introduces a new security model based on Windows Identity Framework that relies on claims to authenticate users. In this new model there is a new component called Security Token Service, which has also be aware of the AgileXRM Membership Provider, apart from SharePoint Web Application. To locate Security Token Service's web.config, open IIS

Manager Console and explore SecurityTokenServiceApplication under SharePoint Web Services web site:



Add this node to system.web node. If system.web node does not exist, add it just after system.net node:

```
<membership defaultProvider="AgileXrmMembershipProvider">
  <providers>
    <remove name="AgileXrmMembershipProvider" />
    <add connectionStringName="AgileXrmProvider"
      passwordAttemptWindow="10"
      enablePasswordRetrieval="false"
      enablePasswordReset="true"
      requiresQuestionAndAnswer="true"
      applicationName="/"
      requiresUniqueEmail="false"
      passwordFormat="Hashed"
      description="This is AgileXRM Membership provider"
      name="AgileXrmMembershipProvider"
        CRMServerUrl="http://crm:5555"
        Domain="AGILEPOINT"
        User="APService"/>
  </providers>
</membership>
```

```

Password="FA854E2762ED02CF9BE0ECCCBFC1C43E71627CB761C0D960B2B863F50BDF6B320FA82C639D6
039E0E7F0D8037AB53765"
        OrganizationName="AgileXRMEval"
        type="Ascentn.Crm.SPMembershipProvider.AgileXrmMembershipProvider,
Ascentn.Crm.SPMembershipProvider, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=bbd31cb97141b523" />
    </providers>
</membership>

```

Note that this configuration has the CRMServerUrl, Domain, User, Password and OrganizationName parameters to connect to XRM, instead of getting those parameters from AgilePoint Configuration List, because Security Token Service does not have access to SharePoint Web Application. As Password is encrypted, you can get the encrypted string from AgilePoint Configuration List on SharePoint site collection.

Create new SharePoint Web Application to use AgileXRM Membership Provider to authenticate users:

From SharePoint Central Administration, choose create new Web Application and select Claims Based Authentication:

On *Claims Authentication Types* section, leave *Enable Windows Authentication* checked to allow internal users (Active Directory) access to the new SharePoint web application using Active Directory credentials. Check the *Enable Forms Based Authentication (FBA)* and set *AgileXrmMembershipProvider* in *ASP .NET Membership provider name*:

Create New Web Application

Claims Authentication Types

Choose the type of authentication you want to use for this zone.

Negotiate (Kerberos) is the recommended security configuration to use with Windows authentication. If this option is selected and Kerberos is not configured, NTLM will be used. For Kerberos, the application pool account needs to be Network Service or an account that has been configured by the domain administrator. NTLM authentication will work with any application pool account and with the default domain configuration.

Basic authentication method passes users' credentials over a network in an unencrypted form. If you select this option, ensure that Secure Sockets Layer (SSL) is enabled.

ASP.NET membership and role provider are used to enable Forms Based Authentication (FBA) for this Web application. After you create an FBA Web application, additional configuration is required.

Trusted Identity Provider Authentication enables federated users in this Web application. This authentication is Claims token based and the user is redirected to a login form for authentication.

[Learn about configuring authentication.](#)

Create New Web Application

☒ Enable Windows Authentication

☒ Integrated Windows authentication

NTLM

☐ Basic authentication (credentials are sent in clear text)

☒ Enable Forms Based Authentication (FBA)

ASP.NET Membership provider name

AgileXrmMembershipProvider

ASP.NET Role manager name

☐ Trusted Identity provider

There are no trusted identity providers defined.

Once new SharePoint web application was created, modify its web.config to add AgileXRM Membership Provider Information, just after <add name="i"... node located inside <providers> node, located inside <membership> node:

```
<remove name="AgileXrmMembershipProvider" />
<add connectionStringName="AgileXrmProvider"
```

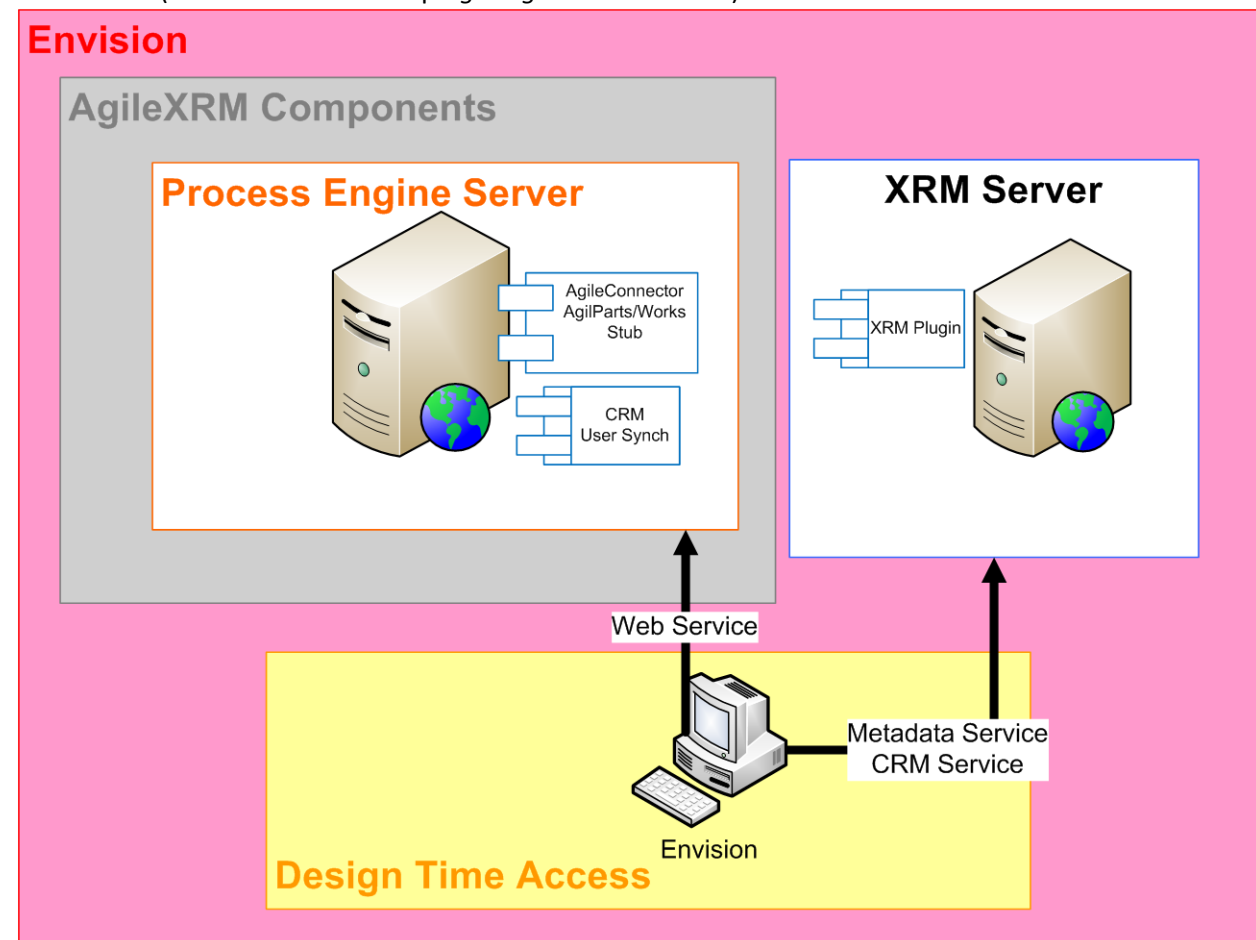
```

passwordAttemptWindow="10"
enablePasswordRetrieval="false"
enablePasswordReset="true"
requiresQuestionAndAnswer="true"
applicationName="/"
requiresUniqueEmail="false"
passwordFormat="Hashed"
description="This is AgileXRM Membership provider"
name="AgileXrmMembershipProvider"
type="Ascentn.Crm.SPMembershipProvider.AgileXrmMembershipProvider,
    Ascentn.Crm.SPMembershipProvider, Version=1.0.0.0, Culture=neutral,
    PublicKeyToken=bdb31cb97141b523" />

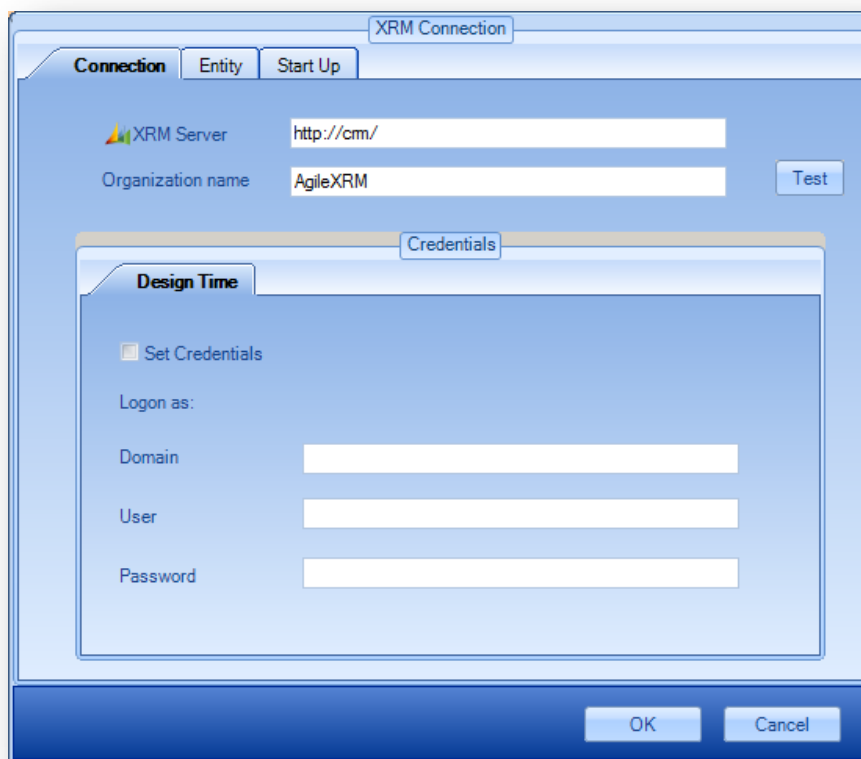
```

1.9 Access from Envision to PES and XRM Server to retrieve metadata and set process template configuration

In order to configure an AgileXRM process template Envision needs to access XRM Server to get entities metadata from CRM and some functionalities require to access CRM service to get some records information (for instance to look up *AgileLightForms* records).



In order to connect to CRM design time connection must be configured in Envision. When an AgileXRM process is opened a window is open to set design time connection configuration:



The configuration of **Design Time** tab is used only at design time to get information from CRM. This configuration does not affect runtime. The same process definition can be used in different organizations.

Here the process designer can use integrated security to connect to CRM or, checking *Set Credentials*, set another user's credentials.

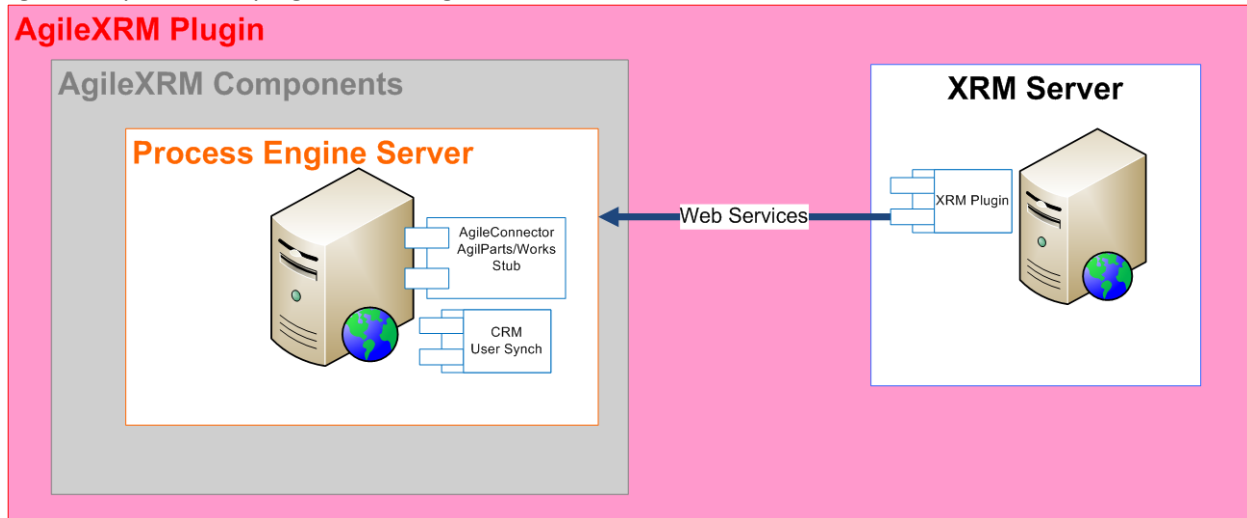
Users that are designing processes need to have access to Metadata, so, at least, they need read permissions on customizations.

If the user is going to deploy processes to PES and enable them on AgileXRM, then he needs to have permissions to deploy processes to PES and permissions to create *AgilePointProcessTemplate* records and *AgilePointTemplatePermissions* records.

If the user does not have permissions to deploy processes this deployment can be done by an Administrator.

1.10 Access to PES from XRM Server to start processes and set activity information

AgileXRM provides a plug-in to manage communication between XRM Server and PES.



This communication is done using Web Services.

This plugin is intended to be used to inform PES that an activity has been completed, reassigned, to start processes using CRM workflows,...

The configuration for this communication is stored in a custom entity *Ascentn Configuration*.

This entity is edited using AgileXRM Deployment management console.

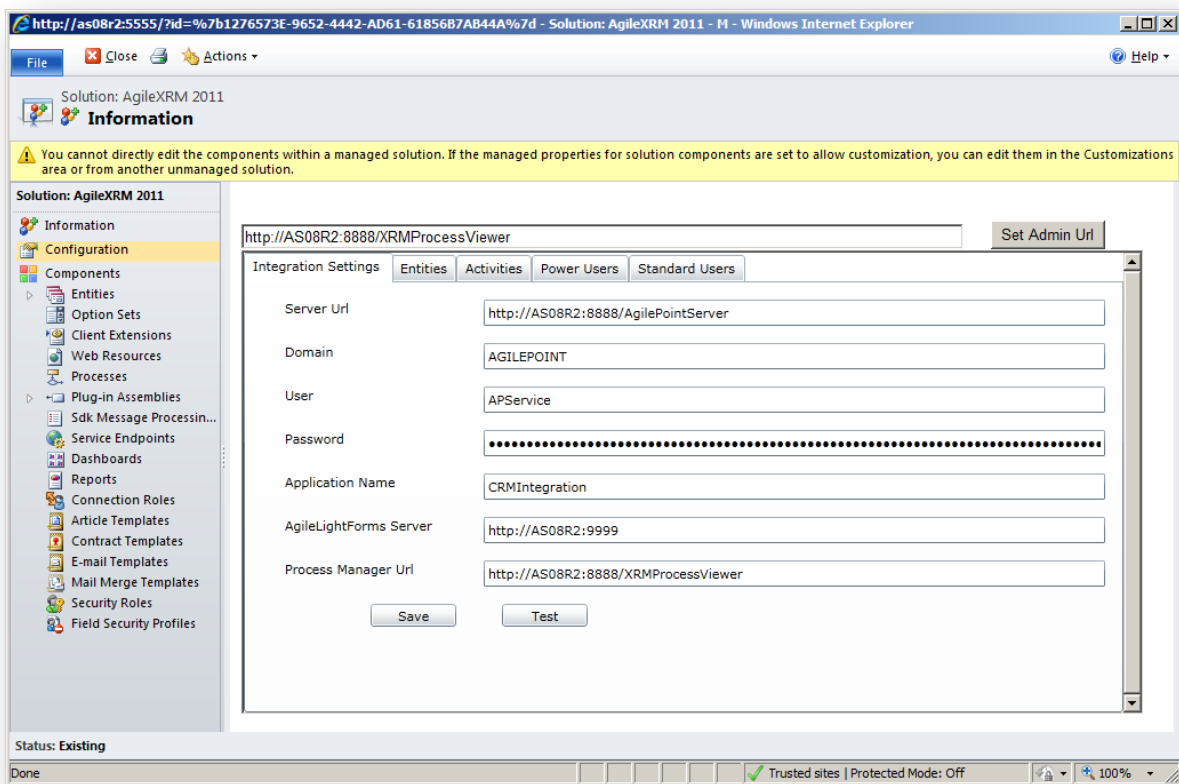
In CRM 4, for each enabled organization the administrator must configure these parameters:

The screenshot shows the 'AgilePoint Integration Configuration' dialog box. It contains the following fields and values:

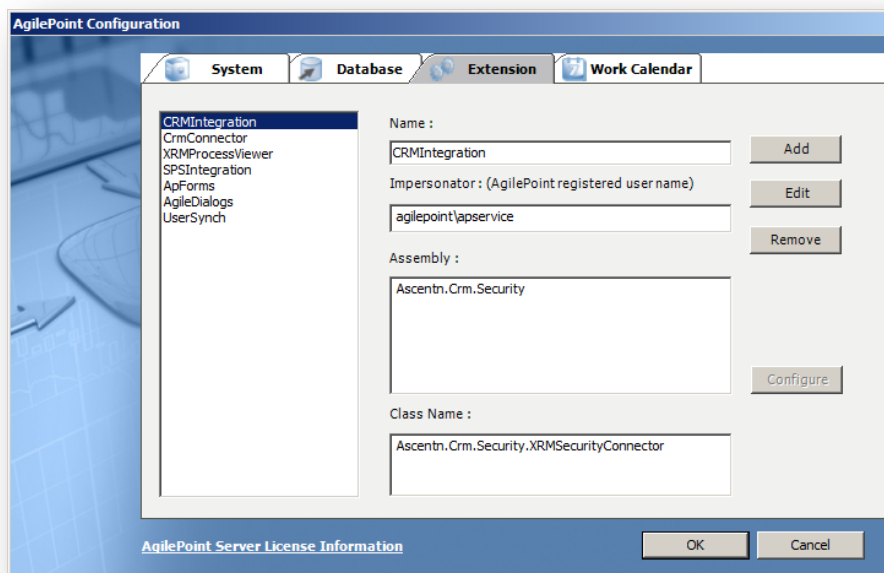
Field	Value
AgilePoint Server:	http://localhost:8888/agilepointserver
Domain:	demo
User:	apservice
Password:	••••••••
Application:	CRMIntegration
AgileLightForms Url:	http://localhost:3112
Process Viewer Url:	http://localhost:8888/SilverlightTaskList/

At the bottom of the dialog box are 'OK' and 'Cancel' buttons.

In CRM 2011, the parameters are configured in the screen below:



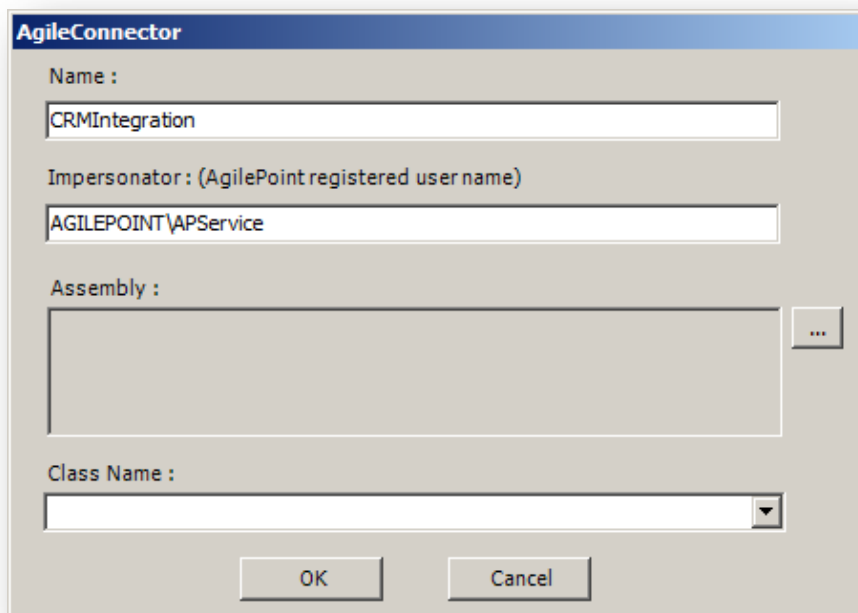
The user and password configured in this window are used by the plug-in to surrogate users in PES. To allow this surrogation, an Extension with the name of Application in this window (in this sample CRMIntegration) must be created in AgilePoint Configuration, and the field of the impersonator for this extension must be de user name configured in this window (in this sample *demo\apservice*):



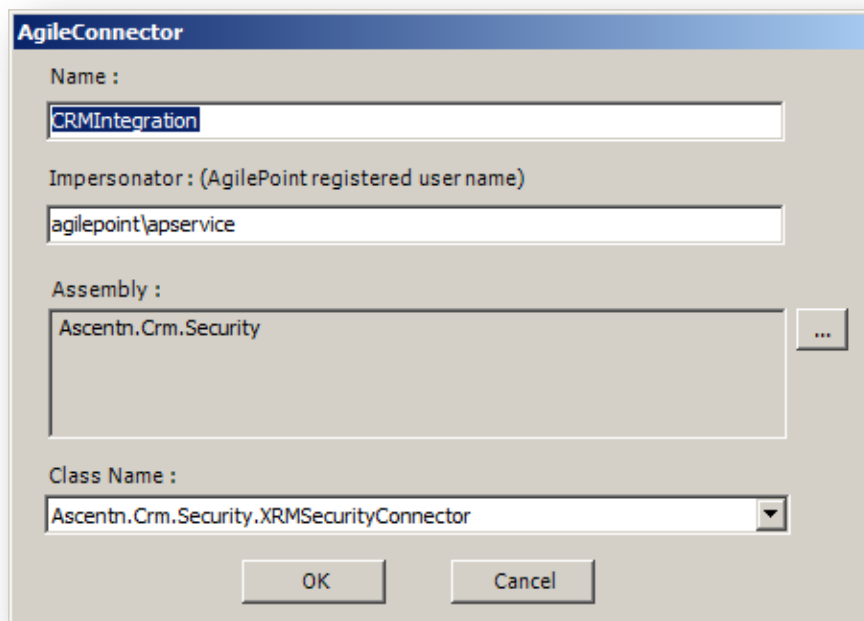
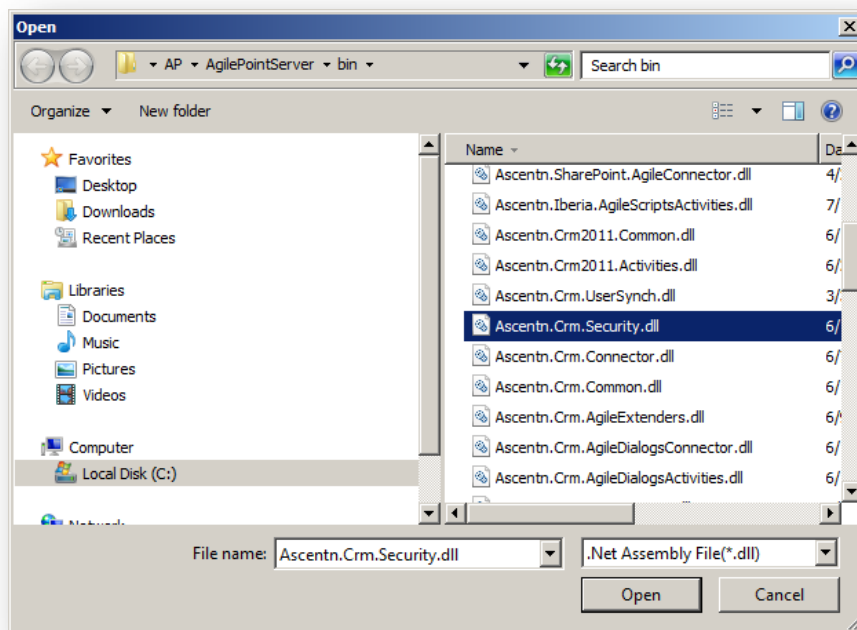
Setting this means that this user (*demo\apservice*) can impersonate users in PES. This is used, for example, to complete a task on behalf of the user that is completing the task in CRM.

This configuration can be shared between organizations or each organization can have its own impersonator configuration.

To apply runtime security this Extension must have Ascentn.Crm.Security component registered as Assembly. To do that click Add and in this window:



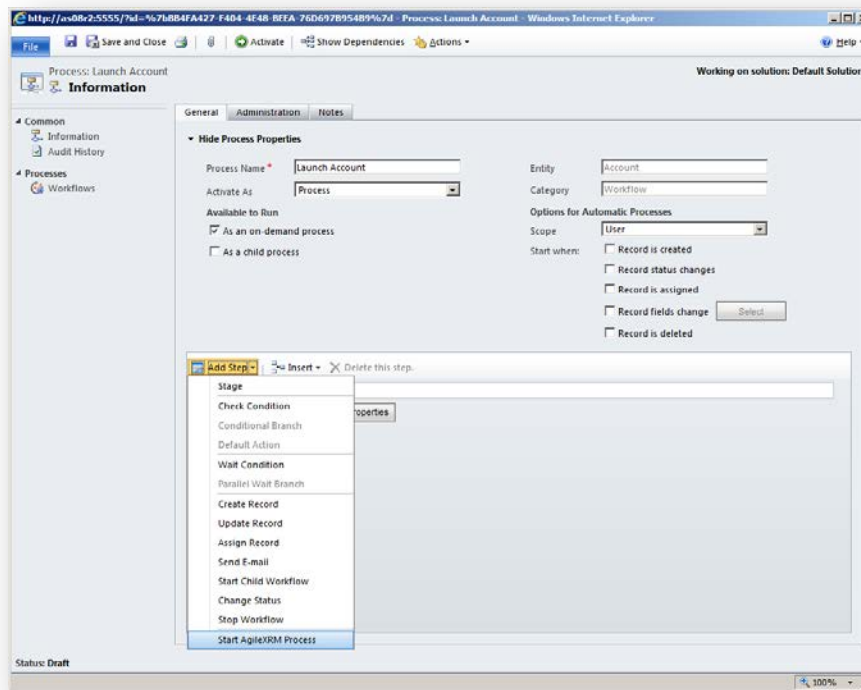
Click the ellipsis button to select *Ascentn.Crm.Security.dll*:



Click OK to save configuration.

Launching processes using AgileXRM CRM workflow activity

AgileXRM provides an activity to launch processes using a CRM workflow.



When this activity is used the AgileXRM workflow activity launches the process surrogating the user provided by the CRM context. If the workflow is launched on demand, the user provided by the context is the user that launched the workflow. When the workflow is automatic, the user is the owner of the workflow.

When this activity is used, the user that the CRM context provides must have permissions to start processes of the corresponding template.

2. AgileLightForms Security

In AgileLightForms (ALF) architecture, users connect to ALF Server (through CRM or SharePoint), and ALF Server connect to CRM to retrieve forms, and any data source specified in form connections. Every data source manager can implement its own security rules to access data.

3. Connections from Client Browser to AgileLightForms Server

AgileLightForms user interface, either for designing forms or for using them, is implemented in Silverlight that, in turn, is hosted in an ASP .NET Site (AgileLightForms Server), but end users do not directly connect to ALF Server.

3.1 Form design

To design forms, users connect to CRM and edit AgileLightForms entities (*ascent_agilelightforms*). The form for this entity in CRM embeds the ALF Form Designer.

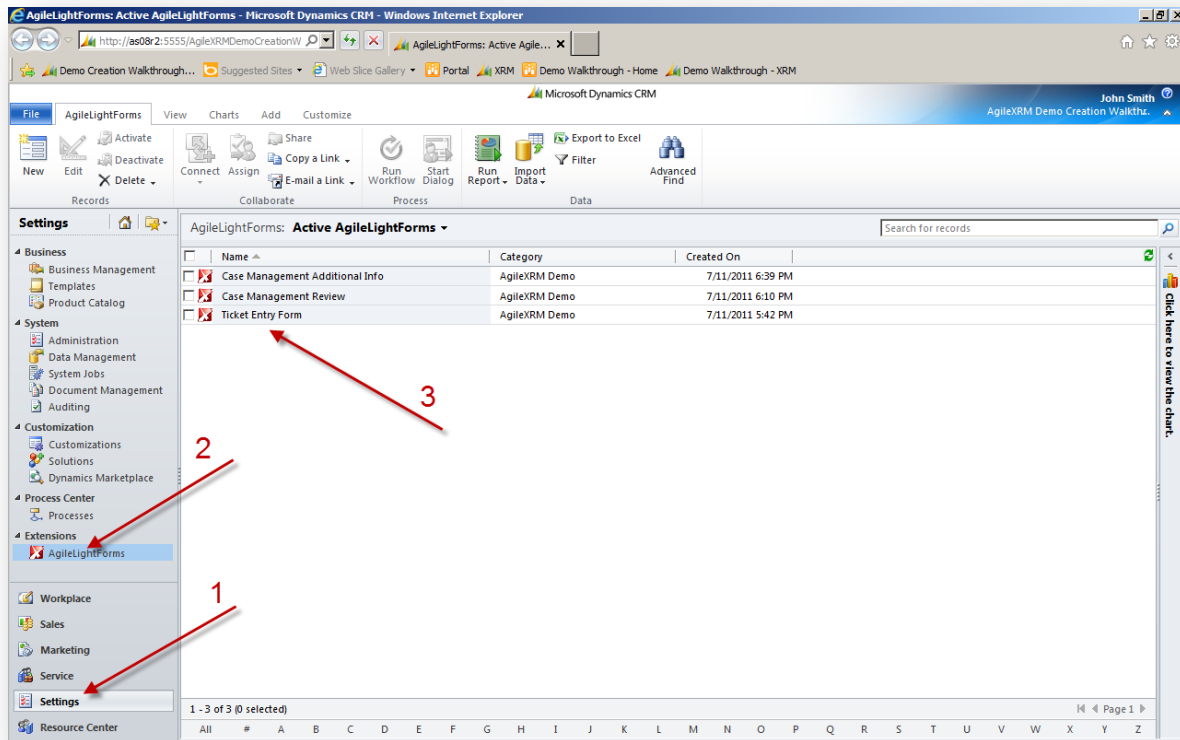


Figure 1 - Opening AgileLightForms Designer

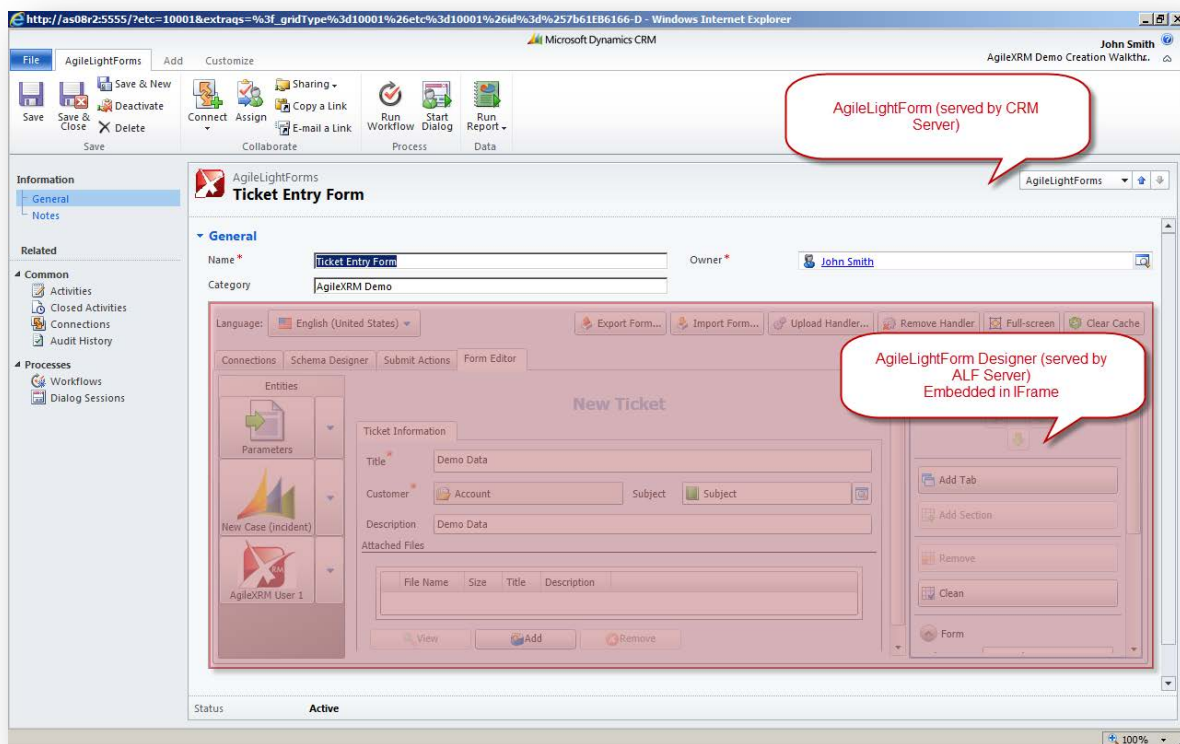


Figure 2 - AgileLightForms Designer

In order for a user to be able to design a form, that user must be able to open the form in CRM, so form designers have to be CRM users that have permissions to manage AgileLightForm entities.

3.2 Form usage

Form usage is different depending on whether forms are stand-alone forms or are embedded in CRM activity forms.

3.2.1 CRM Forms

End users use CRM Activity (Task, Phone Call, Fax...) forms by opening the associated CRM Activity Form directly in CRM.

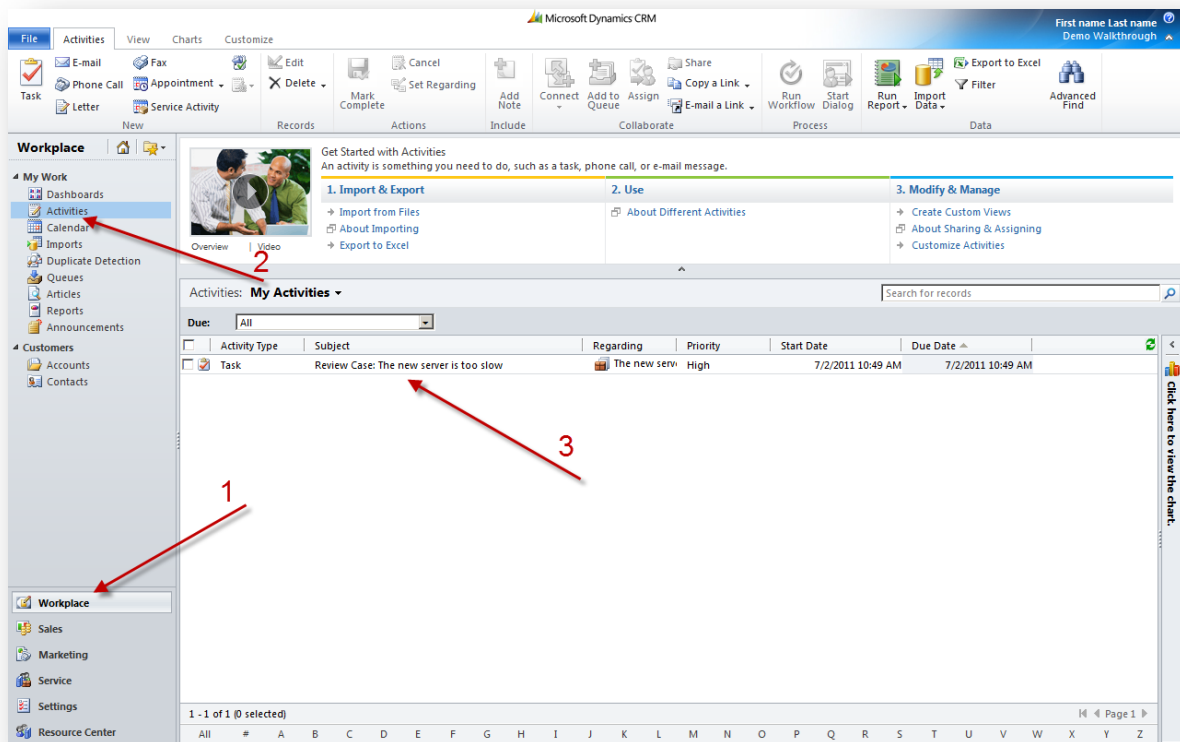


Figure 3 - Opening a CRM Activity Form

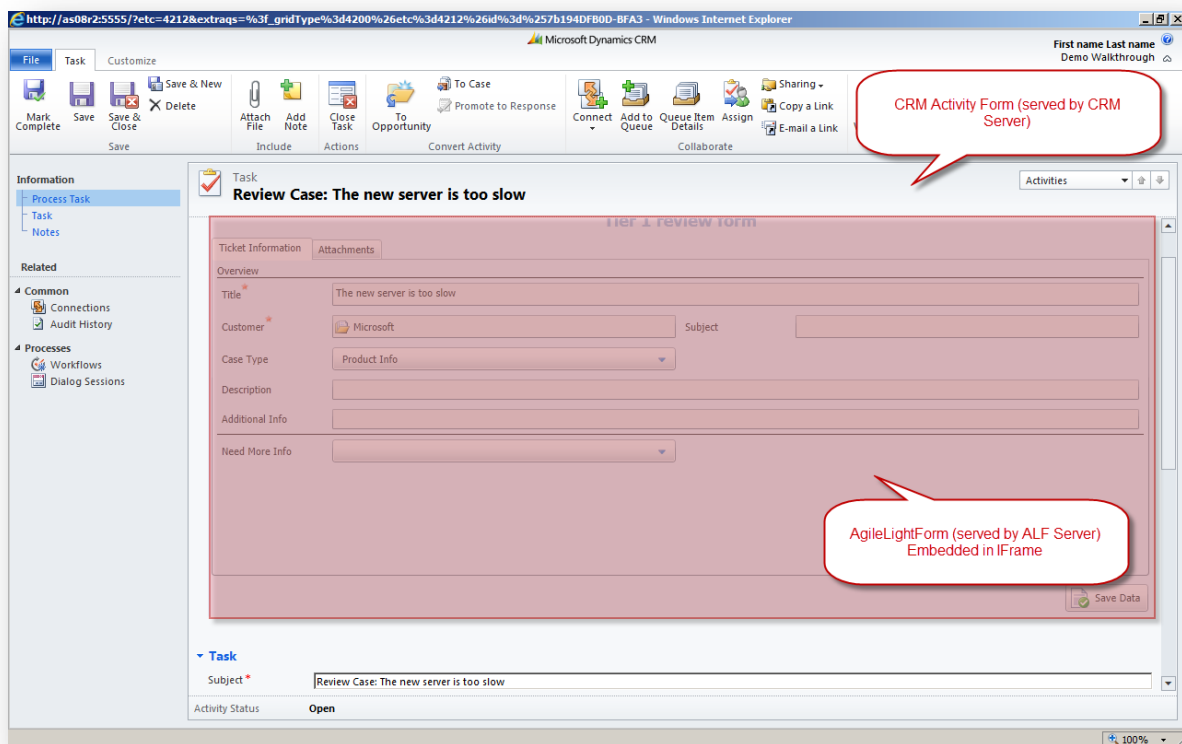


Figure 4 - AgileLightForm embedded in a CRM Activity Form

For a user to be able to use an AgileLightForm embedded in a CRM activity, that user has to have permissions to open that activity, so this kind of forms can only be used by CRM users. This makes sense, given that these forms are embedded inside CRM forms.

3.2.2 External Forms

For external task forms (forms to either launch a new process or complete a manual task), end users navigate to forms using the AgileXRM WebParts in SharePoint. These WebParts redirect the user to the appropriate page in ALF Server.

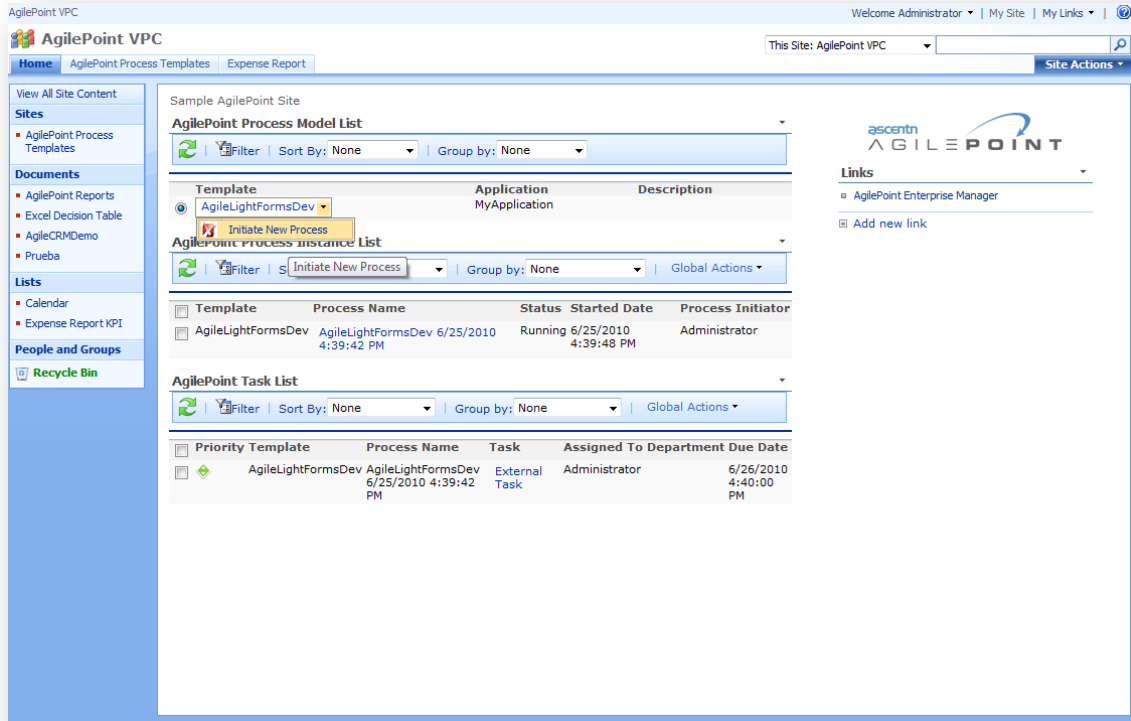


Figure 5 - Opening a Start-up AgileLightForm from an AgileXRM WebPart in SharePoint

The screenshot displays the AgilePoint VPC web application interface. The top navigation bar includes 'Home', 'AgilePoint Process Templates', and 'Expense Report'. The left sidebar contains sections for 'Sites', 'Documents', 'Lists', and 'People and Groups'. The main content area is titled 'Sample AgilePoint Site' and contains three lists: 'AgilePoint Process Model List', 'AgilePoint Process Instance List', and 'AgilePoint Task List'. The 'AgilePoint Task List' is currently selected, showing a table with columns: Priority, Template, Process Name, Task, Assigned To, Department, and Due Date. A context menu is open over the 'External Task' in the first row, listing actions: View Process, Cancel Task, Cancel Process, Reassign, Task Rework, Create Linked Work Item, View Description, and Open External. The 'Open External' option is highlighted. The right sidebar features the 'ascentn AGILE POINT' logo and a 'Links' section with 'AgilePoint Enterprise Manager' and 'Add new link'.

Priority	Template	Process Name	Task	Assigned To	Department	Due Date
	AgileLightFormsDev	AgileLightFormsDev 6/25/2010 4:39:42 PM	External Task	Administrator		6/26/2010 4:40:00 PM

Figure 6 - Opening an External Task AgileLightForm from an AgileXRM WebPart in SharePoint

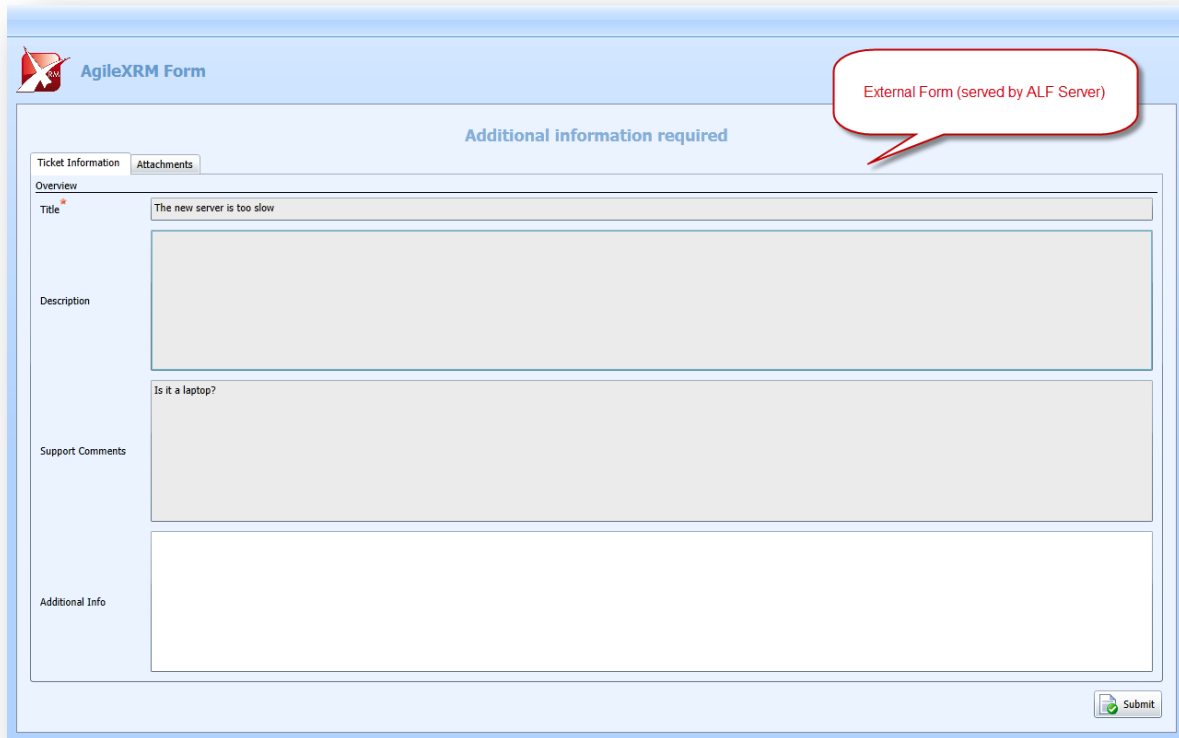


Figure 7 - External AgileLightForm opened from an AgileXRM WebPart in SharePoint

Forms that are opened from AgileXRM WebParts in SharePoint can be opened by any AgileXRM user that has successfully logged in to the AgileXRM site.

4. Connections to CRM Server from AgileLightForms Server

The account used for the ALF Application Pool in IIS must be a CRM user that is a member of the PrivUserGroup Active Directory User Group, so it can impersonate other CRM users.

That account has to also be able to read AgileLightForms entities (ascent_agilelightforms).

Apart from those two basic permissions, if forms are configured to use System Permissions (see below), this account has to be able to read and write entities as needed by those forms.

4.1 Form Storage

Forms are always read using the credentials of the account used in the ALF Application Pool. That is why this account has to have permissions to read AgileLightForms. The reason for this is that forms have to be read even for external users, so no impersonation can take place.

Also, this saves the burden of granting read permissions to forms to all potential form users, and also avoids their reading form templates, which could impose a security risk.

On the other hand, when designing forms, ALF impersonates the form designer, that has to be a CRM user with permissions to create and update forms.

The rest of connections to CRM, either in design-time or in run-time, will be done using a CRM Connection object in the form template.

4.2 Connections with credentials

If the CRM connection in the form template has credentials (at least a user name is specified), those

credentials are used both in design-time and in run-time.

4.3 Design-time connections without credentials

If a connection to CRM related to a CRM connection without credentials is done in design-time, ALF impersonates the form designer. So, Form designers must have access to metadata. Also, if a Lookup window is opened to search for a default value for an attribute, only entities visible to form designer will be retrieved.

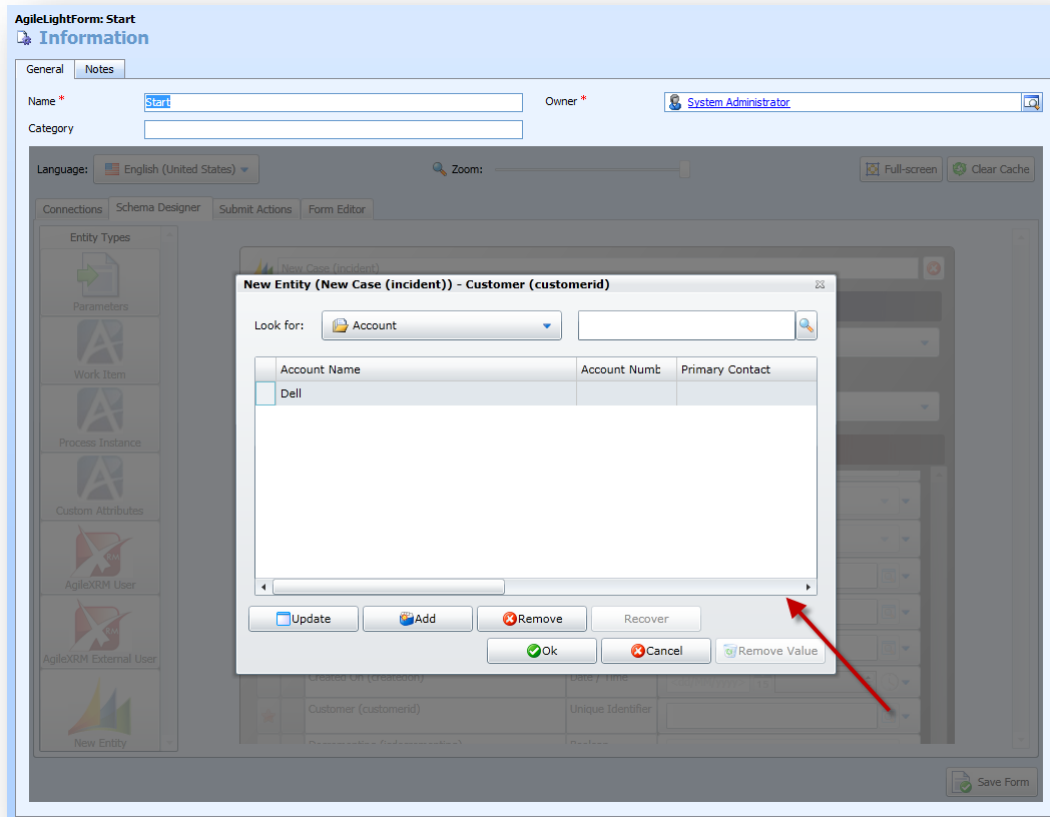


Figure 8 - Lookup Window opened in Design-time to set a Default Value for an Attribute

4.4 Run-time connections without credentials

In run-time, when a connection to a CRM server has to be established, the credentials used for that connection depend on the Permissions configured for the form, and also the Permissions configured for the process template.

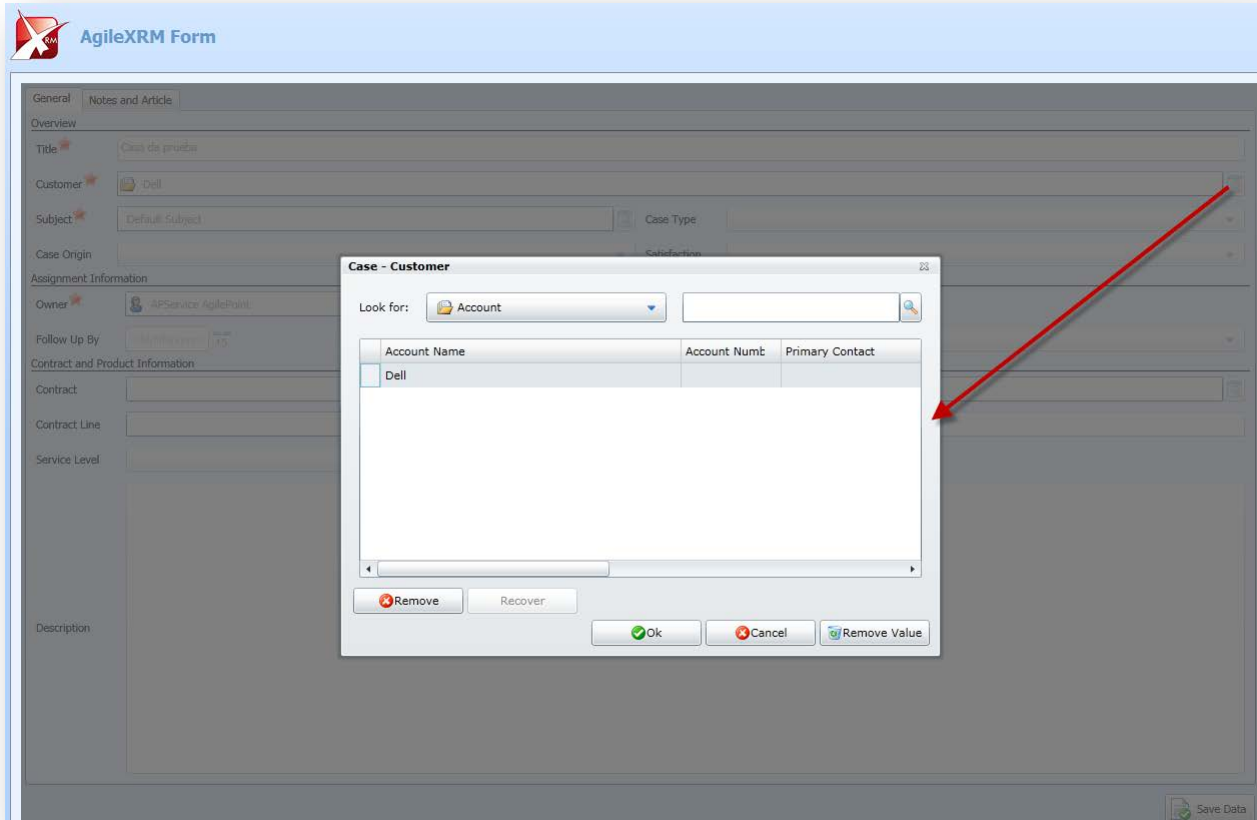


Figure 9 - Lookup Window opened in run-time (only display names are shown on title) to set an Attribute Value

4.4.1 Connections with Process Template Permissions

When Form Permissions are set to “Process Template”, Process Template Permissions are used. Process Template Permissions can be “System”, “Owner” or “Process Initiator”, and form will behave as if “System”, “Process Template Owner”, or “Process Initiator” Permissions had been chosen directly in form template (see below).

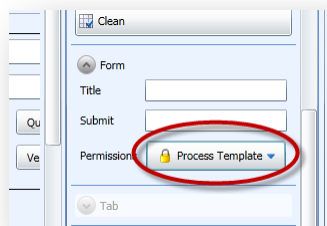


Figure 10 - Form configured to run using Process Template Permissions

4.4.2 Connections with System Permissions

When Form Permissions are set to “System” (or Form Permissions are “Process Template” and Process Permissions are “System”), no impersonation is done. That is, all access to CRM will be done using the credentials of the account of the ALF Application Pool. If this mode is used, that account should be granted all necessary permissions.

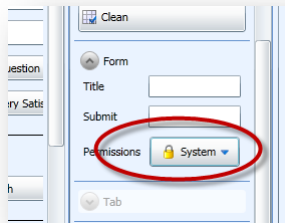


Figure 11 - Form configured to run using System Permissions

4.4.3 Connections with Form User Permissions

When Form Permissions are set to “Form User”, the user of the form is impersonated. This restricts the use of that form to CRM users, so it should be used in CRM Activity Forms only. This is the only case in which different users can see different data when accessing the same task (in this case, the same CRM Activity).

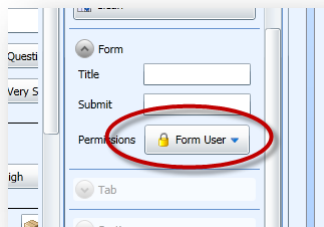


Figure 12 - Form configured to run using Form User Permissions

4.4.4 Connections with Process Template Owner Permissions

When Form Permissions are set to “Process Template Owner” (or Form Permissions are “Process Template” and Process Permissions are “Owner”), ALF impersonates the process template owner, i.e., the owner of the AgilePointProcessTemplate (ascentn_agilepointprocesstemplate) entity that represents the process template.

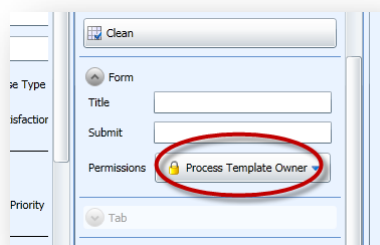


Figure 13 - Form configured to run using Process Template Owner Permissions

4.4.5 Connections with Process Initiator Permissions

When Form Permissions are set to “Process Initiator” (or Form Permissions are “Process Template” and Process Permissions are “Process Initiator”), ALF impersonates the process instance initiator. Special care should be taken to make sure that forms configured with “Initiator” permissions are initiated by CRM users. Otherwise, they will fail if they access CRM.

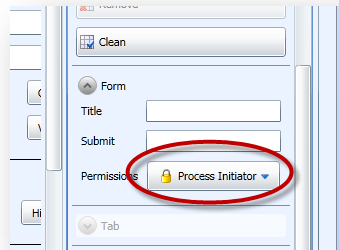


Figure 14 - Form configured to run using Process Initiator Permissions

5. Connections to AgilePoint Server from AgileLightForms Server

Connections to AgilePoint from ALF Server are always related to an AgilePoint Connection in a Form Template.

5.1 AgileLightForms Connector

AgileLightForms relies on an AgilePoint connector to enforce security in its connections to that server. As all AgilePoint connectors, AgileLightForms connector can be found in tab Extension of AgilePoint Server Configuration application.

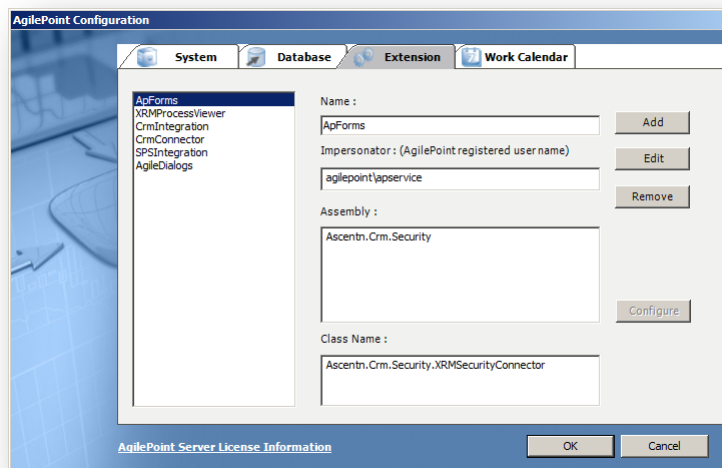


Figure 15 – AgileLightForms Connector in AgilePoint Server Configuration

The name of the connector is ApForms, and the impersonator must be the account of the AgileLightForms Server Application Pool. This way, AgileLightForms Server will be allowed to impersonate other AgilePoint users.

The connector itself is in the Ascentn.Crm.Security assembly, and has no configuration parameters.

5.2 Connections with credentials

If the AgilePoint connection in Form Template contains credentials (that is, contains at least a user name), those credentials are used for all connections to AgilePoint.

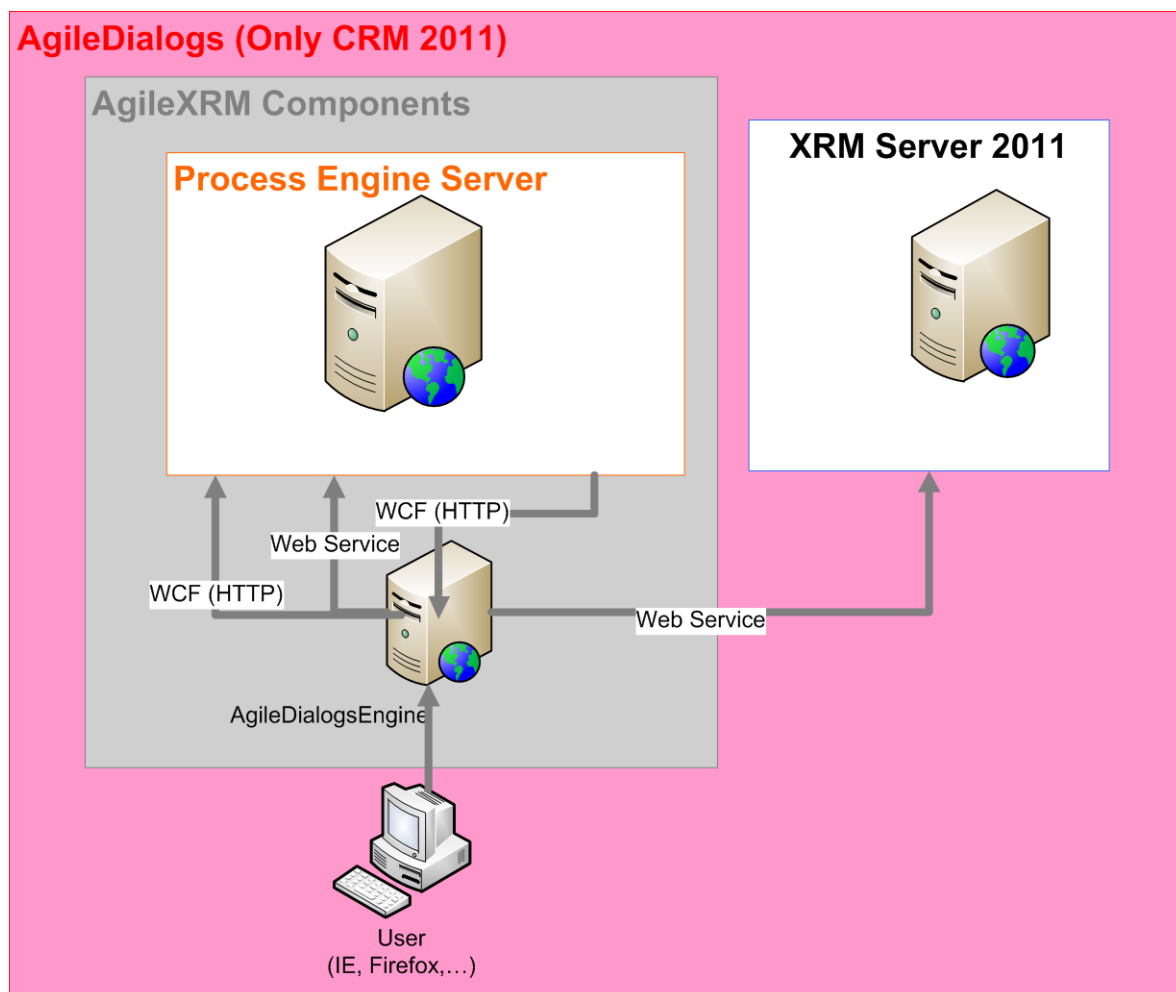
5.3 Connections without credentials

Most actions done through AgilePoint connections without credentials are executed under the permissions of the account of the ALF Application Pool. So, this account has to be a valid AgilePoint user. On the other hand, the Complete Work Item Submit Action impersonates the user that is trying to complete the work item (that has to match the user the work item is assigned to).

6. AgileDialogs Security (only for CRM 2011)

There are several aspects of communications to take into account for AgileDialogs:

- Access from AgileDialogs Engine to PES.
- Access from AgileDialogs Engine to XRM Server.
- Access from PES to AgileDialogs Engine.
- Access from Users navigator (IE, Chrome, Safari or Firefox).



6.1 Access from AgileDialogs Engine to PES

There are two modes of access from AgileDialogs Engine to PES:

- Web Service.
- WCF using HTTP binding.

6.1.1 Web Service Access

AgileDialogs connects to PES using web services to get information about processes, tasks,... This connection is done using the credentials of the application pool where AgileDialogs Engine is installed.

6.1.2 WCF using HTTP Binding

This method of access is used to start dialogs, complete steps or pages and do rollbacks in AgileDialogs. AgileDialogs Engine acts a subscriber in a publisher-subscriber pattern where PES is the publisher. The configuration for this communication is in *web.config* files in PES application and in AgileDialogs application.

This WCF configuration should not be changed unless specific behaviors (due to firewall rules, for instance) are needed.

AgileDialogs Engine calls a service in this address:

Error! Hyperlink reference not valid.

This service is configured to use HTTP binding with NTLM Authentication Scheme.

6.2 Access from PES to AgileDialogs Engine.

PES acts as a publisher of events related to AgileDialogs processes, when an event that affects the Dialogs Engine is raised, PES calls the corresponding AgileDialogs Engine to notify the event.

This call uses a service exposed by AgileDialogs Engine in this URL:

<http://<YourServer>/AgileDialogs/NotificationReceiver/NotificationReceiver.svc>

This call uses HTTP binding with Anonymous authentication scheme.

6.3 Access from AgileDialogs Engine to XRM Server.

Several AgileDialogs features require retrieving data from XRM repository to be used in dialogs.

This data is retrieved using web services calls to XRM Server.

The credentials used to connect to XRM Server are the credentials configured in the process template: Owner, System or Process Initiator. The behavior is the same that for AgileXRM processes. The idea is that connections to CRM to retrieve options for combos, records for grids, ... are made on behalf of the owner of the process, the dialog initiator or the system account (in this case the user of the application pool in AgileDialogs).