



Process Platform Frontend Developers Guide

Owner: Resultmaker Research & Development
Version: 6.0.3
Revision: 2013-09-25

Lars Pedersen, Charles Symons

Contents

1	Context and Definition	8
2	Configuration.....	8
2.1	Resultmaker.OC.Frontend.config.....	8
2.2	Web.config	16
3	Client Scripts.....	18
3.1	Inter-frame communication	18
3.2	Class: RM_ShowHide.....	18
3.2.1	Declaring and Using	18
3.2.2	Properties.....	18
3.2.3	Methods	18
3.2.4	Modes	21
3.2.5	Correct order of modes.....	22
3.3	Navigation buttons ("Back"/"Continue") functions	23
3.3.1	RM_RenameBackButton	23
3.3.2	RM_RenameContinueButton	23
3.3.3	RM_SetHidingBackButton	23
3.3.4	RM_HideBackButton	24
3.3.5	RM_ShowBackButtonWeak	24
3.3.6	RM_ShowBackButtonStrong.....	24
3.3.7	RM_SetHidingContinueButton.....	24
3.3.8	RM_HideContinueButton	25
3.3.9	RM_ShowContinueButtonWeak	25
3.3.10	RM_ShowContinueButtonStrong	25
3.3.11	RM_SetDisablingBackButton	25
3.3.12	RM_DisableBackButton	26
3.3.13	RM_EnableBackButtonWeak	26
3.3.14	RM_EnableBackButtonStrong	26
3.3.15	RM_SetDisablingContinueButton	27
3.3.16	RM_DisableContinueButton	27
3.3.17	RM_EnableContinueButtonWeak.....	27
3.3.18	RM_EnableContinueButtonStrong	27
3.3.19	RM_EnableContinueButton	28
3.3.20	RM_ShowContinueButton	28
3.3.21	RM_ResetBackButton	28
3.3.22	RM_ResetContinueButton	28

3.4	Collections.....	29
3.4.1	RM_Addon_Params	29
3.5	Functions.....	29
3.5.1	Some examples of most used functions	29
3.5.2	getquestion	30
3.5.3	getdescription (obsolete)	30
3.5.4	setdescription.....	30
3.5.5	getquestionname	31
3.5.6	getanswer.....	31
3.5.7	getdate	31
3.5.8	displaydate.....	32
3.5.9	RM_getnumber	32
3.5.10	RM_displaynumber.....	33
3.5.11	getselectedname	33
3.5.12	RM_ActivatePDFButton	33
3.5.13	RM_GetProjectID.....	33
3.5.14	getquestiontextobj	34
3.5.15	RM_ActivateContinueButton.....	34
3.5.16	RM_OpenLink	35
3.5.17	RM_MoveNext(0)	35
3.5.18	RM_GetQuestionType	35
3.5.19	RM_OpenCodePicker.....	36
3.5.20	RM_OpenFileStore.....	36
3.5.21	RM_Validate	36
3.5.22	RM_GoToActivity	36
3.5.23	RM_FormatNumber.....	37
3.5.24	GetShowHideObject	37
3.5.25	RM_ShowHideIconByElem	38
3.5.26	RM_ShowHideIconByName	38
3.5.27	getquestiontext	38
3.5.28	RM_calcnnumber.....	39
3.5.29	RM_AttachOnLoadEvent	39
3.5.30	RM_ConditionalAlert	40
3.5.31	RM_ShowHideElement	41
3.5.32	RM_GetShowHideObjectFromElement	41
3.6	Events.....	43

3.6.1	onclick	43
3.6.2	onchange.....	43
3.6.3	RM_CustomValidation()	43
3.7	Constants	43
3.7.1	RM_oneday	43
3.7.2	RM_oneweek	44
3.7.3	RM_oneyear.....	44
3.8	Debug Client Script.....	44
3.8.1	Debug Client Script in Microsoft Visual Studio 2005.....	44
3.8.2	Debug Client Script in Internet Explorer with Microsoft Script Debugger	44
3.9	Query string API	44
3.9.1	ProjectTypeFilter	44
3.9.2	Destination	45
3.9.3	StartProject	45
3.9.4	Id	45
3.9.5	SkinName	46
3.9.6	ResumeProject	46
3.9.7	DefaultProjectListType	47
3.9.8	ProjectListType.....	47
3.9.9	ProjectListSortNew.....	47
3.9.10	ProjectListSortRunning	47
3.9.11	ProjectListSortCompleted	47
3.9.12	ProjectListSortPending	48
3.9.13	ProjectListSortBy.....	48
3.9.14	ProjectListItemsPerPage	48
3.9.15	RM_EnableEnrichedWorkflow.....	48
3.9.16	RM_WorkflowOnly	48
3.9.17	ResumeCurrent.....	48
3.9.18	ResumeFromInvitation	49
3.9.19	GoToActivity	49
3.9.20	RedirectToSenderURL.....	50
3.9.21	RedirectToSenderText	50
3.9.22	ShowLogin.....	50
3.9.23	ShowLoginIf	50
4	Process Frontend User Object Documentation.....	50
4.1	User object interface.....	51

5	Project Properties	51
5.1	Project Level	51
5.1.1	DefaultPurposes	51
5.1.2	Purpose rule	52
6	Project list layout	52
6.1	Running and Completed projects	57
7	Custom Properties	58
7.1	Project Level	59
7.1.1	projectlistvariable	59
7.1.2	showphases	62
7.1.3	showactivitysets	62
7.1.4	menulabel	63
7.1.5	waittext	63
7.1.6	exitlink	63
7.1.7	projecttype category	64
7.1.8	projectstartpage	66
7.1.9	projectendpage	66
7.1.10	projectheader	66
7.1.11	relativenavigation	67
7.1.12	backbuttontext	67
7.1.13	continuebuttontext	68
7.2	Phase Level	68
7.2.1	showphase	68
7.3	ActivitySet Level	69
7.3.1	showactivityset	69
7.3.2	Predecessor	69
7.4	Activity Level	70
7.4.1	waittext	70
7.5	Questionnaire Level	70
7.5.1	helpid	70
7.5.2	exitlink	70
7.5.3	backbuttontext	71
7.5.4	continuebuttontext	71
7.6	Page Level	72
7.6.1	waittext	72
7.6.2	exitlink	72

7.6.3	script.....	72
7.6.4	buildtable	73
7.6.5	backbuttontext.....	76
7.6.6	backbuttonhidden.....	76
7.6.7	backbuttondisabled	76
7.6.8	continuebuttontext.....	77
7.6.9	continuebuttonhidden	77
7.6.10	continuebuttondisabled	77
7.7	Question Level.....	78
7.7.1	placeholder text	78
7.7.2	oioexportname.....	78
7.7.3	transactionidquestionname	78
7.7.4	signexportname	79
7.7.5	signenableskip.....	79
7.7.6	signcollectionidquestionname	79
7.7.7	applettypequestionname	79
7.7.8	forceapplettypequestionname	80
7.7.9	exportname.....	80
7.7.10	attributes	80
7.7.11	displaytextattributes.....	80
7.7.12	default.....	81
7.7.13	required	81
7.7.14	maxlength	82
7.7.15	hidden	82
7.7.16	displayonly	82
7.7.17	taborder.....	83
7.7.18	focus	83
7.7.19	applettype.....	83
7.7.20	url.....	84
7.7.21	openmode.....	84
7.7.22	linktext	84
7.7.23	applicationname	85
7.7.24	mode.....	85
7.7.25	width.....	86
7.7.26	height.....	86
7.7.27	icon	86

7.7.28	positionelement.....	87
7.7.29	mapping.....	87
7.8	Answer Level	88
7.8.1	attributes.....	88
7.8.2	displaytextattributes	88
7.8.3	taborder	88
7.9	Flash custom properties.....	89
7.9.1	hideMenuOnFirstPage	89
7.9.2	flashTheme.....	89
7.9.3	foldInOutSpeedExternal	90
7.9.4	menuScrollSpeedExternal	90

1 Context and Definition

This document is a guide for developers working with the Process Platform Frontend. It is written for third-party developers as well as in house developers that have the task of working with the system.

2 Configuration

This section describes the configuration settings in the Process Frontend file Resultmaker.OC.FrontEnd.config and the web.config

2.1 Resultmaker.OC.Frontend.config

CONFIGURATION SETTING NAME	DESCRIPTION	EXAMPLE VALUES
Section: Resultmaker.OC.FrontEnd		
Language	This setting determines the Frontend user interface language for the fixed texts.	English, Danish
Culture	This setting determines the Frontend culture, that controls the presentation of the date time and numeric elements. Available from version 6.0	da-DK, en-US
FrontEndIdentifier	Identifier of the FrontEnd instance (used e.g. as a comment in HTML source and in error messages). Used in scenarios with multiple load balanced Frontend.	RM_OCFrontEnd_1
PortalWillBeClosed	Determines whether to present the "Portal will be closed"-message on the screen. "1" will present the message, "0" will omit the message.	0, 1
PortalWillBeClosed.Text	The text that will be presented on screen when PortalWillBeClosed setting is set to "1".	Portal will be closed at 20:15.
PortalNowClosed	Determines whether to present the "Portal now closed"-message on the screen. "1" will present the message, "0" will omit the message.	0, 1
PortalNowClosed.Text	The text that will be presented on screen when PortalNowClosed setting is set to "1".	Portal is closed until 22:00.
RequireHTTPS	Specifies whether or not the Frontend will redirect to HTTPS when the protocol is HTTP. "1" will force the redirect to HTTPS. "0" will not redirect.	0 or 1
RequireDomain	Specifies the domain that the Frontend will run on. If the domain is not as specified the Frontend will redirect to the specified domain. The	www.mydomain.com

	domain is specified as "www.mydomain.com" without protocol.	
SecurityDomain	Specifies top level domain that the Frontend, Data Selector, FileStore and other custom application extensions must use in order to be able to make cross-window JavaScript communication. The domain is specified as "mydomain.com".	Resultmaker.com
ServerTransport	Specifies whether to use the HTTP or HTTPS transport protocol when the Frontend communicates with the Backend.	http, https
Server	Specifies the network name of the Backend server.	localhost
ServerPort	Specifies the HTTP port to use when the Frontend communicates with the Backend.	80
ServerResource	Specifies the relative path to the aspx file on the Backend server. The default location is "/oc4/process.aspx".	/oc4/Process.aspx
ServerWebService	Specifies the relative path to the Backend web service. The default location is "/oc4/OC.asmx".	/oc4/OC.asmx
DynamicDropDownWebService	Specifies the relative path to the Backend web service, that exposes Dynamic Dropdown collections. The default location is "/oc4/admin/transfer.asmx". Available from version 6.0	/oc4/admin/transfer.asmx
ServerConfigurationWebService	ServerConfigurationWebService: Specifies the relative path to the Backend web service, that exposes selected central configuration. The default location is "/oc4/accessconfiguration/accessconfiguration.asmx". -->	/oc4/accessconfiguration/accessconfiguration.asmx
ServerSecurityWebService	Specifies the relative path to the Backend web service, that exposes the Process Platform security api. The default location is "/oc4/Admin/Security.asmx".	/oc4/admin/security.asmx
ServerUsersWebService	Specifies the relative path to the Backend web service, that exposes the Process Platform user api. The default location is "/oc4/Users.asmx".	/oc4/Users.asmx
ServerPendingWebService	Specifies the relative path to the Backend web service, that exposes the Process Platform Pending list api.	

	This functionality is deprecated in version 6.0.	
ServerUseDelegationForRequests	Determines whether the Frontend should use delegation of windows credentials for the Backend calls. This requires that the Process Engine IIS application is setup with Windows Authentication. And is also required to be set to True if the Process Engine IIS application is setup to Windows Authentication. The delegated credentials can either be the account running the Frontend IIS application or the windows credentials of the end user logged. This is determined in the config setting ServerUseImpersonationForRequests. Allowed values are false and true. See also the config setting LoginInformationProvider. Available from version 6.0	true,false
ServerUseImpersonationForRequests	Determines whether the Frontend should impersonate calls to the Process Engine with the Windows credentials of the logged on user or not to impersonate. Allowed values are false and true. If true, it is required that the Process Engine IIS application is setup with Windows Authentication. See the config setting ServerUseDelegationForRequests. If false, it is required that the Frontend is configured with a LoginInformationProvider and a corresponding configure SecurityProvider in the Process Ending. See the config setting LoginInformationProvider. Available from version 6.0	true, false
DataSelectorURL	Relative path including server name to the Data Selector web page.	localhost/RMFrontEnd/DataSelector
FileStoreURL	Relative path including server name to the File Store web page.	localhost/RMFrontEnd/FileStore/UploadFile.aspx
ExportServerURL	Full http path to the Export Server web	http://localhost/exportserver/export.a

	service.	smx
StreamerURL	Relative path including server name to the File Streamer web page.	/RMFrontEnd/Layouts/Streamer.aspx
SubmitSignDataServiceURL	Full http path to the Signing web service.	http://localhost/BlueBox/Transactions_30.asmx
TicketServiceURL	Full http path to the Signing web service.	http://localhost/tokenservice/tokenservice.asmx
ProjectTypeFilter	Specifies the filter value to apply when rendering the project list for New projects. The filtering of the ProjectTypeFilter happens on the basis of the Category selected on the workflow at design time. Default is empty and it means no filtering.	
DefaultProjectListType	Specifies what type of project list to present if not selected explicitly by the user. "N" ("New"), "R" ("Running"), "C" ("Completed")	N,R,C
ProjectListSortNew	Specifies how to sort the project list for new projects/workflows. Sorting by multiple columns is possible by comma separation of the columns and descending sorting is possible. Example: "CreatedDate desc,ProjectText desc, Authority desc"	ProjectText,Authority
ProjectListSortRunning	Specifies how to sort the project list for started/running projects/workflows. Sorting by multiple columns is possible by comma separation of the columns and descending sorting is possible. Example: "CreatedDate desc,ProjectText,Authority"	CreatedDate desc,ProjectText,Authority
ProjectListSortCompleted	Specifies how to sort the project list for finished projects/workflows. Sorting by multiple columns is possible by comma separation of the columns and descending sorting is possible. Example: "CreatedDate desc,ProjectText,Authority"	CreatedDate desc,ProjectText,Authority
ProjectListSortPending	Specifies how to sort the project list for pending projects/workflows. Sorting by multiple columns is possible by comma separation of the columns and descending sorting is possible. Example: "DateBegin desc,Subject. This configuration element is no longer used as Pending List is	DateBegin desc,Subject

	deprecated in version 6.0.	
ProjectListItemsPerPage	Specifies how many projects/workflows to display per page on the project list pages.	500
LayoutsPathForServer	Specifies the relative path to the Layouts folder for the server site logic.	/RMFrontEnd/Layouts
LayoutsPathForClient	Specifies the relative path to the Layouts folder for the client site logic.	/RMFrontEnd/Layouts
PresentationFolderName	Specifies the relative path to the Presentation folder.	Presentation
SkinName	Specifies the path to the skin folder, which is located beneath the Presentation folder.	Default
XsltFolderName	Specifies the path to the Xslt folder, which is located beneath the folder specified in SkinName.	Xslt
CssFolderName	Specifies the path to the CSS folder, which is located beneath the folder specified in SkinName.	Css
ImagesFolderName	Specifies the path to the Image folder, which is located beneath the folder specified in SkinName.	Images
ProjectListTemplatesFolderName	Specifies the path to the ProjectListTemplates folder, which is located beneath the folder specified in SkinName. Empty value blocks the Project List with a message to the user.	ProjectListTemplates
RenderCssLinks	To be described.	0 or 1
TranslationsPath	Specifies the relative path to the folder containing files with translation text for presentation.	/RMFrontEnd/Layouts/Presentation/Translations
JSPPath	Specifies the relative path to the folder containing the core Java Script files.	/RMFrontEnd/Layouts/Core/JavaScript
AppletPath	Specifies the relative path to the folder containing the signing and login applet files.	/RMFrontEnd/Layouts/Core/Applets
AllowEnrichedWorkflow	Specifies whether it is possible to show the enriched workflow menu. Deprecated functionality.	0 or 1
EnableEnrichedWorkflow	Specifies whether to show the enriched workflow menu or not. Requires AllowEnrichedWorkflow to be set to 1. Deprecated functionality.	0 or 1
WorkflowOnLeft	Specifies whether to show the workflow menu to the left of to the right. 1 is left and 0 is right.	0 or 1
SessionRefreshURL	Specifies the relative url to the web page that can refresh the session. To disable "session refresh"	Layouts/SessionRefresh.aspx

	functionality, provide an empty value or remove the line.	
SessionRefreshOnClientActivity	Specifies whether client activity will refresh the session timeout period. Valid values are 1 and 0.	0 or 1
SessionExpirationWarningFactor	Specifies the factor to multiply to the session timeout to get the time to displays a warning about session timeout. 0 Disables the warning.	0 or some decimal value between 0 and 1 specified with . as decimal separator i.e. 0.75
SessionExpiredDisplayAlert	Specifies whether to present the session expired message to the user.	0 or 1
LoginInformationProvider	Specifies the provider to use for login handling. Supported types are: Resultmaker.OC.FrontEnd.Login.LoginInformationProvider, Resultmaker.OC.FrontEnd.Login: User name/password and certificate login. Resultmaker.OC.FrontEnd.HttpContextUser.LoginInformationProvider, Resultmaker.OC.FrontEnd.HttpContextUser: Simple Windows Authentication in the Frontend. Converts the Windows Authentication logon to a Backend logon and does not use delegation between Fronend and Backend. Requires Windows Authentication to be enabled and Anonymous Access to be disabled in the Frontend application in IIS. Resultmaker.OC.FrontEnd.HttpContextUser.LoginInformationWithTokenProvider, Resultmaker.OC.FrontEnd.HttpContextUser: Windows Authentication in the Frontend. Uses delegation to pass on the logon to the Backend logon. Requires Windows Authentication to be enabled and Anonymous Access to be disabled in the Frontend application in IIS and in the Backend application in the IIS. Requires that the ServerUseDelegationForRequests is set to true.	Resultmaker.OC.FrontEnd.Login.LoginInformationProvider, Resultmaker.OC.FrontEnd.Login
ProjectStartPermissionProvider	Deprecated	
SignPermissionProvider	Deprecated, should always be set to Resultmaker.OC.FrontEnd.AlwaysAllo	Resultmaker.OC.FrontEnd.CnnSign.SignPermissionProvider,

	wSignPermissionProvider, Resultmaker.OC.FrontEnd Allowed values: Resultmaker.OC.FrontEnd.CnnSign.Sig nPermissionProvider, Resultmaker.OC.FrontEnd.CnnSign and Resultmaker.OC.FrontEnd.AlwaysAllo wSignPermissionProvider, Resultmaker.OC.FrontEnd	Resultmaker.OC.FrontEnd.CnnSign
SatelliteManager	Controls the session sharing between Fronend and satellite applications like File Store client and Data Selector. The value specifies which class to load for session handling. Should always be set to Resultmaker.OC.SatelliteManager.Shar edSessionSatelliteManager, Resultmaker.OC.SatelliteManager	Resultmaker.OC.SatelliteManager.Shar edSessionSatelliteManager, Resultmaker.OC.SatelliteManager
Plugins.InfoPathForm	Controls the InfoPath integration. The value specifies which class to load for this. Should always be set to Resultmaker.OC.FrontEnd.Plugins.Info Path.InfoPathForm, Resultmaker.OC.FrontEnd.Plugins.Info Path.	Resultmaker.OC.FrontEnd.Plugins.Info Path.InfoPathForm, Resultmaker.OC.FrontEnd.Plugins.Info Path
ErrorLogPath	Specifies a folder path on disk, where internal Frontend logs are written. The folder must exist in order for logs to be created.	c:\Logs\OCErrors
HelpServer	Specifies a folder path on disk, where internal Frontend logs are written. The folder must exist in order for logs to be created.	/static/rm/documents
HelpFileExt	Specifies the extension for the help files.	.htm
DecimalSymbol	To be described	,
DateFormat	To be described	dd-mm-yyyy
JavaScriptDebug	Enables a mode, where the Process Consultant is notified about recomandations for form design.	0 or 1
MainPageURL	URL that links from the project/workflow list page to some specified page. If empty the link is not shown.	
MainPageText	Text for the URL that links from the project/workflow list page to some specified page. If empty the link and text is not shown. -->	
DefaultProjectExitLinkTarget	Target browser frame for the link from	

	the workflow page to the project/workflow list page. If empty no Target is used.	
DefaultProjectExitLinkURL	URL link from the workflow page to the project/workflow list page. Standard value is the javascript function "javascript:RM_GoBackToProjectList();".	"javascript:RM_GoBackToProjectList();"
DefaultProjectExitLinkText	Text for the link from the workflow page to the project/workflow list page. If no value is specified the default value from the Translations file is being used.	
LoginPath	The relative path to the login page. If empty the link to the default login page is being used. The standard value is empty.	
LogoutPath	The relative path to the logout page. The standard value is "Layouts/LogOut.aspx".	Layouts/LogOut.aspx
AfterLoginRedirURL	Url to redirect to after login (must be relative to "LoginPath", or absolute).	
AfterLogoutRedirURL	Url to redirect to after logout (must be relative to "LogoutPath", or absolute).	../?ShowLogin=1
NoCache	Specifies whether to enable or disable the browser caching. Setting the NoCache to 1 is recommended.	0 or 1
ServerType	The server type. Used for TDC signing applet. Valid values are Test or Production.	Test, Production
SignWithLoginCertificate	Specifies if the Signing applet will present only the certificate used for login. Valid values are 1 and 0.	0 or 1
SignCheckEnabled	Deprecated	0 or 1
SuperSensitiveSignCheckEnabled	Deprecated.	0 or 1
ThrowExceptions	If set to 0, exceptions are caught and presented in a nice and end user friendly format. If set to 1, normal yellow .NET exceptions are thrown. They can be used by load testing tools to validate application health.	0 or 1
HiddenValuesInSession	Stores session values in the ASP.NET session or in hidden post fields. If set to 0, session values are stored in hidden post fields. If set to 1, session values are stored in the ASP.NET session. This makes it easier to use load testing tools. -->	0 or 1
CacheMinutesForDropDownWebS	The time span in which the look ups in	60

ervice	the Dynamic Dropdown web service are cached in the Frontend. Note the unit of this cache is in minutes.	
IsRequireLoggedout	Determines whether to present a logout link or not.	1 or 0
ErrorMessageDisplayMode	There are three ways to present error messages in the Resultmaker Process Frontend application. Each one presents different amounts of information. The value is case sensitive.	minimal: In this mode the minimal amount of error information is displayed to the user. Only the custom error message is displayed. hidden: This displays the normal custom errors message and by pressing CTRL+SHIFT and clicking on the custom errors message, the usual error details will be displayed. detail: This is display both custom error and detail error to the frontend where an error is occurred.
RequireAuthenticationMode	Specifies the Authentication Mode	Windows, Forms
Section: Resultmaker.OC.EntityFrontEnd		
EntityConnectionString		
EntityIdQueryStringKey		EntityID
EnableEntitySessionCaching		0 or 1
ProjectTypeFilterByEntityID		0 or 1
ShowHeader		0 or 1
EnablePrefill		0 or 1
EnableSigning		0 or 1

2.2 Web.config

The web.config contains a few configuration parameters. They are documented here.

```

<appSettings>
  <add key="RM_SecurityDomain" value="pptest60" />
  <add key="RM_SharedSessionPersister"
value="Resultmaker.Web.SharedSessionState.InProcessSharedSessionPersister,
Resultmaker.Web.SharedSessionState,Version=6.0.0000.0, Culture=neutral,
PublicKeyToken=e135b4d1c4e6af7d" />
  <add key="RM_OCFrontEnd.ascx" value="Layouts/FullHosting.ascx" />
</appSettings>

....

<system.serviceModel>
  <bindings>
    <webHttpBinding>
      <binding name="BasicpointBinding">

```



```

    <security mode="TransportCredentialOnly">
      <transport clientCredentialType="Windows" />
    </security>
  </binding>
</webHttpBinding>
</bindings>
<behaviors>
  <endpointBehaviors>
    <behavior name="RMFrontEnd.RmAjaxControlsWebServiceAspNetAjaxBehavior">
      <enableWebScript />
    </behavior>
  </endpointBehaviors>
  <serviceBehaviors>
    <behavior name="metadataAndDebug">
      <serviceMetadata httpGetEnabled="true" httpGetUrl="" />
      <serviceDebug httpHelpPageEnabled="true" includeExceptionDetailInFaults="true" />
    </behavior>
  </serviceBehaviors>
</behaviors>
<serviceHostingEnvironment aspNetCompatibilityEnabled="true" />
<services>
  <service name="RMFrontEnd.RmAjaxControlsWebService"
behaviorConfiguration="metadataAndDebug">
    <endpoint address=""
behaviorConfiguration="RMFrontEnd.RmAjaxControlsWebServiceAspNetAjaxBehavior"
binding="webHttpBinding" bindingConfiguration="BasicpointBinding"
contract="RMFrontEnd.RmAjaxControlsWebService">
      <identity>
        <dns value="localhost" />
      </identity>
    </endpoint>
  </service>
</services>
</system.serviceModel>

```

CONFIGURATION SETTING NAME	DESCRIPTION	EXAMPLE VALUES
Section: system.serviceModel		
Security mode	This WCF configuration for the Ajax web service for the dynamic dropdowns needs to be configured for the http protocol you are running on the server. For HTTP you need to set the following security mode for the binding: <pre><security mode="TransportCredentialOnly"></pre> For HTTPS you need to set the following security mode for the binding: <pre><security mode="Transport"></pre>	TransportCredentialOnly, Transport

3 Client Scripts

3.1 Inter-frame communication

By using the “*IFRAME*” or “*FRAMESET*” HTML tags, the main HTML document may contain the frames with their own HTML documents inside.

To enable an inter-frame communication with the use of JavaScript calls, FrontEnd sets the security domain of the main HTML document (*document.domain* property in JavaScript) to the shortest possible domain (defined in “SecurityDomain” config setting since v.2.11). E.g. for single-word domain “MyServer” (usually used inside the intranet) the security domain is set to “MyServer”; for longer names like “MyServer.MyDomain.MyCompany.com” the security domain contains only two last words, i.e. “MyCompany.com” in this example.

In FrontEnd v2.10.0.0 and earlier, this is done only if the main document contains an IFRAME with either codepicker or file store.

In FrontEnd v2.10.0.1 and later, this is done on every page, to enable an inter-frame communication with any custom HTML document in the frame, and to enable the reliable session timeout handling (refreshing the session, timeout warning etc.).

In every HTML document in the frame that pretends to communicate with the main document it is necessary to set explicitly the security domain to exactly the same value as the main document does. Even if the default security domain is already correct, it is necessary anyway to set it explicitly to this value. Both codepicker and file store do that. In any custom HTML document it is necessary to do the same. For security reasons it is a good practice to set the security domain to the value determined on the server, and not on the client. For details see any available documentation of *document.domain* property.

3.2 Class: RM_ShowHide

3.2.1 Declaring and Using

3.2.2 Properties

3.2.2.1 *IsHidden*

Property: *RM_ShowHide.IsHidden*

FrontEnd Version: 1.6

Description:

Example:

3.2.3 Methods

3.2.3.1 *AddElementById*

Method: *RM_ShowHide.AddElementById()*

FrontEnd Version: 1.6

Description: Adds a layout ID to the internal collection of *RM_ShowHide* object.

Example: SHobj = GetShowHideObject();
 SHobj.AddElementById("row_1");

3.2.3.2 RemoveElementById

Method: *RM_ShowHide.RemoveElementById()*

FrontEnd Version: 1.6

Description: Remove a layout ID from the internal collection of *RM_ShowHide* object.

Example:

3.2.3.3 AddElementByQuestionName

Method: *RM_ShowHide.AddElementByQuestionName()*

FrontEnd Version: 1.6

Description: Adds a question element and all its answers to the internal collection of *RM_ShowHide* object.

Example: objShow.AddElementByQuestionName("MyRadio");

3.2.3.4 RemoveElementByQuestionName

Method: *RM_ShowHide.RemoveElementQuestionName()*

FrontEnd Version: 1.6

Description: Removes a question element and all its answers from the internal collection of *RM_ShowHide* object.

Example: objShow.RemoveElementByQuestionName("MyRadio");

3.2.3.5 AddElementByAnswerName

Method: *RM_ShowHide.AddElementByAnswerName (strQuestion, strAnswer)*

FrontEnd Version: 2.5

Description: Adds a single answer (specified by question name and answer name) to the internal collection of *RM_ShowHide* object.

Example: objShow.AddElementByAnswerName("MyRadio", "Option0");

3.2.3.6 RemoveElementByAnswerName

Method: *RM_ShowHide.RemoveElementByAnswerName (strQuestion, strAnswer)*

FrontEnd Version: 2.5

Description: Removes a single answer (specified by question name and answer name) from the internal collection of *RM_ShowHide* object.

Example: `objShow.RemoveElementByAnswerName("MyRadio", "Option0");`

3.2.3.7 Hide

Method: *RM_ShowHide.Hide()*

FrontEnd Version: 1.6

***Description:** Hides all elements in an internal collection of *RM_ShowHide* object.

***Example:** `SHobj = GetShowHideObject();
SHobj.AddElementById("row_1");
SHobj.Hide();`

3.2.3.8 Show

Method: *RM_ShowHide.Show()*

FrontEnd Version: 1.6

***Description:** Show all elements in an internal collection of *RM_ShowHide* object.

***Example:** `SHobj = GetShowHideObject();
SHobj.AddElementById("row_1");
SHobj.Show();`

3.2.3.9 ClearElements

Method: *RM_ShowHide.ClearElements()*

FrontEnd Version: 1.6

***Description:** Clear all fields in an internal collection of *RM_ShowHide* object.

***Example:** `SHobj = GetShowHideObject();
SHobj = AddElementByQuestionName("TekstField0");
SHobj.ClearElements();`

3.2.3.10 RequiredSet

Method: *RM_ShowHide.RequiredSet (bRequired)*

FrontEnd Version: 2.5

Description: For *bRequired* = {true}: marks all included elements as “required” (“*” is added and validation is affected respectively).
 For *bRequired* = {false}: marks all included elements as “not required” (“*” is removed and validation is affected respectively).
 Does not affect the elements with “Required” property set to “Required”.
 Due to *RequiredOn()* and *RequiredOff()* methods this method may be considered as redundant, but in some cases its usage may simplify the conditions in code.

Example: `objShow.RequiredSet(true);`

3.2.3.11 RequiredOn

Method: *RM_ShowHide.RequiredOn()*

FrontEnd Version: 2.5

Description: Marks all included elements as “required” (“*” is added and validation is affected respectively). Does not affect the elements with “Required” property set to “Required”.

Example: `objShow.RequiredOn();`

3.2.3.12 RequiredOff

Method: *RM_ShowHide.RequiredOff()*

FrontEnd Version: 2.5

Description: Marks all included elements as “not required” (“*” is removed and validation is affected respectively). Does not affect the elements with “Required” property set to “Required”.

Example: `objShow.RequiredOff();`

3.2.4 Modes**3.2.4.1 Safe Mode****FrontEnd Version: 1.6**

***Description:** When safe mode is on *RM_ShowHide.Hide()* will not hide the object, if any fields in the internal collection contains data. Default is off.

Methods: *SafeModeOn()*, *SafeModeOff()*

***Example:** `SHobj = GetShowHideObject();
 SHobj = AddElementByQuestionName("TekstField0");
 SHobj.SafeModeOn();
 SHobj.Hide(); //only hide if empty
 SHobj.SafeModeOff();`

3.2.4.2 Clear Mode

FrontEnd Version: 1.6

***Description:** Clears all fields in an internal collection of RM_ShowHide object, next time the object is set to hidden. Default is off.

Methods: *ClearModeOn(), ClearModeOff()*

***Example:**

```
SHobj = GetShowHideObject();
SHobj = AddElementByQuestionName("TekstField0");
SHobj.ClearModeOn();
SHobj.Hide(); //hide and clear value
```

3.2.4.3 Hide Modes

HIDE MODE	METHOD	DESCRIPTION	FEVER
Collapse (removes completely from page)	<i>SetToCollapse()</i>		1.6
Hide (vanishes though leaves a placeholder)	<i>SetToHide()</i>		1.6
Disabled (get greyed out is unable to be edited)	<i>SetToDisabled()</i>		1.6
Read Only	<i>SetToReadOnly()</i>		1.6

***Example:**

```
SHobj = GetShowHideObject();
SHobj = AddElementByQuestionName("TekstField0");
SHobj.SetToDisabled();
SHobj.Hide(); //gray out the field
```

3.2.5 Correct order of modes

Recommended way to turn on/off the different modes

```
SomeObject.SafeModeOn();
SomeObject.Hide();
SomeObject.SafeModeOff();
SomeObject.ClearModeOn();
```

Navigating away and to the page will not cause any information to be deleted.

Information will only be deleted when the user activates some hide mechanism like pushing a button.

3.3 Navigation buttons ("Back"/"Continue") functions

3.3.1 RM_RenameBackButton

Function: RM_RenameBackButton(*caption text*)

FrontEnd Version: 2.11

Description: Defines the custom text as a caption for the "Back" button.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>caption text</i>	No	The text to use

Example:

```
RM_RenameBackButton("Previous page");
```

3.3.2 RM_RenameContinueButton

Function: RM_RenameContinueButton(*caption text*)

FrontEnd Version: 2.11

Description: Defines the custom text as a caption for the "Continue" button.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>caption text</i>	No	The text to use

Example:

```
RM_RenameContinueButton("Next page");
```

3.3.3 RM_SetHidingBackButton

Function: RM_SetHidingBackButton(*hide*)

FrontEnd Version: 2.11.1

Description: For *hide* = {true} it arbitrarily hides the "Back" button; for *hide* = {false} it arbitrarily shows the "Back" button; for *hide* = {null} it reverts the arbitrary hiding/showing-related changes that any functions introduced to the "Back" button.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>hide</i>	Yes	Null, or Boolean value that indicates whether to hide or show the "Back" button.

Example:

```
RM_SetHidingBackButton(true);
```

3.3.4 RM_HideBackButton

Function: RM_HideBackButton()

FrontEnd Version: 2.11.1

Description: Arbitrarily hides the "Back" button.

Returns: none

Arguments: none

Example:

```
RM_HideBackButton();
```

3.3.5 RM_ShowBackButtonWeak

Function: RM_ShowBackButtonWeak([Id])

FrontEnd Version: 2.11.1

Description: Tries to show the "Back" button by releasing the hiding lock identified by the Id (if supplied), and by reverting the arbitrary hiding. After that the button may remain hidden if there are any other conditions keeping it hidden.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>allowingId</i>	Yes	Id of the hiding lock.

Example:

```
RM_ShowBackButtonWeak();
RM_ShowBackButtonWeak("MyElement_id");
```

3.3.6 RM_ShowBackButtonStrong

Function: RM_ShowBackButtonStrong()

FrontEnd Version: 2.11.1

Description: Arbitrarily shows the "Back" button.

Returns: none

Arguments: none

Example:

```
RM_ShowBackButtonStrong();
```

3.3.7 RM_SetHidingContinueButton

Function: RM_SetHidingContinueButton(*hide*)

FrontEnd Version: 2.11.1

Description: For *hide* = {true} it arbitrarily hides the "Continue" button; for *hide* = {false} it arbitrarily shows the "Continue" button; for *hide* = {null} it reverts the arbitrary hiding/showing-related changes that any functions introduced to the "Continue" button.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
------	----------	-------------

<i>hide</i>	Yes	Null, or Boolean value that indicates whether to hide or show the "Continue" button.
-------------	-----	--

Example:

```
RM_SetHidingContinueButton(true);
```

3.3.8 RM_HideContinueButton

Function: RM_HideContinueButton()

FrontEnd Version: 2.11.1

Description: Arbitrarily hides the "Continue" button.

Returns: none

Arguments: none

Example:

```
RM_HideContinueButton();
```

3.3.9 RM_ShowContinueButtonWeak

Function: RM_ShowContinueButtonWeak([Id])

FrontEnd Version: 2.11.1

Description: Tries to show the "Continue" button by releasing the hiding lock identified by the Id (if supplied), and by reverting the arbitrary hiding. After that the button may remain hidden if there are any other conditions keeping it hidden.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>allowingId</i>	Yes	Id of the hiding lock.

Example:

```
RM_ShowContinueButtonWeak();
RM_ShowContinueButtonWeak("MyElement_id");
```

3.3.10 RM_ShowContinueButtonStrong

Function: RM_ShowContinueButtonStrong()

FrontEnd Version: 2.11.1

Description: Arbitrarily shows the "Continue" button.

Returns: none

Arguments: none

Example:

```
RM_ShowContinueButtonStrong();
```

3.3.11 RM_SetDisablingBackButton

Function: RM_SetDisablingBackButton(*disable*)

FrontEnd Version: 2.11.1

Description: For *disable* = {true} it arbitrarily disables the "Back" button; for *disable* = {false} it arbitrarily enables the "Back" button; for *disable* = {null} it reverts the arbitrary disabling/enabling-related changes that any functions introduced to the "Back" button.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>disable</i>	Yes	Null, or Boolean value that indicates whether to disable or enable the "Back" button.

Example:

```
RM_SetDisablingBackButton(true);
```

3.3.12 RM_DisableBackButton

Function: RM_DisableBackButton()

FrontEnd Version: 2.11.1

Description: Arbitrarily disables the "Back" button.

Returns: none

Arguments: none

Example:

```
RM_DisableBackButton();
```

3.3.13 RM_EnableBackButtonWeak

Function: RM_EnableBackButtonWeak([Id])

FrontEnd Version: 2.11.1

Description: Tries to enable the "Back" button by releasing the disabling lock identified by the Id (if supplied), and by reverting the arbitrary disabling. After that the button may remain disabled if there are any other conditions keeping it disabled.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>allowingId</i>	Yes	Id of the disabling lock.

Example:

```
RM_EnableBackButtonWeak();
RM_EnableBackButtonWeak("MyElement_id");
```

3.3.14 RM_EnableBackButtonStrong

Function: RM_EnableBackButtonStrong()

FrontEnd Version: 2.11.1

Description: Arbitrarily enables the "Back" button.

Returns: none

Arguments: none

Example:

```
RM_EnableBackButtonStrong();
```

3.3.15 RM_SetDisablingContinueButton

Function: RM_SetDisablingContinueButton(*disable*)

FrontEnd Version: 2.11.1

Description: For *disable* = {true} it arbitrarily disables the "Continue" button; for *disable* = {false} it arbitrarily enables the "Continue" button; for *disable* = {null} it reverts the arbitrary disabling/enabling-related changes that any functions introduced to the "Continue" button.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>disable</i>	Yes	Null, or Boolean value that indicates whether to disable or enable the "Continue" button.

Example:

```
RM_SetDisablingContinueButton(true);
```

3.3.16 RM_DisableContinueButton

Function: RM_DisableContinueButton()

FrontEnd Version: 2.5.4020

Description: Arbitrarily disables the "Continue" button.

Returns: none

Arguments: none

Example:

```
RM_DisableContinueButton();
```

3.3.17 RM_EnableContinueButtonWeak

Function: RM_EnableContinueButtonWeak([Id])

FrontEnd Version: 2.11.1

Description: Tries to enable the "Continue" button by releasing the disabling lock identified by the Id (if supplied), and by reverting the arbitrary disabling. After that the button may remain disabled if there are any other conditions keeping it disabled.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>allowingId</i>	Yes	Id of the disabling lock.

Example:

```
RM_EnableContinueButtonWeak();
RM_EnableContinueButtonWeak("MyElement_id");
```

3.3.18 RM_EnableContinueButtonStrong

Function: RM_EnableContinueButtonStrong()

FrontEnd Version: 2.11.1**Description:** Arbitrarily enables the "Continue" button.**Returns:** none**Arguments:** none

Example:

```
RM_EnableContinueButtonStrong();
```

3.3.19 RM_EnableContinueButton**Function:** RM_EnableContinueButton ()**FrontEnd Version:** 2.5.4020**Description:** Same as *RM_EnableContinueButtonWeak()* without arguments; kept for backward compatibility.**Returns:** none**Arguments:** none

Example:

```
RM_EnableContinueButton();
```

3.3.20 RM_ShowContinueButton**Function:** RM_ShowContinueButton(*allowingId*)**FrontEnd Version:** 2.2**Description:** Same as *RM_EnableContinueButtonWeak(allowingId)*; kept for backward compatibility.**Returns:** none**Arguments:**

NAME	OPTIONAL	DESCRIPTION
<i>allowingId</i>	Yes	Id of the disabling lock.

Example:

```
RM_ShowContinueButton();
RM_ShowContinueButton("MyElement_id");
```

3.3.21 RM_ResetBackButton**Function:** RM_ResetBackButton()**FrontEnd Version:** 2.11.1**Description:** Reverts the arbitrary hiding/showing- and disabling/enabling-related changes that any functions introduced to the "Back" button.**Returns:** none**Arguments:** none

Example:

```
RM_ResetBackButton();
```

3.3.22 RM_ResetContinueButton**Function:** RM_ResetContinueButton()

FrontEnd Version: 2.11

Description: Reverts the arbitrary hiding/showing- and disabling/enabling-related changes that any functions introduced to the "Continue" button.

Returns: none

Arguments: none

Example:

```
RM_ResetContinueButton();
```

3.4 Collections

3.4.1 RM_Addon_Params

Collection: RM_Addon_Params[]

FrontEnd Version: 1.6

Type: string

Description: This collection contains all the parameter strings for the page's add-on blocks. Change the appropriate string if your add-on needs custom URL parameters. The string will be appended to the end of the URL.

Example:

```
RM_Addon_Params['MyAddon'] = 'param1="fred"&param2="michael"';
```

3.5 Functions

3.5.1 Some examples of most used functions

3.5.1.1 Radio button

Get: `getanswer("MyQuestion","MyOption").checked`

Set: `getanswer("MyQuestion","MyOption").checked = true/false`

3.5.1.2 Dropdown

Get: `getquestion("MyQuestion").value`

Set: `getquestion("MyQuestion").value = "MyOption"`

3.5.1.3 Checkbox

Get: `getquestion("MyQuestion").checked`

Set: `getquestion("MyQuestion").checked = true/false`

3.5.1.4 Text area/field

Get: `getquestion("MyQuestion").value`

Set: `getquestion("MyQuestion").value`

3.5.2 getquestion

Function: `getquestion(question name)`

FrontEnd Version: 1.6

Description: Gets a question object for JavaScript manipulation.

Returns: question object

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question name</i>	No	The name of the question

Example:

```
getquestion("MyQuestion").value = "New Value";
getquestion("MyQuestion").focus(); //set focus to the field
getquestion("MyQuestion").readOnly = true; //set a field to read only
getquestion("MyQuestion").checked = true; //set a check box to checked
getquestion("MyQuestion").innerHTML = "fred"; //change the question text to fred.
```

3.5.3 getdescription (obsolete)

Function: `getdescription(question name)`

FrontEnd Version: 1.6

Description: Gets a description object (of type DIV) for JavaScript manipulation. Modifying the properties of this object affects the description visible on the page, but it doesn't affect the description visible as a tooltip.

This function is obsolete, maintained for backward compatibility only; use `setdescription()` instead that affects also the tooltip.

Returns: DIV object

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question name</i>	No	Name of the question

Example:

```
getdescription("MyQuestion").innerText = "My Description"; This is not very browser
compatible
getdescription("MyQuestion").innerHTML = "My <b>Description</b>";
```

3.5.4 setdescription

Function: `setdescription(question name, description)`

FrontEnd Version: 2.6.1

Description: Sets a description text for the specified question. It affects both the description visible on the page and the description visible as a tooltip.

Returns: none

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question name</i>	No	Name of the question

<i>description</i>	No	Description text
--------------------	----	------------------

Example:

```
setdescription("MyQuestion", "My Description");
setdescription("MyQuestion", "My <b>Description</b>");
setdescription("MyQuestion", "");
```

3.5.5 getquestionname

Function: `getquestionname(question object)`

FrontEnd Version: 1.6

***Description:** Gets the question name. Can be very useful in conjunction with the java script word "this"

Returns: the question name.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question object</i>	No	

* Example: `//call this with an event like onChange="alertTxt(this);"`

```
function alertTxt(obj)
{
    var Qname = getquestionname(obj);
    alert(getquestion(Qname).value); //writes the value of the field
}
```

3.5.6 getanswer

Function: `getanswer(question name, answer name)`

FrontEnd Version: 1.6

***Description:** Gets or set the value of a radio button

Returns: the answer object (usually a radio button object)

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question name</i>	No	
<i>answer name</i>	No	

*Example: `getanswer("RadioButton0", "Option1").checked = true;`

3.5.7 getdate

Function: `getdate(question name)`

FrontEnd Version: 1.6

***Description:** Useful to manipulate a date written in the dd-mm-yyyy format.

Returns: the date as a number so that it can be compared with other dates. (The number of milliseconds since midnight 1 January, 1970.)

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question name</i>	No	Can be a question name or the word "today".
<i>date part</i>	Yes	A string specifying the part of the date to be returned. It can be "day", "month" or "year". If omitted getdate returns the number of milliseconds since 1 January 1970.

***Example:** //display the date of today

```
getquestion("TekstField0").value = displaydate(getdate("today"));
```

3.5.8 displaydate

Function: displaydate(*date in number format*)

FrontEnd Version: 1.6

***Description:** Gets the date in dd-mm-yyyy format.

Returns: The date in our standard text format.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>date in number format</i>	No	

***Example:** //display the date of today

```
getquestion("TekstField0").value = displaydate(getdate("today"));
```

3.5.9 RM_getnumber

Function: RM_getnumber(*question name [, no of decimals]*)

FrontEnd Version: 1.6

Description: Takes a number in JavaScript format and returns a number as a string with the specified number of decimals in the currently configured number display format.

Returns: A number in JavaScript format so that it can be compared and calculated.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question name</i>	No	
<i>no of decimals</i>	Yes	

Example:

```
// This example adds 10% to a question's value.
getquestion("myquestion").value = RM_displaynumber( RM_getnumber("myquestion") * 1.1
);
```


3.5.10 RM_displaynumber

Function: RM_displaynumber(*number* [, *no of decimals*])

FrontEnd Version: 1.6

Description: Displays a number as a string in the current number display format.

Returns: string

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>number</i>	No	A number in JavaScript format
<i>no of decimals</i>	Yes	

Example:

```
// This example adds 10% to a question's value.
getquestion("myquestion").value = RM_displaynumber( RM_getnumber("myquestion") * 1.1
);
```

3.5.11 getselectedname

Function: getselectedname(*question name*)

FrontEnd Version: 1.6

Description:

Returns: the answer text of the answer that is currently selected.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question name</i>	No	The name of the drop down box

Example:

3.5.12 RM_ActivatePDFButton

Function: RM_ActivatePDFButton()

FrontEnd Version: 1.6

Description: "Pushes" the open PDF button.

Returns: none

Arguments: none

Example:

```
RM_ActivatePDFButton();
```

3.5.13 RM_GetProjectID

Function: RM_GetProjectID()

FrontEnd Version: 1.6**Description:** Returns the current project ID.**Returns:** a string containing the current project ID.**Arguments:** none**Example:**

```
Var myProjectID = RM GetProjectID();
```

3.5.14 getquestiontextobj**Function:** getquestiontextobj (*question name*)**FrontEnd Version: 1.6****Description:** Gets the question text object so that the question text can be changed at runtime.**Returns:** a div object containing the question text.**Arguments:**

NAME	OPTIONAL	DESCRIPTION
<i>question name</i>	No	The name of question of which text you want the object of.

Example:

```
getquestiontextobj("myQuestion").innerHTML = "New question text";
```

3.5.15 RM_ActivateContinueButton**Function:** RM_ActivateContinueButton()**FrontEnd Version: 1.6****Description:** "Pushes" the open continue button.**Returns:** none**Arguments:** none**Example:**

```
RM_ActivateContinueButton();
```

3.5.16 RM_OpenLink

Function: RM_OpenLink (*url*)

FrontEnd Version: 1.7.1

Description: Opens a URL in a new window.

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>url</i>	No	The http link to open.

Example:

```
RM_OpenLink("http://www.yahoo.com/");
```

3.5.17 RM_MoveNext(0)

Function: RM_MoveNext(0)

FrontEnd Version: 2.2

Description: Prevents the project to proceed, it submits, but stays on the same page.

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
?	No	?.

Example:

```
RM_MoveNext (0) ;
```

3.5.18 RM_GetQuestionType

Function: RM_GetQuestionType(question)

FrontEnd Version: 2.4

Description: Gets the type as a string of the given question.

Returns: The question type as a string.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>question</i>	No	The name of the question.

Example:

```
var questionType = RM_GetQuestionType("MyQuestion");
```

3.5.19 RM_OpenCodePicker

Function: RM_OpenCodePicker (pickerName)

FrontEnd Version: 2.4

Description: Runs and / or refreshes a code picker.

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>pickerName</i>	No	The name of the codepicker (not to be confused with the application name.)

Example:

```
RM_OpenCodePicker("MyCodePicker");
```

3.5.20 RM_OpenFileStore

Function: RM_OpenFileStore (fileStoreName)

FrontEnd Version: 2.4

Description: Runs and / or refreshes a File Store client.

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>fileStoreName</i>	No	The name of the file store

Example:

```
RM_OpenFileStore("MyFileStore");
```

3.5.21 RM_Validate

Function: RM_Validate(input, validation)

FrontEnd Version: 2.4

Description: Validates a string using standard validation routines.

Returns: True or the error message as a string.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>input</i>	No	The string to be validated
<i>validation</i>	No	The validation routine to use.

Example:

```
var result = RM_Validate("485734639", "CheckCpr");
```

3.5.22 RM_GoToActivity

Function: RM_GoToActivity(activity)

FrontEnd Version: 2.4

Description: Redirects to the specified activity. You need to specify the phase, activityset and activity separated by backslashes (\).The path takes the format of *\phase\activityset\activity*. The path **must** start with a backslash.

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>activity</i>	No	The complete path of the activity to go to. You need to specify the phase, activityset and activity separated by backslashes (\).The path takes the format of <i>\phase\activityset\activity</i> . The path must start with a backslash. Note that in JavaScript notation of the string the backslash (\) has to be escaped with another backslash to preserve the meaning of the first one. Thus the final string has to be entered with the use of double backslashes as <i>"\\myPhase\\myActivityset\\myActivity"</i> .

Example:

```
RM_GoToActivity("\\myPhase\\myActivityset\\myActivity");
```

3.5.23 RM_FormatNumber

Function: RM_FormatNumber(qname,rmNumber)

FrontEnd Version: 6.0

Description: This function formats, validates and inserts a number (rmNumber) into the specified form question (qname). The format will be applied according to the form questions properties set in the Process Designer. The function works with the question types percentage, per mille, decimal, currency and integer.

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>qname</i>	No	The Question name.
<i>rmNumber</i>	No	The value to format and display.

Example:

```
RM_FormatNumber("TextField0", 23.2);
```

3.5.24 GetShowHideObject

Function: GetShowHideObject()

FrontEnd Version:

Description: Gets the ShowHide class object with its functionality.

Returns: none.

Arguments: none

Example:

```
GetShowHideObject();
```

3.5.25 RM_ShowHidelconByElem

Function: RM_ShowHidelconByElem(oElem,pShow,pKeepSpace)

FrontEnd Version:

Description: Show or hide icon of specified element object

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>oElem</i>	No	The element object
<i>pShow</i>	No	The element icon will be showed or not.
<i>pKeepSpace</i>	No	Preseve spacing or not.

Example:

```
var element = document.getElementById("TextField0 ID");
RM_ShowHideIconByElem(element,true,true);
```

3.5.26 RM_ShowHidelconByName

Function: RM_ShowHidelconByName(pName,pShow,pKeepSpace)

FrontEnd Version:

Description: Show or hide icon of specified element name

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>pName</i>	No	The element name
<i>pShow</i>	No	The element icon will be showed or not.
<i>pKeepSpace</i>	No	Preseve spacing or not.

Example:

```
RM_ShowHideIconByName("TextField0",true,true);
```

3.5.27 getquestiontext

Function: getquestiontext(RM_obj)

FrontEnd Version:

Description: Show or hide icon of specified element name

Returns: The text on specified question element object.

Arguments:

NAME	OPTIONAL	DESCRIPTION
<i>RM_obj</i>	No	The question element object.

Example:

```
var element = document.getElementById("TextField0 ID");
getquestiontext(element);
```

3.5.28 RM_calnumber**Function:** RM_calnumber(qname1,operator,qname2,pDecimals)**FrontEnd Version:****Description:** Calculate mathematic operation based on 2 element name**Returns:** none.**Arguments:**

NAME	OPTIONAL	DESCRIPTION
<i>Qname1</i>	No	The first element name
<i>Operator</i>	No	The mathematic operation to be performed
<i>Qname2</i>	No	The second element name.
<i>pDecimal</i>	No	Decimal digit

Example:

```
RM_calnumber("TextField0","+", "TextFiled1",2);
// Note:If TextField0 have value 5 and TextField1 has value 5, the function will
return 10.00
```

3.5.29 RM_AttachOnLoadEvent**Function:** RM_AttachOnLoadEvent(FunctionName)**FrontEnd Version:** 6.0

Description: This function will try to schedule a call to the javascript function given in the "FunctionName" input parameter on the OnLoad event of the page. That is you can use this function, if your code needs to run OnLoad. If it is not possible to attach the function call to the OnLoad event, a delay of 100 ms is made and the function is called at that time.

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
FunctionName	No	The name of the function to be called when the page has been loaded.

Example:

<code>RM_AttachOnLoadEvent("LoadMap");</code>

3.5.30 RM_ConditionalAlert

Function: RM_ConditionalAlert(elementToOperateBy, logicalOperator, conditionValue, alertIfFalseMessage)

FrontEnd Version: 6.0

Description: This function will try to schedule a call to the javascript function given in the "FunctionName" input parameter on the OnLoad event of the page. That is you can use this function, if your code needs to run OnLoad. If it is not possible to attach the function call to the OnLoad event, a delay of 100 ms is made and the function is called at that time.

Returns: boolean

Arguments:

NAME	OPTIONAL	DESCRIPTION
elementToOperateBy	No	The question element to evaluate the condition value against.
logicalOperator	No	The logical operator to evaluate against. Can be <, >, =, <= or >=.
conditionValue	No	Specifies the condition value to evaluate by.
alertIfFalseMessage	No	The text message to present if the condition evaluates to false.

Example:

Events	
OnChange	<code>RM_ConditionalAlert('Group0/TextField0', '>=', '10', 'Specialekarakteren skal være 10 eller derover for at kunne søge', true);</code>
OnClick	
OnEnter	
OnKeyPress	
OnLeave	
OnValidate	

```
RM_ConditionalAlert('Group0/TextField0', '>=', '10', 'Specialekarakteren skal være 10 eller derover for at kunne søge', true);
```


3.5.31 RM_ShowHideElement

Function: RM_ShowHideElement(elementToShowHide, elementToOperateBy, conditionValue, showHideAction, clearModeOn)

FrontEnd Version: 6.0

Description: Shows or hides the specified element, based on the specified element and show hide action.

If the condition is not met, the opposite show/hide action is activated.

Returns: none.

Arguments:

NAME	OPTIONAL	DESCRIPTION
elementToShowHide	No	Name of the question element to show or hide.
elementToOperateBy	No	Specifies the question element to evaluate the criteria against.
conditionValue	No	Specifies the condition value to evaluate by.
showHideAction	No	Determines whether to show or hide if the evaluation returns true.
clearModeOn	No	Specifies whether to set the clear mode or not.

Example:

Events	
OnChange	RM_ShowHideElement('Group0', 'Candidates/dt:Fax', '12345', 'hide', false);
OnClick	
OnEnter	
OnKeyPress	
OnLeave	
OnValidate	

```
RM_ShowHideElement('Group0', 'Candidates/dt:Fax', '12345', 'hide', false);
```

3.5.32 RM_GetShowHideObjectFromElement

Function: RM_GetShowHideObjectFromElement(element, clearModeOn, setToCollapse)

FrontEnd Version: 6.0

Description: Get the show hide element by specifying the question element name.

Returns: object.

Arguments:

NAME	OPTIONAL	DESCRIPTION
element	No	Name of the question element to get the show hide object for.
clearModeOn	No	Determines whether to set the clear mode or not. Specified as a boolean string.
setToCollapse	No	Determines wheter to set the collapse mode or not. Specified as a boolean string.

Example:

```
var objShowHideElement = RM GetShowHideObjectFromElement(elementToShowHide, 'true',  
'true');
```

3.6 Events

3.6.1 onclick

FrontEnd Version: 1.6

Description:

Example:

3.6.2 onchange

FrontEnd Version: 1.6

Description:

Example:

3.6.3 RM_CustomValidation()

FrontEnd Version: 1.6

Description: The RM_CustomValidation() function is a function that can be implemented by developers and is called as part of the page submit.

The function must implement its own validation alerts and return the validation result as either true or false. Returning true will allow the page to be submitted, false will not.

Example:

```
// The following example show how the RM_CustomValidation() implements the mandatory
return values true or false.

function RM_CustomValidation(){
    // Do stuff on submit
    // Return true if submit is allowed, false otherwise

    var amount = getquestion('DropDown0').value;
    if (amount > 1 )
        return true;
    else{
        alert("You selected less than two aircraft");
        return false;
    }
}
```

3.7 Constants

3.7.1 RM_oneday

FrontEnd Version: 1.6

Description: Represents one day when calculating dates.

Example:

```
// The following example adds one day to a question of date type.
getquestion("MyQuestion").value = displaydate(getdate("MyQuestion") + RM_oneday);
```

3.7.2 RM_oneweek

FrontEnd Version: 1.6

Description: Represents one week when calculating dates.

Example:

3.7.3 RM_oneyear

FrontEnd Version: 1.6

Description: Represents one day when calculating dates.

Example:

3.8 Debug Client Script

3.8.1 Debug Client Script in Microsoft Visual Studio 2005

It is possible to debug client scripts in the following way:

1. Open the page you want to debug in a browser
2. Start Microsoft Visual Studio and choose Debug and Attach to Process...
3. In the Attach to Process window find your process and Attach
4. Open Script Explorer, if not open already, choose Debug, Windows and Script Explorer
5. Double click the file containing the script you want to debug
6. Set breakpoint(s) in the file
7. Refresh the browser window, and you are ready to debug in Microsoft Visual Studio.

3.8.2 Debug Client Script in Internet Explorer with Microsoft Script Debugger

First you need to install Microsoft Script Debugger from Microsoft

<http://www.microsoft.com/downloads/info.aspx?na=22&p=1&SrcDisplayLang=en&SrcCategoryId=&SrcFamilyId=&u=%2fdownloads%2fdetails.aspx%3fFamilyID%3d2f465be0-94fd-4569-b3c4-dffdf19ccd99%26DisplayLang%3den>

In your client code write “debugger;” where you want to debug from, and choose the Script Debugger when the page is loaded. If installed it is also possible use Microsoft Visual Studio this way.

This will only work if you uncheck the checkbox Disable script debugging (Internet Explorer) in advanced options.

3.9 Query string API

This section describes the Process FrontEnd query string api.

A typical url with query string could look like this:

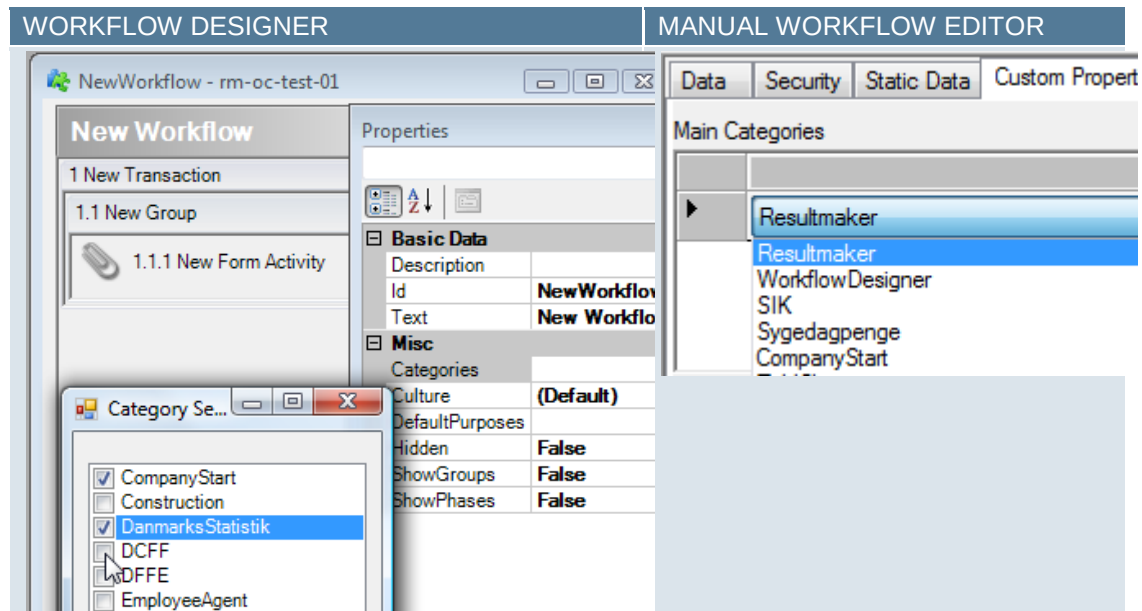
<http://ServerName/RMFrontEnd/Default.aspx?id=newworkflow>

3.9.1 ProjectTypeFilter

FrontEnd Version: 3.0

Description:

Setting the ProjectTypeFilter results in the project lists being filtered by the name that is specified in the value. This corresponds to the Categories set in the properties of the Workflow Designer in the Process Designer 3.3 and later. ProjectTypeFilter also corresponds to the Main Categories set in the custom properties in the Manual Workflow Editor.



Example:

<http://ServerName/RMFrontEnd/Default.aspx?ProjectTypeFilter=CompanyStart>

3.9.2 Destination

FrontEnd Version: 3.0 to 6.0.4000

Description:

The same as ProjectTypeFilter.

This parameter is deprecated but still supported until version 6.0.4000.

Example:

3.9.3 StartProject

FrontEnd Version: 3.0

Description:

This parameter will start a workflow from a workflow template with the name specified in the parameter. This will spawn a workflow instance.

Example:

<http://ServerName/RMFrontEnd/Default.aspx?StartProject=newworkflow>

3.9.4 Id

FrontEnd Version: 3.0

Description:

The same as StartProject.

Example:

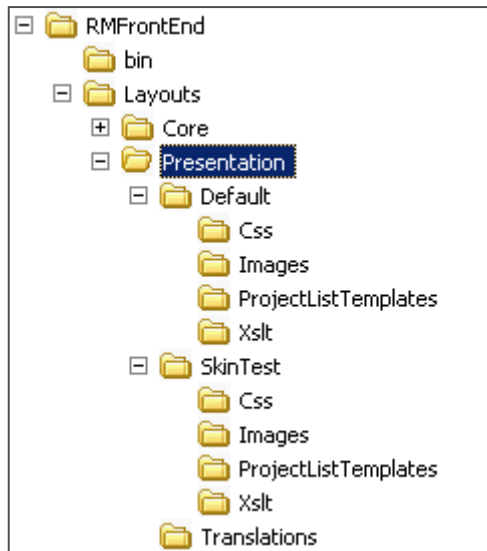
<http://ServerName/RMFrontEnd/Default.aspx?id=newworkflow>

3.9.5 SkinName

FrontEnd Version: 3.0

Description:

It is possible to define several skins. These skins are stored in the following folder structure.



The SkinName parameter overwrites the skin value set in the Resultmaker.OC.FrontEnd.config.

Example:

<https://ServerName/RMFrontEnd/Default.aspx?Skinname=SkinTest>

3.9.6 Language

FrontEnd Version: 6.0.5300

Description:

With this querystring parameter, it is possible to specify the language to use for the application texts. The language texts are taken from a file stored in the folder /Layouts/Presentation/Translations/ in the Frontend with the file name Translations.[Language].xml. The language parameter value must be the language ISO code i.e. en, ge, da, se etc.

Example:

<https://ServerName/RMFrontEnd/Default.aspx?Language=da>

3.9.7 ResumeProject

FrontEnd Version: 3.0

Description:

With this querystring parameter it is possible to resume a project/workflow with a given project id. The logged in user is validated for the proper permissions for the workflow.

Example:

<https://ServerName/RMFrontEnd/Default.aspx?ResumeProject=1001>

3.9.8 DefaultProjectListType

FrontEnd Version: 3.0

Description:

This parameter makes it possible by query string to access a specific type of Project list, New Projects, Running Projects, Completed Projects.

Valid options are "N" ("New"), "R" ("Running"), "C" ("Completed").

Note that "P" ("Pending") is deprecated in version 6.0.

Example:

<https://ServerName/RMFrontEnd/Default.aspx?DefaultProjectListType=R>

3.9.9 ProjectListType

FrontEnd Version: 3.0

Description:

Example:

3.9.10 ProjectListSortNew

FrontEnd Version: 3.0

Description:

Specifies how to sort the project list for new projects/workflows.

Sorting by multiple columns is possible by comma separation of the columns and descending sorting is possible.

Options: "CreatedDate desc,ProjectText desc, Authority desc"

Example:

<https://ServerName/RMFrontEnd/Default.aspx?ProjectListSortNew=ProjectText,Authority>

3.9.11 ProjectListSortRunning

FrontEnd Version: 3.0

Description:

Specifies how to sort the project list for started/running projects/workflows.

Sorting by multiple columns is possible by comma separation of the columns and descending sorting is possible.

Options: "CreatedDate desc,ProjectText desc, Authority desc"

Example:

<https://ServerName/RMFrontEnd/Default.aspx?ProjectListSortRunning=ProjectText,Authority>

3.9.12 ProjectListSortCompleted

FrontEnd Version: 3.0

Description:

Specifies how to sort the project list for finished projects/workflows.

Sorting by multiple columns is possible by comma separation of the columns and descending sorting is possible.

Options: "CreateDate desc,ProjectText,Authority"

Example:

<https://ServerName/RMFrontEnd/Default.aspx?ProjectListSortCompleted=ProjectText,Authority>

3.9.13 ProjectListSortPending

FrontEnd Version: 3.0. Deprecated in version 6.0

Description:

Options: DateBegin,Subject

Example:

3.9.14 ProjectListSortBy

FrontEnd Version: 3.0 to 6.0.4000

Description:

Deprecated as a querystring parameter.

Example:

3.9.15 ProjectListItemsPerPage

FrontEnd Version: 3.0

Description:

Specifies how many projects/workflows to display per page on the project list pages.

Example:

<https://ServerName/RMFrontEnd/Default.aspx?ProjectListItemsPerPage=80>

3.9.16 RM_EnableEnrichedWorkflow

FrontEnd Version: 3.0 to 5.2

Description:

Specifies whether to show the enriched workflow menu or not. Requires AllowEnrichedWorkflow to be set to 1.

Deprecated in version 6.0

Example:

3.9.17 RM_WorkflowOnly

FrontEnd Version: 3.0 to 5.2

Description:

Deprecated in version 6.0

Example:

3.9.18 ResumeCurrent

FrontEnd Version: 3.0

Description:

Specifies that the Frontend should resume the workflow based on the workflow ID stored in the ASP.NET session.

Example:

<https://ServerName/RMFrontEnd/Default.aspx?ResumeCurrent=1>

3.9.19 ResumeFromInvitation**FrontEnd Version:** 3.0**Description:**

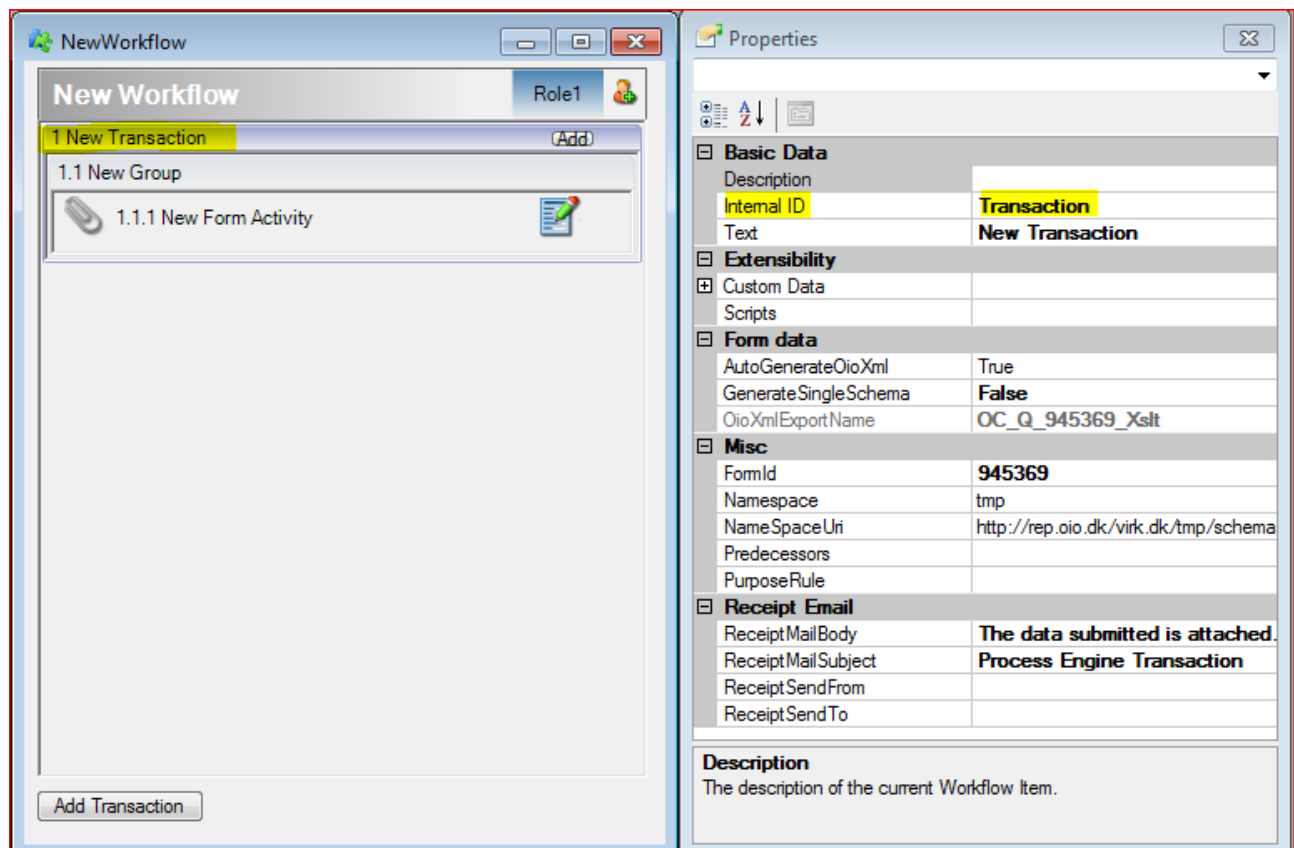
The ResumeFromInvitation functionality makes it possible to send a link to a third party user, and invite this user to a workflow instance in a specified workflow role. The workflow instance id, the role name to give the third party user access to is stored in the TokenService and a token to that information is issued in the form of a guid. This Tokenservice guid is the value of the ResumeFromInvitation query string parameter.

Example:

<https://ServerName/RMFrontEnd/Default.aspx?ResumeFromInvitation=4e0f09df-250c-4513-995a-61b6d51d5a39>

3.9.20 GoToActivity**FrontEnd Version:** 3.0**Description:**

Specify a path to the activity from the activities Internal ID. Activity levels are separated by a backslash "\". The Internal IDs can be found in the Process Designer.

**Figur 1** How to the the Internal ID from the Transaction level

GoToActivity=\Transaction\WorkflowGroup\FormActivity2

See the related JavaScript function in section 3.5.22.

Note that this query string parameter only works in conjunction with StartProject/ID or ResumeProject, ResumeCurrent or ResumeFromInvitation.

Example:

http://Server/RMFrontEnd/default.aspx?id=SoftwareTest_QueryString_GoToActivity&GoToActivity=\Transaction\Group\Act2

3.9.21 RedirectToSenderURL

FrontEnd Version: 3.0 to 5.2

Description:

Not supported in version 6.0.

Example:

3.9.22 RedirectToSenderText

FrontEnd Version: 3.0 to 5.2

Description:

Not supported in version 6.0.

Example:

3.9.23 ShowLogin

FrontEnd Version: 3.0 to 5.2

Description:

Setting this parameter to 1 will show the login page. Note that the parameter will not be removed, so that subsequent get requests to the same url, will show the login page again.

No longer supported in 6.0.

Example:

3.9.24 ShowLoginIf

FrontEnd Version: 3.0 to 5.2

Description:

Setting this parameter to 1 will show the login page if the user is not logged in.

This functionality requires that a login, username/password or certificate login, is enabled.

No longer supported in 6.0.

Example:

<http://server/RMFrontEnd/default.aspx?ShowLoginIf=1>

4 Process Frontend User Object Documentation

This section describes the Process Frontend user object interface and its implementations.

The user object has to be provided by the platform to the Process Frontend during log-on and will be accessible within the running session. The object hides actual implementation of user handling on the platform, which thereby has to be implemented by the platform provider.

4.1 User object interface

The section describes the interface `ILoginInformation` for the Resultmaker user object.

`ILoginInformation` description

PROPERTY NAME	TYPE	DESCRIPTION
CommonName	String	Returns the friendly user name (as friendly as possible, for displaying purposes). Goes to "LoggedInCommonName" user variable. Examples: "Børge Åge Bæntsen" or "BAB".
Email	String	Returns the users email.
IsLoggedIn	Boolean	Returns true if the user is logged in.
IsRequireLoggedout	Boolean	Returns true if the user is require to log out. Some implementations may use {false} to mark the object as "not-logged-in" (to invalidate it).
SecurityToken	String	Returns the security token used to communicate with the Process Engine Security Provider. Traditionally goes to the "UserUmsID" user variable. Example: "WinDelegation:OUTOFBANDPAYLOAD==" for the IIS integrated Windows security provider.
UserCode	string	Returns the user unique code. /// If the implementation has no good candidate for the unique code, it may be the string representation of UserGUID. /// Example: "CVR:25369580-RID:1088688576960" (for certificate login); "BAB" (for user/password login).

5 Project Properties

Project properties are the properties a project has on different levels. They are the properties, which can be edited in the Process Designer.

5.1 Project Level

5.1.1 DefaultPurposes

Default purposes are a list of purposes, which are evaluated at project start time. It is possible to use several purposes separated by commas.

5.1.2 Purpose rule

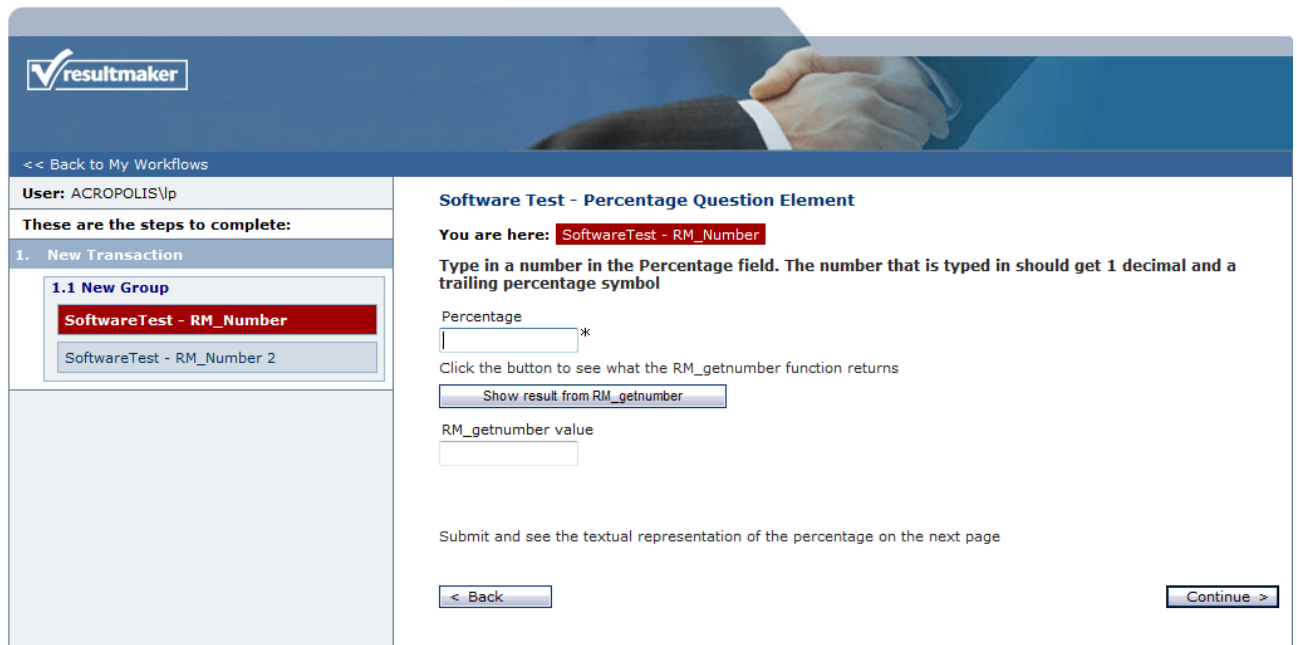
The purpose rule property determines the conditions under which a work flow item is presented or hidden in the flow, based on the runtime purposes set and the value of the purpose rule property.

The purposes can be AND, OR, NOT. The default value is OR. Advanced purpose rules can be constructed such as:

Not (PurposeA AND PurposeB OR (PurposeB AND PurposeC)).

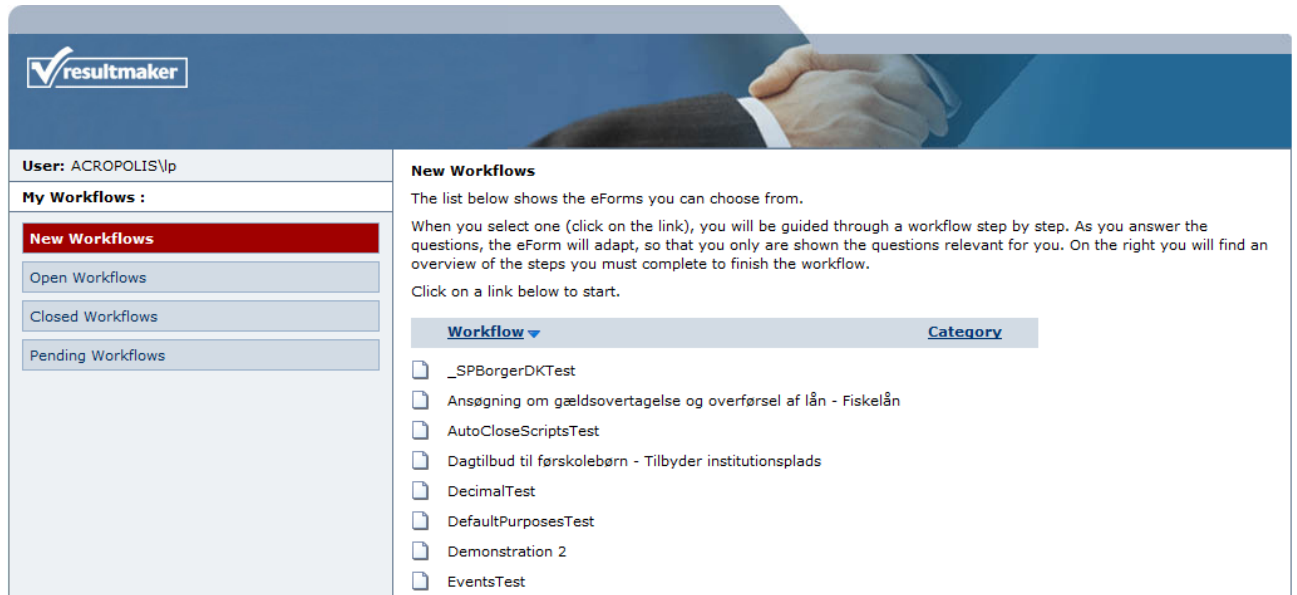
6 Project list layout

The Resultmaker Process Frontend consists of a series of different views. The main view is the workflow view where the workflow execution is rendered. Another series of important pages are the workflow/project list pages. This section describes the format of the Frontend Project List XML file and how to change the layout of the project list pages.



The screenshot displays the Resultmaker Process Frontend interface. On the left, a sidebar shows the navigation structure: 'User: ACROPOLIS\lp', 'These are the steps to complete:', '1. New Transaction', and '1.1 New Group'. Under '1.1 New Group', two items are listed: 'SoftwareTest - RM_Number' (highlighted in red) and 'SoftwareTest - RM_Number 2'. The main content area is titled 'Software Test - Percentage Question Element'. It includes a breadcrumb 'You are here: SoftwareTest - RM_Number', instructions to 'Type in a number in the Percentage field. The number that is typed in should get 1 decimal and a trailing percentage symbol', a 'Percentage' input field with an asterisk, a 'Show result from RM_getnumber' button, an 'RM_getnumber value' input field, and a 'Continue >' button at the bottom right. A '< Back' button is also present at the bottom left of the main area.

Figure 2 Main view: Workflow View



User: ACROPOLIS\lp

My Workflows :

- New Workflows**
- Open Workflows
- Closed Workflows
- Pending Workflows

New Workflows

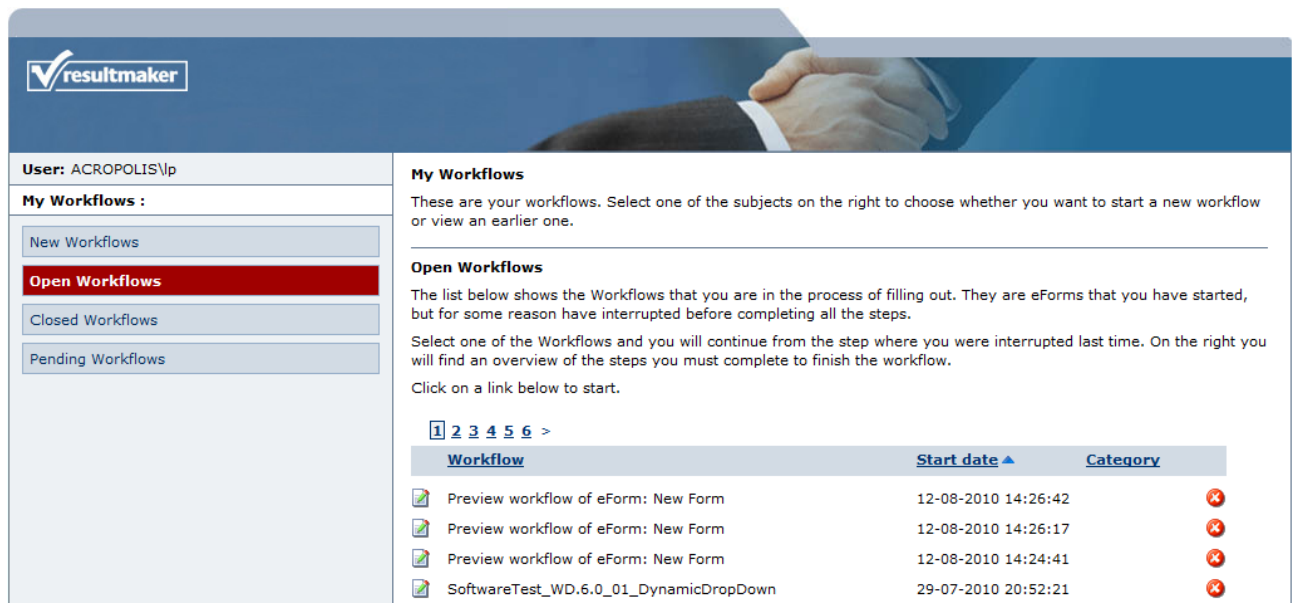
The list below shows the eForms you can choose from.

When you select one (click on the link), you will be guided through a workflow step by step. As you answer the questions, the eForm will adapt, so that you only are shown the questions relevant for you. On the right you will find an overview of the steps you must complete to finish the workflow.

Click on a link below to start.

Workflow	Category
_SPBorgerDKTest	
Ansøgning om gældsovertagelse og overførsel af lån - Fiskelån	
AutoCloseScriptsTest	
Dagtilbud til førskolebørn - Tilbyder institutionsplads	
DecimalTest	
DefaultPurposesTest	
Demonstration 2	
EventsTest	

Figure 3 Workflow list view: New Workflows



User: ACROPOLIS\lp

My Workflows :

- New Workflows
- Open Workflows**
- Closed Workflows
- Pending Workflows

Open Workflows

These are your workflows. Select one of the subjects on the right to choose whether you want to start a new workflow or view an earlier one.

Open Workflows

The list below shows the Workflows that you are in the process of filling out. They are eForms that you have started, but for some reason have interrupted before completing all the steps.

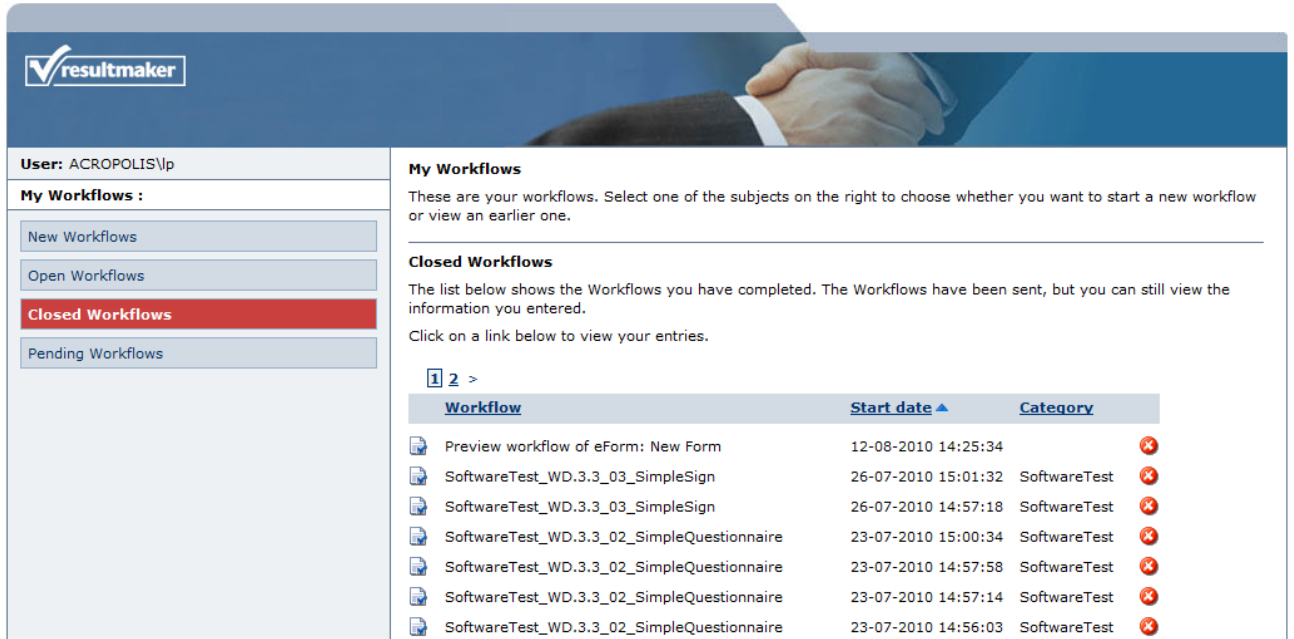
Select one of the Workflows and you will continue from the step where you were interrupted last time. On the right you will find an overview of the steps you must complete to finish the workflow.

Click on a link below to start.

1 2 3 4 5 6 >

Workflow	Start date	Category
Preview workflow of eForm: New Form	12-08-2010 14:26:42	
Preview workflow of eForm: New Form	12-08-2010 14:26:17	
Preview workflow of eForm: New Form	12-08-2010 14:24:41	
SoftwareTest_WD.6.0_01_DynamicDropDown	29-07-2010 20:52:21	

Figure 4 Workflow list view: Open Workflows



User: ACROPOLIS\vp

My Workflows :

- New Workflows
- Open Workflows
- Closed Workflows**
- Pending Workflows

My Workflows

These are your workflows. Select one of the subjects on the right to choose whether you want to start a new workflow or view an earlier one.

Closed Workflows

The list below shows the Workflows you have completed. The Workflows have been sent, but you can still view the information you entered.

Click on a link below to view your entries.

1 2 >

Workflow	Start date ▲	Category
Preview workflow of eForm: New Form	12-08-2010 14:25:34	
SoftwareTest_WD.3.3_03_SimpleSign	26-07-2010 15:01:32	SoftwareTest
SoftwareTest_WD.3.3_03_SimpleSign	26-07-2010 14:57:18	SoftwareTest
SoftwareTest_WD.3.3_02_SimpleQuestionnaire	23-07-2010 15:00:34	SoftwareTest
SoftwareTest_WD.3.3_02_SimpleQuestionnaire	23-07-2010 14:57:58	SoftwareTest
SoftwareTest_WD.3.3_02_SimpleQuestionnaire	23-07-2010 14:57:14	SoftwareTest
SoftwareTest_WD.3.3_02_SimpleQuestionnaire	23-07-2010 14:56:03	SoftwareTest

Figure 5 Workflow list view: Closed Workflows

The Workflow list view in Figure 2-4 can be controlled by the file *DefaultProjectListTemplate.xml*.

This file is located in the folder `\Layouts\Presentation\Default\ProjectListTemplates\` in the RMFrontEnd virtual folder.

The typical full path the the file is:

`C:\inetpub\wwwroot\RMFrontEnd\Layouts\Presentation\Default\ProjectListTemplates\DefaultProjectListTemplate.xml`

The file has the following basic structure.

```
- <ProjectListTemplate Name="Default" LeftHeading="Indberetninger">
  + <Menu>
  + <Menu LoggedIn="True">
  + <MainText>
  + <MainText LoggedIn="True">
  + <Projects List="New">
  + <Projects List="New" LoggedIn="True">
  + <Projects List="Running" LoggedIn="True">
  + <Projects List="Completed" LoggedIn="True">
</ProjectListTemplate>
```

Out of the box, the full file looks like this:

```
<?xml version="1.0" encoding="utf-8" ?>
<ProjectListTemplate Name="Default" LeftHeading="Workflows">
  <Menu>
    <Item Name="ListNew" OText="NewProjects"
    ODescription="Select_a_new_project_" />
    <Item Name="Current" OText="ActiveProject"
    ODescription="Select_your_active_project_" />
    <Item Name="ListRunning" OText="RunningProjects"
    ODescription="ProjectsAvailableAfterLogin" Disabled="True" />
    <Item Name="ListCompleted" OText="ClosedProjects"
    ODescription="ProjectsAvailableAfterLogin" Disabled="True" />
  </Menu>
  <Menu LoggedIn="True">
    <Item Name="ListNew" OText="NewProjects"
    ODescription="Select a new project " />
```

```

    <Item Name="Current"                OText="ActiveProject"
        OCDescription="Select_your_active_project_" />
    <Item Name="ListRunning"            OText="RunningProjects"
        OCDescription="Select_one_of_your_saved_running_projects_" />
    <Item Name="ListCompleted"          OText="ClosedProjects"
        OCDescription="Select_one_of_your_saved_closed_projects_" />
</Menu>

<MainText>
    <Insert Name="Projects" Value="New" />
</MainText>
<MainText LoggedIn="True">
    <Insert Name="Projects" Value="New" />
    <Insert Name="Projects" Value="Running" />
    <Insert Name="Projects" Value="Completed" />
</MainText>

<Projects List="New">
    <div class="RM_ProjectCompletionBlock">
        <div class="RM_ProjectCompletionSubHeader">
            <OText Name="NewProjects" disable-output-escaping="yes" />
        </div>
        <div class="RM_ProjectCompletionText">
            <OText Name="ProjectList.New.Text" disable-output-escaping="yes" />
        </div>
        <div class="RM_ProjectCompletionSubHeader">
            <OText Name="ProjectList.NotLoggedIn.Header" disable-output-escaping="yes"
/>
        </div>
        <div class="RM_ProjectCompletionText">
            <OText Name="ProjectList.NotLoggedIn.Note" disable-output-escaping="yes" />
        </div>
        <div class="RM_ProjectCompletionText">
            <div>
                <Insert Name="LoginURL" OText="ProjectList.LoginLink.1" />&#32;<OText
Name="ProjectList.LoginLink.2" disable-output-escaping="yes" />
            </div>
            <div>
                <OText Name="ProjectList.OrderSignature" disable-output-escaping="yes" />
            </div>
        </div>
        <table>
            <InsertList>
                <tr>
                    <td>
                        <Insert Name="Bullet" />
                    </td>
                    <td>
                        <Insert Name="ProjectText" />
                    </td>
                    <td>
                        <Insert Name="Authority" />
                    </td>
                </tr>
            </InsertList>
        </table>
    </div>
</Projects>

<Projects List="New" LoggedIn="True">
    <div class="RM_ProjectCompletionBlock">
        <div class="RM_ProjectCompletionSubHeader">
            <OText Name="NewProjects" disable-output-escaping="yes" />
        </div>
        <div class="RM_ProjectCompletionText">
            <OText Name="ProjectList.New.Text" disable-output-escaping="yes" />
        </div>
        <table>
            <InsertList>
                <tr>
                    <td>
                        <Insert Name="Bullet" />
                    </td>
                    <td>
                        <Insert Name="ProjectText" />
                    </td>
                    <td>

```

```

        <Insert Name="Authority" />
      </td>
    </tr>
  </InsertList>
</table>
</div>
</Projects>

<Projects List="Running" LoggedIn="True">
  <div class="RM_ProjectCompletionBlock">
    <div class="RM_ProjectCompletionSubHeader">
      <OCText Name="ProjectList.MyProjects" disable-output-escaping="yes" />
    </div>
    <div class="RM_ProjectCompletionText">
      <OCText Name="ProjectList.LoggedIn.Note" disable-output-escaping="yes" />
    </div>
    <hr />
    <div class="RM_ProjectCompletionSubHeader">
      <OCText Name="RunningProjects" disable-output-escaping="yes" />
    </div>
    <div class="RM_ProjectCompletionText">
      <OCText Name="ProjectList.Running.Text" disable-output-escaping="yes" />
    </div>
    <table>
      <InsertList>
        <tr>
          <td>
            <Insert Name="Bullet" />
          </td>
          <td>
            <Insert Name="ProjectText" />
          </td>
          <td>
            <Insert Name="CreatedDate" />
          </td>
          <td>
            <Insert Name="Authority" />
          </td>
          <td>
            <Insert Name="Delete" />
          </td>
        </tr>
      </InsertList>
    </table>
  </div>
</Projects>

<Projects List="Completed" LoggedIn="True">
  <div class="RM_ProjectCompletionBlock">
    <div class="RM_ProjectCompletionSubHeader">
      <OCText Name="ProjectList.MyProjects" disable-output-escaping="yes" />
    </div>
    <div class="RM_ProjectCompletionText">
      <OCText Name="ProjectList.LoggedIn.Note" disable-output-escaping="yes" />
    </div>
    <hr />
    <div class="RM_ProjectCompletionSubHeader">
      <OCText Name="ClosedProjects" disable-output-escaping="yes" />
    </div>
    <div class="RM_ProjectCompletionText">
      <OCText Name="ProjectList.Completed.Text" disable-output-escaping="yes" />
    </div>
    <table>
      <InsertList>
        <tr>
          <td>
            <Insert Name="Bullet" />
          </td>
          <td>
            <Insert Name="ProjectText" />
          </td>
          <td>
            <Insert Name="CreatedDate" />
          </td>
          <td>
            <Insert Name="Authority" />
          </td>
          <td>
            <Insert Name="Delete" />
          </td>
        </tr>
      </InsertList>
    </table>
  </div>
</Projects>

```



```

        <td>
          <Insert Name="Delete" />
        </td>
      </tr>
    </InsertList>
  </table>
</div>
</Projects>
</ProjectListTemplate>

```

With this file it is possible to control how the workflow list views are rendered. There are the following different views that can be rendered depending on whether the user is logged in or not:

- <Projects List="New">
- <Projects List="New" LoggedIn="True">
- <Projects List="Running" LoggedIn="True">
- <Projects List="Completed" LoggedIn="True">

6.1 Running and Completed projects

The Running and Completed projects type definitions are explained here.

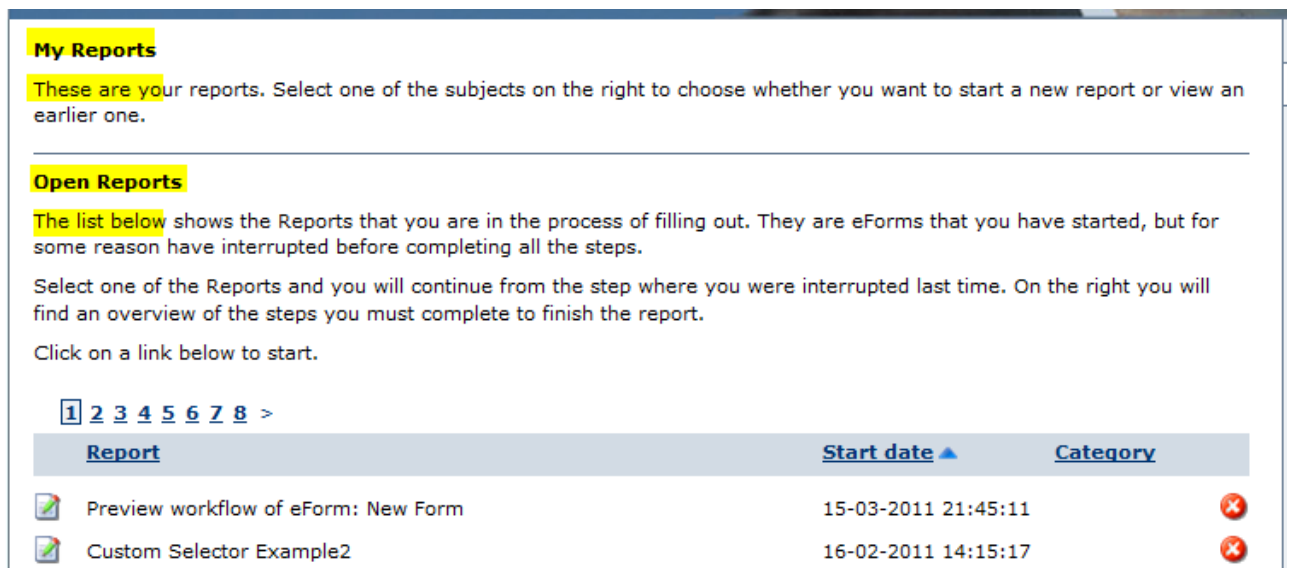
The Div in the first part references Translations from the translation file

```

<div class="RM_ProjectCompletionSubHeader">
  <OCText Name="ProjectList.MyProjects" disable-output-escaping="yes" />
</div>
<div class="RM_ProjectCompletionText">
  <OCText Name="ProjectList.LoggedIn.Note" disable-output-escaping="yes" />
</div>
<hr />
<div class="RM_ProjectCompletionSubHeader">
  <OCText Name="RunningProjects" disable-output-escaping="yes" />
</div>
<div class="RM_ProjectCompletionText">
  <OCText Name="ProjectList.Running.Text" disable-output-escaping="yes" />
</div>

```

These four OCText elements references translations and the are presented in the places shown in this example, marked with yellow.



My Reports

These are your reports. Select one of the subjects on the right to choose whether you want to start a new report or view an earlier one.

Open Reports

The list below shows the Reports that you are in the process of filling out. They are eForms that you have started, but for some reason have interrupted before completing all the steps.

Select one of the Reports and you will continue from the step where you were interrupted last time. On the right you will find an overview of the steps you must complete to finish the report.

Click on a link below to start.

1 2 3 4 5 6 7 8 >

Report	Start date ▲	Category
 Preview workflow of eForm: New Form	15-03-2011 21:45:11	
 Custom Selector Example2	16-02-2011 14:15:17	

The next section, the table and InsertList elements define the columns for the actual project list. The values in the Name attribute are reserved words. It is possible to insert the text elements

ProjectText and CreatedDate from the project table of the Process Platform. It is also possible to insert bullets and a delete button. And it is possible to remove these as they are merged with normal html.

```
<table>
  <InsertList>
    <tr>
      <td>
        <Insert Name="Bullet" />
      </td>
      <td>
        <Insert Name="ProjectText" />
      </td>
      <td>
        <Insert Name="CreatedDate" />
      </td>
      <td>
        <Insert Name="Authority" />
      </td>
      <td>
        <Insert Name="Delete" />
      </td>
    </tr>
  </InsertList>
</table>
</div>
</Projects>
```

7 Custom Properties

Custom Properties are internal properties on the Process Platform workflow object. They can reside on all levels of the workflow object. It is possible to modify a selected range of Custom Properties in the Advanced Workflow Editor. The editing of these properties have been discontinued in the Workflow Designer.

In the following sections the custom properties adhering to the different workflow levels are described.

7.1 Project Level

7.1.1 projectlistvariable

Custom Property: projectlistvariable

FrontEnd Version: 2.3

Process Designer Version: Not possible to modify in any designer

Description: The projectlistvariable Custom Property is created in order to convey non-static information from the back end to the Process Platform project lists. The Custom Property can be set in Process Engine back end scripts and can be presented on the project list page. Several Custom Properties can be set, so the form an array. The are distinguished on the Key attribute

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Name</i>	No	projectlistvariable	
<i>Value</i>	No	The name to be displayed on the menu heading.	Provides the text of the menu heading.
<i>Key</i>	No	The name of the key distinguishing between the different projectlistvariables	

Example:

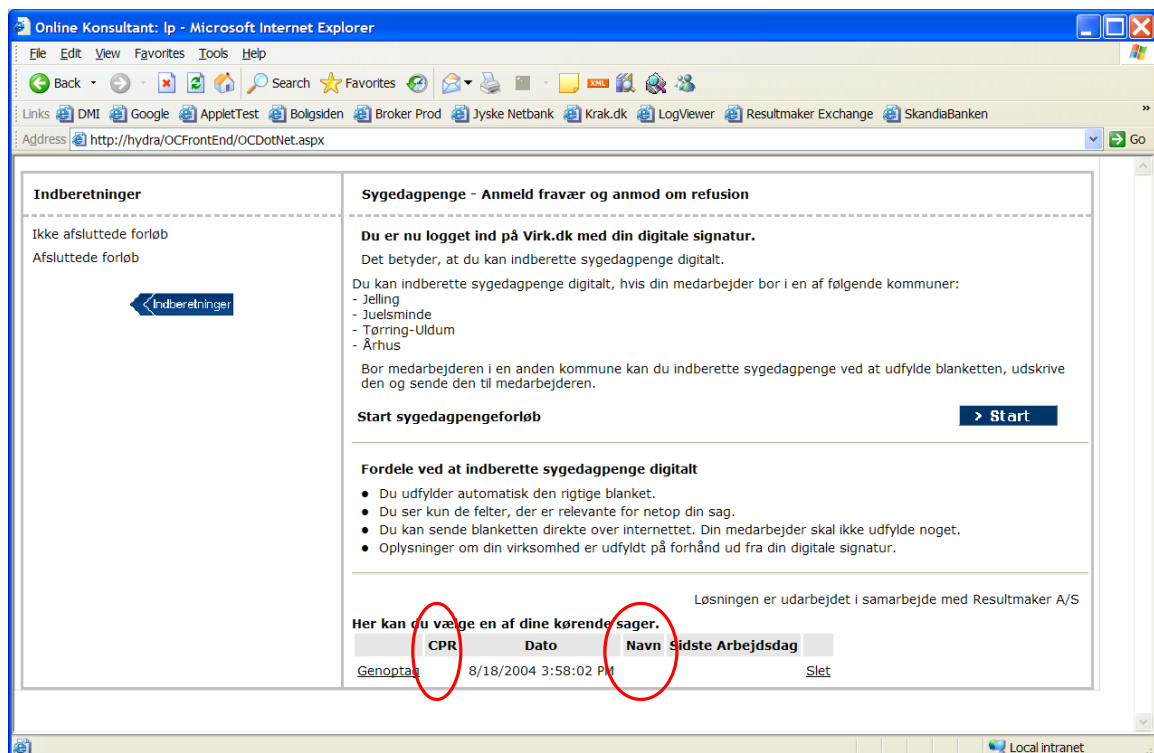


Figure 1: Project list page (not) showing the value from Custom Property

The Custom Properties have to be set like this

```
<customproperties>
```

```
<customproperty name="projectlistvariable"
key="kl:EmployeePersonDataGroup[1].cpr:CivilRegistrationNumber" value ="0211721111"
/>
<customproperty name="projectlistvariable"
key="kl:EmployeePersonDataGroup[1].dkcc:PersonNameGroup[1].dkcc:PersonGivenName"
value ="Lars Pedersen" />
</customproperties>
```

In the project list template insert the following to use the above custom properties.

```
<Insert Name="Variable"
Value="kl:EmployeePersonDataGroup[1].cpr:CivilRegistrationNumber" />

<Insert Name="Variable"
Value="kl:EmployeePersonDataGroup[1].dkcc:PersonNameGroup[1].dkcc:PersonGivenName" />
```

The CustomProperties are set in the following way from the Process Engine backend script:

```
'-----
'   Insert a new Custom Property
'-----
option explicit

dim cp, projectListVariables, projectVariable, cpXml

' create the object.
set cp = CreateObject(
"Resultmaker.OC.Content.CustomProperties.ProjectCustomProperties" )

' add some data
'cpXml = "<customproperties/>"
cpXml = Project.CustomProperties
cp.LoadXml( cpXml )

' get a list of variables.
set projectListVariables = cp.ProjectListVariables

' create a new variable.
set projectVariable = CreateObject(
"Resultmaker.OC.Content.CustomProperties.KeyValueCustomProperty" )

' fill the variable with data.
projectVariable.Key = "fred"
projectVariable.Value = "freddy"

' add variable to customproperties.
Set projectVariable = projectListVariables.AddByValue( projectVariable )

Project.CustomProperties = cp.ToString()

'-----
'   Edit an existing Custom Property
'-----
option explicit

dim cp, projectListVariables, projectVariable, cpXml

' create the object.
set cp = CreateObject(
"Resultmaker.OC.Content.CustomProperties.ProjectCustomProperties" )

' add some data
cpXml = Project.CustomProperties
cp.LoadXml( cpXml )

' get a list of variables.
set projectListVariables = cp.ProjectListVariables
```

```

' get the variable.
set projectVariable = projectListVariables( "fred" )

' change the values.
projectVariable.Key = "fred2"
projectVariable.Value = "freddier"

Project.CustomProperties = cp.ToString()

'-----
'  REMOVE an existing Custom Property
'-----
option explicit

dim cp, projectListVariables, projectVariable, cpXml

' create the object.
set cp = CreateObject(
"Resultmaker.OC.Content.CustomProperties.ProjectCustomProperties" )

' add some data
cpXml = Project.CustomProperties
cp.LoadXml( cpXml )

' get a list of variables.
set projectListVariables = cp.ProjectListVariables

' get the variable.
set projectVariable = projectListVariables( "fred" )

' remove variable.
projectListVariables.Remove(projectVariable)

Project.CustomProperties = cp.ToString()

'-----
'  LOOP through the collection
'-----

option explicit
dim cp, projectListVariables, projectVariable

set cp = CreateObject(
"Resultmaker.OC.Content.CustomProperties.ProjectCustomProperties" )

cp.LoadXml( Project.CustomProperties )

for each projectVariable in projectListVariables
    MsgBox "name=" & projectVariable.Name & " key=" & projectVariable.Key &
" value=" & projectVariable.Value
next

```

7.1.2 showphases

Custom Property: *showphases*

FrontEnd Version: 2.0 – 6.0

Process Designer Version: 3.3 – 6.0 in both Workflow Designer and Advanced Workflow Designer

Description: Indicates to the FrontEnd whether to show or hide all the phases in the menu. This can be overridden by the *showphase* Custom Property on the individual phase.

This CustomProperty is valid on the project level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	true false	Indicates to the FrontEnd whether to show or hide all the phases in the menu

Example:

```
<customproperties>
  <customproperty name="showphases" value="true" />
</customproperties>
```

7.1.3 showactivitysets

Custom Property: *showactivitysets*

FrontEnd Version: 2.0 – 6.0

Process Designer Version: 3.3 – 6.0 in both Workflow Designer and Advanced Workflow Designer

Description: Indicates to the Process Frontend whether to show or hide all the activitysets in the menu. This can be overridden by the *showactivityset* Custom Property on the individual activityset.

This CustomProperty is valid on the project level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	true false	Indicates to the FrontEnd whether to show or hide all the activitysets in the menu

Example:

```
<customproperties>
  <customproperty name="showactivitysets" value="true" />
</customproperties>
```

7.1.4 menulabel

Custom Property: *menulabel*

FrontEnd Version: 1.7.1

Process Designer Version: 3.3 – 6.0 in Advanced Workflow Designer. Not possible with Workflow Designer.

Description: Sets the text displayed on the menu if differs from the project text.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	The name to be displayed on the menu heading.	Provides the text of the menu heading.

Example:

```
<customproperties>
  <customproperty name="menulabel" value="My Project menu heading" />
</customproperties>
```

7.1.5 waittext

Custom Property: *waittext*

FrontEnd Version: 1.8.0

Process Designer Version: 3.3 – 6.0 in Advanced Workflow Designer

Description: Sets the text or HTML displayed when the continue button is pressed.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	The text or HTML to be displayed.	Provides the text or HTML displayed when the continue button is pressed.

Example:

```
<customproperties>
  <customproperty name="waittext" value="Please wait while the robots take a long
time doing whatever it is robots do." />
</customproperties>
```

7.1.6 exitlink

Custom Property: *exitlink*

FrontEnd Version: 2.3

Process Designer Version: 3.3 – 6.0 in both Workflow Designer and Advanced Workflow Designer

Description: The exitlink custom property overrides the “back to project list” link on the projects.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>text</i>	Yes	A string	The text to display in the exit link
<i>url</i>	Yes	A url	The url to link to

Example:

```
<customproperties>
  <customproperty name="exitlink" text="Back to start" url="http://somewhere/else" />
</customproperties>
```

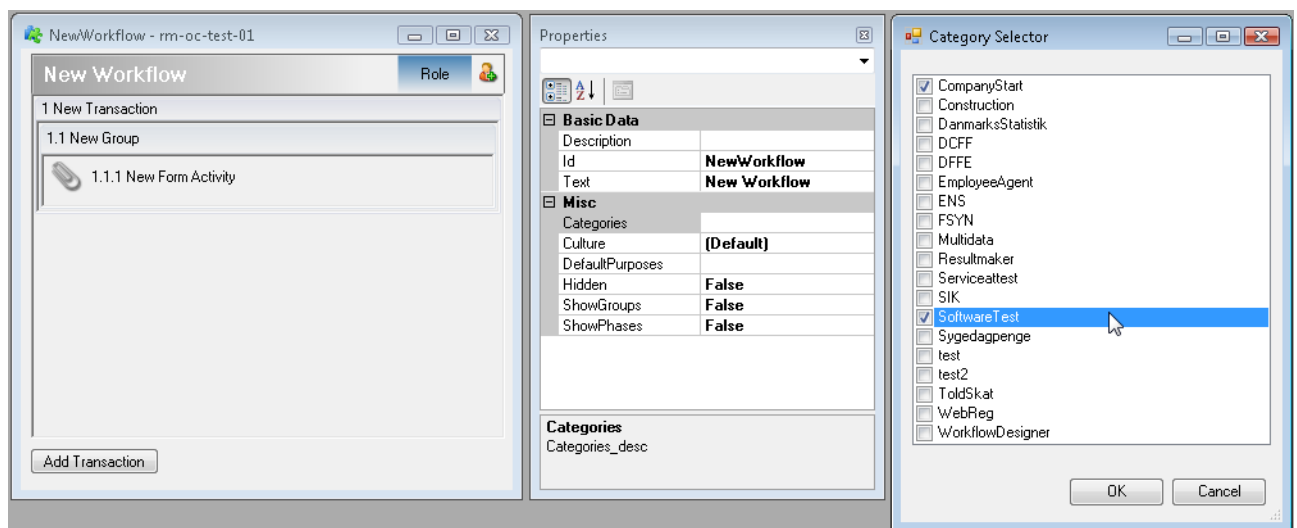
7.1.7 projecttype|category

Custom Property: *projecttype*

FrontEnd Version: 1.6 – 6.0

Process Designer Version: 3.3 – 6.0 in both Workflow Designer and Advanced Workflow Designer

Description: In order to be able to make different project lists, the lists have to be retrieved based on extra information from the project. This information is placed in CustomProperties at the project level under the name projecttype. The value is arbitrary. It is possible to set several project type filters. In Workflow Designer the project property is called Category.



Figur 6Workflow Designer with the property grid and the Category Selector

Attributes: none.

Sub-tags: *insert*

Example:

```
<customproperties>
  <customproperty name="projecttype">
    SoftwareTest
  </customproperty>
</customproperties>
```

A project can have several project types but they have to be placed in separate CustomProperties:

```
<customproperty name="projecttype">
  SoftwareTest
</customproperty>
<customproperty name="projecttype">
  CompanyStart
</customproperty>
```

To use this CustomProperty functionality, the Frontend has to add an extra argument to the command, which retrieves the project lists. Please note the casing.

A normal command retrieving the full project list looks like this:

```
<Command Name="GetProjectList2">
  <Argument Name="UserName">Virk<Argument>
  <Argument Name="UserToken">{o10710-m1ki-k3sy}<Argument>
  <Argument Name="LogonType">Anonymous<Argument>
</Command>
```

The limited project list command looks like this:

```
<Command Name="GetProjectList2">
  <Argument Name="UserName">Virk<Argument>
  <Argument Name="UserToken">{o10710-m1ki-k3sy}<Argument>
  <Argument Name="LogonType">Anonymous<Argument>
  <Argument Name="ProjectTypes">SoftwareTest<Argument>
</Command>
```

The value of the ProjectTypes argument can be a single word or a comma separated list (CompanyStart,SoftwareTest).

The current list of Categories is fetched from the Process Engine web service, which takes the data from the ProjectTypes.xml file located in the folder
c:\FileRepository\private\AccessConfigurationFiles\

The file can be edited to present the necessary categories.

The default installed files looks like this:

```
<?xml version="1.0" encoding="utf-8" ?>
<ProjectTypes>
  <ProjectType>Resultmaker</ProjectType>
  <ProjectType>WorkflowDesigner</ProjectType>
  <ProjectType>SIK</ProjectType>
  <ProjectType>Sygedagpenge</ProjectType>
  <ProjectType>CompanyStart</ProjectType>
  <ProjectType>ToldSkat</ProjectType>
  <ProjectType>FSYN</ProjectType>
  <ProjectType>Construction</ProjectType>
  <ProjectType>Serviceattest</ProjectType>
  <ProjectType>DanmarksStatistik</ProjectType>
  <ProjectType>WebReg</ProjectType>
  <ProjectType>Multidata</ProjectType>
  <ProjectType>EmployeeAgent</ProjectType>
  <ProjectType>DFFE</ProjectType>
  <ProjectType>DCFF</ProjectType>
  <ProjectType>ENS</ProjectType>
  <ProjectType>test</ProjectType>
  <ProjectType>test2</ProjectType>
  <ProjectType>SoftwareTest</ProjectType>
</ProjectTypes>
```

If the files is changed, the keep a backup as software updates can overwrite the file.

7.1.8 projectstartpage

Custom Property: *projectstartpage*

FrontEnd Version: 1.6

Description:

Attributes: none.

Sub-tags: *insert*

Example:

```
<customproperties>
  <customproperty name="projectstartpage"><table><tr><td><insert name = "welcometext"
/></td></tr>
<tr><td>You have now finished the project. Bla, bla, bla. </td></tr></table></customproperty>
</customproperties>
```

7.1.8.1 insert

Custom Property: *insert*

FrontEnd Version: 1.6

Description:

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>name</i>	No	"welcometext" (inserts the welcome text from the welcome text field in the project)	The name of the information you would like to insert.

Example:

7.1.9 projectendpage

Custom Property: *projectendpage*

FrontEnd Version: 1.6

Description:

Attributes: none.

Example:

```
<customproperties>
  <customproperty name="projectendpage"><table><tr><td>You have now finished the
project. Bla, bla, bla. </td></tr></table></customproperty>
</customproperties>
```

7.1.10 projectheader

Custom Property: *projectheader*

FrontEnd Version: 1.6

Description:

Attributes: none.

Example:

```
<customproperties>
  <customproperty name="projectheader"><table><tr><td>This is project is brought to
you by Resultmaker </td></tr></table></customproperty>
</customproperties>
```

7.1.11 relativenavigation

Custom Property: *relativenavigation*

FrontEnd Version: 2.5

Description: Causes the user to move to the following activity instead of the next activity when pressing the "Continue" button.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	true false	Indicates whether to use the relative navigation.

Example:

```
<customproperties>
  <customproperty name="relativenavigation" value="true" />
</customproperties>
```

7.1.12 backbuttontext

Custom Property: *backbuttontext*

FrontEnd Version: 2.11.1

Description: Defines the custom text as a caption for the "Back" button. It may be overwritten by the same custom property on Page or Questionnaire level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	Any text	The text to use.

Example:

```
<customproperties>
  <customproperty name="backbuttontext" value="Previous page" />
</customproperties>
```

7.1.13 continuebuttontext

Custom Property: *continuebuttontext*

FrontEnd Version: 2.11.1

Description: Defines the custom text as a caption for the "Continue" button. It may be overwritten by the same custom property on Page or Questionnaire level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	Any text	The text to use.

Example:

```
<customproperties>
  <customproperty name="continuebuttontext" value="Next page" />
</customproperties>
```

7.2 Phase Level

7.2.1 showphase

Custom Property: *showphase*

FrontEnd Version: 2.0

Description: Indicates to the FrontEnd whether to show or hide the current phase in the menu. This can override the *showphases* Custom Property on the project level.

This CustomProperty is valid on the phase level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	true false	Indicates to the FrontEnd whether to show or hide the current phase in the menu

Example:

```
<customproperties>
  <customproperty name="showphase" value="true" />
</customproperties>
```

7.3 ActivitySet Level

7.3.1 showactivityset

Custom Property: *showactivityset*

FrontEnd Version: 2.0

Description: Indicates to the FrontEnd whether to show or hide the current activityset in the menu. This can override the *showactivitysets* Custom Property on the project level.

This CustomProperty is valid on the activitset level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	true false	Indicates to the FrontEnd whether to show or hide the current activityset in the menu

Example:

```
<customproperties>
  <customproperty name="showactivityset" value="true" />
</customproperties>
```

7.3.2 Predecessor

Custom Property:

FrontEnd Version:

Description: An activity (activity set, activity) can take an earlier activity on same level as predecessor, which mean that the activity cannot start until the earlier activity has completed.

For a predecessor relation it is possible to set NestedReset, which means the an activity is reset if its predecessor is reset.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Nested reset</i>	Yes	true false	Indicates to the FrontEnd whether to show or hide the current activityset in the menu

7.4 Activity Level

7.4.1 waittext

Custom Property: waittext

FrontEnd Version: 1.8.0

Description: Sets the text or HTML displayed when the continue button is pressed. This overrides the Project level waittext custom property.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	The text or HTML to be displayed.	Provides the text or HTML displayed when the continue button is pressed.

Example:

```
<customproperties>
  <customproperty name="waittext" value="Please wait while the robots take a long
time doing whatever it is robots do." />
</customproperties>
```

7.5 Questionnaire Level

7.5.1 helpid

Custom Property: *helpid*

FrontEnd Version: 1.6

Description: The helpid custom property is used to identify the html filename that is to be used as the help file for the questionnaire

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>value</i>	no	Name of the html file that contains the help text.	

Example:

```
<customproperties>
  <customproperty name="helpid" value="H73282" />
</customproperties>
```

7.5.2 exitlink

Custom Property: *exitlink*

FrontEnd Version: 2.3

Description: The exitlink custom property overrides the “back to project list” link on the projects.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>text</i>	Yes	A string	The text to display in the exit link
<i>url</i>	Yes	A url	The url to link to

Example:

```
<customproperties>
  <customproperty name="exitlink" text="Back to start" url="http://somewhere/else" />
</customproperties>
```

7.5.3 backbuttontext

Custom Property: *backbuttontext*

FrontEnd Version: 2.11.1

Description: Defines the custom text as a caption for the "Back" button. If existing and not empty, it has precedence over the same custom property on Project level. It may be overwritten by the same custom property on Page level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	Any text	The text to use.

Example:

```
<customproperties>
  <customproperty name="backbuttontext" value="Previous page" />
</customproperties>
```

7.5.4 continuebuttontext

Custom Property: *continuebuttontext*

FrontEnd Version: 2.11.1

Description: Defines the custom text as a caption for the "Continue" button. If existing and not empty, it has precedence over the same custom property on Project level. It may be overwritten by the same custom property on Page level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	Any text	The text to use.

Example:

```
<customproperties>
  <customproperty name="continuebuttontext" value="Next page" />
</customproperties>
```

7.6 Page Level

7.6.1 waittext

Custom Property: waittext

FrontEnd Version: 1.8.0

Description: Sets the text or HTML displayed when the continue button is pressed. This overrides the Project level waittext and Activity level waittext custom properties.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	The text or HTML to be displayed.	Provides the text or HTML displayed when the continue button is pressed.

Example:

```
<customproperties>
  <customproperty name="waittext" value="Please wait while the robots take a long
time doing whatever it is robots do." />
</customproperties>
```

7.6.2 exitlink

Custom Property: exitlink

FrontEnd Version: 2.3

Description: The exitlink custom property overrides the “back to project list” link on the projects.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>text</i>	Yes	A string	The text to display in the exit link
<i>url</i>	Yes	A url	The url to link to

Example:

```
<customproperties>
  <customproperty name="exitlink" text="Back to start" url="http://somewhere/else" />
</customproperties>
```

7.6.3 script

Custom Property: script

FrontEnd Version: 1.6

Description:

Attributes: none.

Example:

```
<customproperties>
  <customproperty name="script">
<![CDATA[

// do some JavaScript. I'm sure Michael has a good example.

]]>
</customproperty>
```



```
</customproperties>
```

7.6.4 buildtable

Custom Property: *buildtable*

FrontEnd Version: 1.6

Description:

Attributes: none.

Sub-tags: *insertquestion*

Example:

```
<customproperties>
<customproperty name="buildtable">
<table>
<tr>
<td>
This is an html text describing testing the buildtable customproperty.

</td>
</tr>
</table>
</customproperty>
</customproperties>
```

or

```
<customproperties>
<customproperty name="buildtable">
<table border="1" width="100%">
<tr><td>question here: <insertquestion name="Question3"
/></td><td>testing456</td></tr>

<tr><td><insertquestion name="Linetest" questionpart="text"
/></td><td><insertquestion name="Linetest" /></td></tr>

<tr><td colspan="2" align="center"><insertquestion name="calctable" /></td></tr>

<tr><td><insertquestion name="Question2" answer="Option3" /></td><td><insertquestion
name="Question2" answer="Option1" /></td></tr>

<tr><td><insertquestion name="Question2" answer="opt2" />

<insertpicker name="dstItems" id="picker1">
  <map questionname="Question2" field="itemID" />
  <map questionname="Question3" field="entryDate" />
</insertpicker>

</td></tr>

<tr><td colspan="2"><insertquestion name="ss11" /></td></tr>
</table>
</customproperty>
</customproperties>
```

Special tags used in the content of "buildtable":

7.6.4.1 *insertquestion*

Tag: *insertquestion*

FrontEnd Version: 1.6

Description: Inserts a question field inside the table.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Name</i>	No	The question name.	
<i>questionpart</i>	Yes	The questionpart attribute specifies the part of the question you wish in insert. There are currently 4 possible parts: "field" (the question input field) "text" (the question text) "description" (the question's description) " " (the selector that accompanies the question, if any, type e.g. the date picker) If omitted questionpart will default to "field".	
<i>Answer</i>	Yes	If the answer attribute is included the field, test or description that is applicable to the answer is inserted and not the entire question. E.g. If you wish to place a radio button that represents one possible answer you would use the answer attribute.	

Example:

```
<insertquestion name="table_name" questionpart="text"/>
```

7.6.4.2 *insertpicker (obsolete)*

Tag: *insertpicker*

FrontEnd Version: 1.6

Description: inserts an icon in the layout that opens the CodePicker.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>name</i>	No	The URL of the CodePicker configuration file to use.	
<i>id</i>	Yes	A valid id	Used if you wish to use the picker in the show hide object.
<i>language</i>	Yes	The name of the CodePicker language configuration file to use.	
<i>icon</i>	Yes	The URL of a gif file to use as an icon.	

Sub-tags: *map*

Example:

```
<insertpicker name="VarecodePicker" id="picker1" language="TranslationsEnglish">
  <map questionname="test" field="Varecode" />
</insertpicker>
```

```
<map questionname="test2" field="Varetxt" />
</insertpicker>
```

7.6.4.3 *insertaddon (obsolete)*

Tag: *insertaddon*

FrontEnd Version: 1.6

Description: inserts an icon in the layout that opens a customised add-on popup window.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>url</i>	No	A URL	The web address of the add-on popup window.
<i>name</i>	No	A string.	The unique name of the addon block you are inserting. This is used for two purposes. Firstly it can be used with a show hide object but using the AddByQuestionName method. Secondly it is used to reference the parameter string so that you can add URL parameters at runtime. E.g. RM_Addon_Params['name'] = 'param1="fred"¶m2="michael"'. The string will be appended at the end of the URL.
<i>description</i>	Yes	A string.	A description of the add-on block that will pop up in a tooltip.
<i>icon</i>	Yes	The url of a gif file to use as an icon.	

Sub-tags: *map*

Example:

```
<insertpicker name="VarecodePicker" id="picker1" language="TranslationsEnglish">
  <map questionname="test" field="Varekode" />
  <map questionname="test2" field="Varetxt" />
</insertpicker>
```

7.6.4.4 *map (obsolete)*

Tag: *map*

FrontEnd Version: 1.6

Description: Adds a new field to fill by the CodePicker or Addon. The number of map tags should equal the number of return-value tags in the Addon or CodePicker's configuration file.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
questionname	No	The name of the question the CodePicker or Addon will read or fill.	
field	No	The name of the field in the Addon or CodePicker application file.	
display	Yes	(Add-on only) An optional string specifying any	

		display attributes the add-on will take e.g. "readonly"	
--	--	---	--

7.6.5 backbuttontext

Custom Property: *backbuttontext*

FrontEnd Version: 2.11.1

Description: Defines the custom text as a caption for the "Back" button. If existing and not empty, it has precedence over the same custom property on Questionnaire and Project level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	Any text	The text to use.

Example:

```
<customproperties>
  <customproperty name="backbuttontext" value="Previous page" />
</customproperties>
```

7.6.6 backbuttonhidden

Custom Property: *backbuttonhidden*

FrontEnd Version: 2.11.1

Description: Indicates whether to hide the "Back" button.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	true false [empty]	If "true" then the "Back" button is hidden; otherwise it has no impact on it.

Example:

```
<customproperties>
  <customproperty name="backbuttonhidden" value="true" />
</customproperties>
```

7.6.7 backbuttondisabled

Custom Property: *backbuttondisabled*

FrontEnd Version: 2.11.1

Description: Indicates whether to disable the "Back" button.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	true false [empty]	If "true" then the "Back" button is disabled; otherwise it has no impact on it.

Example:

```
<customproperties>
  <customproperty name="backbuttondisabled" value="true" />
</customproperties>
```

7.6.8 continuebuttontext

Custom Property: *continuebuttontext*

FrontEnd Version: 2.11.1

Description: Defines the custom text as a caption for the "Continue" button. If existing and not empty, it has precedence over the same custom property on Questionnaire and Project level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	Any text	The text to use.

Example:

```
<customproperties>
  <customproperty name="continuebuttontext" value="Next page" />
</customproperties>
```

7.6.9 continuebuttonhidden

Custom Property: *continuebuttonhidden*

FrontEnd Version: 2.11.1

Description: Indicates whether to hide the "Continue" button.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	true false [empty]	If "true" then the "Continue" button is hidden; otherwise it has no impact on it.

Example:

```
<customproperties>
  <customproperty name="continuebuttonhidden" value="true" />
</customproperties>
```

7.6.10 continuebuttondisabled

Custom Property: *continuebuttondisabled*

FrontEnd Version: 2.11.1

Description: Indicates whether to disable the "Continue" button.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	Yes	true false [empty]	If "true" then the "Continue" button is disabled; otherwise it has no impact on it.

Example:

```
<customproperties>
  <customproperty name="continuebuttondisabled" value="true" />
</customproperties>
```

7.7 Question Level

7.7.1 placeholderText

Custom Property: *placeholderText*

FrontEnd Version: 6.0

Description: The text that is placed in a textbox before a textbox contains any data or has focus. Used as descriptive or instructional text.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>value</i>	no	The placeholder text.	

Example:

```
<customproperties>
  <customproperty name="placeholderText" value="Please enter your surname." />
</customproperties>
```

7.7.2 oioexportname

Custom Property: *oioexportname*

FrontEnd Version: 2.4

Description: The name of the aggregate for the complete signing data. Only needed when multiple signing is used.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>value</i>	no	The name of the aggregate in the export server.	

Example:

```
<customproperties>
  <customproperty name="oioexportname" value="oc_q_lagg" />
</customproperties>
```

7.7.3 transactionidquestionname

Custom Property: *transactionidquestionname*

FrontEnd Version: 2.4

Description: The name of the aggregate for the complete signing data. Only needed when multiple signing is used.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>value</i>	no	The name of the aggregate in the export server.	

Example:

```
<customproperties>
  <customproperty name="oioexportname" value="oc q lagg" />
</customproperties>
```

7.7.4 signexportname

Custom Property: *signexportname*

FrontEnd Version: 2.1

Description: The name of the aggregate to get the signing data from the export server.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>value</i>	no	The name of the aggregate in the export server.	

Example:

```
<customproperties>
  <customproperty name="signexportname" value="oc_q_lagg" />
</customproperties>
```

7.7.5 signenableskip

Custom Property: *signenableskip*

FrontEnd Version: 2.1

Description: The name of the aggregate to get the signing data from the export server.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>value</i>	no	Can be either 'true' or 'false'.	Indicates whether or not to allow the user to skip the signing.

Example:

7.7.6 signcollectionidquestionname

Custom Property: *signcollectionidquestionname*

FrontEnd Version: 2.1

Description: The question name that contains the collection id for the file attachments.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>value</i>	no	A question name.	The name of the question that

Example:

7.7.7 appletypequestionname

Custom Property: *appletypequestionname*

FrontEnd Version: 2.3

Description: Specifies the name of the question that contains what type of applet to use.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A string.	The type of applet to use

Example:

```
<customproperties>
  <customproperty name=" appletypequestionname " value="lite" />
```

```
</customproperties>
```

7.7.8 forceappletypequestionname

Custom Property: *forceappletypequestionname*

FrontEnd Version: 2.3

Description: Specifies the question that determines whether to bypasses the applet detection stage on the client and force it to use the applet specified in the applet type.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	'true' or 'false'	Sets whether or not to bypass the applet detection process.

Example:

```
<customproperties>
  <customproperty name="forceappletypequestionname" value="true" />
</customproperties>
```

7.7.9 exportname

Custom Property: *exportname*

FrontEnd Version: 1.7.1

Description: The exportname custom property is only applicable inside the PDF Button question type. It specifies the export name.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A string.	The export name of the PDF

Example:

```
<customproperties>
  <customproperty name="exportname" value="DP 200A PDF" />
</customproperties>
```

7.7.10 attributes

Custom Property: *attributes*

FrontEnd Version: 1.6

Description:

Attributes: special

Example:

```
<customproperty name="attributes" onchange="getoptiontest()" />
```

7.7.11 displaytextattributes

Custom Property: *displaytextattributes*

FrontEnd Version: 1.6**Description:****Attributes:** special**Example:**

```
<customproperty name="displaytextattributes" onchange="getoptiontest()" />
```

7.7.12 default**Custom Property:** *default***FrontEnd Version: 1.6****Description:****Attributes:**

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	"true" or "false"	
<i>Text</i>	Yes	Default text	

Example:

```
<customproperties>
  <customproperty name="default" value="true" text="Choose Here" />
</customproperties>
```

7.7.13 required**Custom Property:** *required***FrontEnd Version: 1.6****Description:****Attributes:**

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	"true" or "false"	

Example:

```
<customproperties>
  <customproperty name="required" value="true" />
</customproperties>
```

7.7.14 maxlength

Custom Property: *maxlength*

FrontEnd Version: 1.6

Description: The maxlength custom property sets the maximum length of the string the user can type inside the field.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A number (integer)	The maximum number of characters the field will allow.

Example:

```
<customproperties>
  <customproperty name="maxlength" value="10" />
</customproperties>
```

7.7.15 hidden

Custom Property: *hidden*

FrontEnd Version: 1.6

Description: used to place variables and text on the page with out it being displayed. Use this instead of hidden question type. Do not confuse this type of hidden question with RM_ShowHide Scripting Class.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	"true" or "false"	Sets whether the question should be hidden.

Example:

```
<customproperties>
  <customproperty name="hidden" value="true" />
</customproperties>
```

7.7.16 displayonly

Custom Property: *displayonly*

FrontEnd Version: 1.6

Description: Displays the question's variable or default value on the page and don't allow any input.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	"true" or "false"	Sets whether the question should be displayed only.

Example:

```
<customproperties>
  <customproperty name="displayonly" value="true" />
</customproperties>
```

7.7.17 taborder

Custom Property: *taborder*

FrontEnd Version: 1.7.4

Description: Sets the tab order for the question

Attributes:

Name	Optional	Possible Values	Description
<i>Value</i>	No	A number.	A number signifying the tab order of the question

Example:

```
<customproperties>
  <customproperty name="taborder" value="3" />
</customproperties>
```

7.7.18 focus

Custom Property: *focus*

FrontEnd Version: 1.7.4

Description: Sets the focus on this question when the page is loaded.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	"true" or "false"	If true the focus will be on the question when the page is loaded.

Example:

```
<customproperties>
  <customproperty name="focus" value="true" />
</customproperties>
```

7.7.19 applettype

Custom Property: *applettype*

FrontEnd Version: 2.3

Description: This custom property is only applicable to the applet question type. This specifies which applet to use

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A String	The applet to use.

Example:

```
<customproperties>
  <customproperty name="applettype" value="lite" />
</customproperties>
```

7.7.20 url

Custom Property: *url*

FrontEnd Version: 2.0

Description: On some question types such as “display link” and .image it is necessary to specify a URL. This is achieved through the url custom property. If this is not found the FrontEnd will look for the URL in an attached backend variable, failing this it will look in the default value. If a URL is still not found the question type will fail.

Attributes: none

Example:

```
<customproperties>
  <customproperty name="url">http://www.yahoo.com/</customproperty>
</customproperties>
```

7.7.21 openmode

Custom Property: *openmode*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the link question type.

This sets the mode of the link. E.g. whether it should open in a new window for not.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	Either “newwindow” or “self”	The newwindow mode opens a new window when the link is pressed. Self mode opens the link in the current window.

Example:

```
<customproperties>
  <customproperty name="openmode" value="self" />
</customproperties>
```

7.7.22 linktext

Custom Property: *linktext*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the link question type.

This sets the text on the link to something other than the default text.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A string	The text to use on the link

Example:

```
<customproperties>
  <customproperty name="linktext" value="Link here!" />
</customproperties>
```

7.7.23 applicationname

Custom Property: *applicationname*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the codepicker question type.

This sets the name of the codepicker application file to use.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A string	The name of the codepicker application file to use.

Example:

```
<customproperties>
  <customproperty name="applicationname" value="dstEPS live" />
</customproperties>
```

7.7.24 mode

Custom Property: *mode*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the codepicker question type.

This set the codepicker mode to use. A list of modes is available below.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	"inline", "popup" and "window"	The inline mode allows the codepicker to sit in the page as a static element like any other field on the page. The popup mode makes the codepicker appear over the page and then disappears when the user has finished using it – in a similar fashion to the date picker. The window mode opens the codepicker in a new window.

Example:

```
<customproperties>
  <customproperty name="mode" value="popup" />
</customproperties>
```

7.7.25 width

Custom Property: *width*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the codepicker question type.

This sets the width of the codepicker when it is in popup or window modes.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A width in pixels	The width of the codepicker in pixels.

Example:

```
<customproperties>
  <customproperty name="width" value="600" />
</customproperties>
```

7.7.26 height

Custom Property: *height*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the codepicker question type.

This sets the height of the codepicker when it is in popup or window modes.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A height in pixels	The height of the codepicker in pixels.

Example:

```
<customproperties>
  <customproperty name=" height" value="400" />
</customproperties>
```

7.7.27 icon

Custom Property: *icon*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the codepicker question type.

If this custom property is set is will display the specified icon instead of the default one.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A url	The location of the icon

Example:

```
<customproperties>
  <customproperty name=" icon" value="/static/icons/test.gif" />
</customproperties>
```

7.7.28 positionelement

Custom Property: *positionelement*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the codepicker question type.

If the this custom property is set the code picker when in popup mode will position it's self underneath the specified element. (like a date picker)

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A question name	The element that the codepicker should position itself under.

Example:

```
<customproperties>
  <customproperty name=" positionelement" value="myquestion" />
</customproperties>
```

7.7.29 mapping

Custom Property: *mapping*

FrontEnd Version: 2.4

Description: This custom property is only applicable to the codepicker question type.

This custom property contains a number of map elements that are used the map questions with the customproperty fields.

Attributes: none

Sub-tags: *map*

Example:

```
<customproperties>
  <customproperty name="mapping">
    <map questionname="Date" field="date" />
    <map questionname="ID" field="id" />
    <map questionname="Unit" field="PSIName" />
    <map questionname="UnitID" field="PSIPath" />
    <map questionname="Period" field="referencePeriod" />
    <map questionname="PeriodID" field="referencePeriodCode" />
    <map questionname="PathID" field="directionCode" />
    <map questionname="PathLabel" field="directionLabel" />
    <map questionname="OrganisationID" field="organisationID" />
    <map questionname="SelectedOrganisationID" field="selectedOrganisationID" />
    <map questionname="AuthorisationID" field="authorisationID" />
    <map questionname="Source" field="source" />
    <map questionname="Survey" field="survey" />
    <map questionname="AuthorisationsIndicator" field="authorisations" />
  </customproperty>
</customproperties>
```

1.1.1.1.1 map

Custom Property: *map*

FrontEnd Version: 2.4

Description: Adds a field map between the page and the codepicker. The number of map tags should equal the number of return-value tags in the CodePicker's application file.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
questionname	No	The name of the question the CodePicker or Addon will read or fill.	
field	No	The name of the field in the Addon or CodePicker application file.	

7.8 Answer Level

7.8.1 attributes

Custom Property: *attributes*

FrontEnd Version: 1.6

Description:

Attributes: special

Example:

```
<customproperty name="attributes" onchange="getoptiontest()" />
```

7.8.2 displaytextattributes

Custom Property: *displaytextattributes*

FrontEnd Version: 1.6

Description:

Attributes: special

Example:

```
<customproperty name="displaytextattributes" onchange="getoptiontest()" />
```

7.8.3 taborder

Custom Property: *taborder*

FrontEnd Version: 1.7.4

Description: Sets the tab order for the answer

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	A number.	A number signifying the tab order of the answer.

Example:

```
<customproperties>
  <customproperty name="taborder" value="3" />
</customproperties>
```


7.9 Flash custom properties

These properties are deprecated.

7.9.1 hideMenuOnFirstPage

Custom Property: *hideMenuOnFirstPage*

FrontEnd Version: 2.0

Description: Indicates to the Flash FrontEnd to hide the whole menu on the first presentation. This CustomProperty is valid on the project level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	true false	Indicates to the Flash FrontEnd to hide the menu on the first presentation.

Example:

```
<customproperties>
  <customproperty name="hideMenuOnFirstPage" value="true" />
</customproperties>
```

7.9.2 flashTheme

Custom Property: *flashTheme*

FrontEnd Version: 2.0

Description: Indicates to the Flash FrontEnd which theme file to use. This CustomProperty is valid on the project level and on the questionnaire level. The project level will override the default theme and the questionnaire level will override the default and the project level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	ToldSkat EogS	Indicates to the Flash FrontEnd to hide the menu on the first presentation.

Example:

```
<customproperties>
  <customproperty name="flashTheme" value="ToldSkat" />
</customproperties>
```

7.9.3 foldInOutSpeedExternal

Custom Property: *foldInOutSpeedExternal*

FrontEnd Version: 2.0

Description: Sets the speed in which the Flash FrontEnd should fold in and out sub levels. This CustomProperty is valid on the project level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	Number less than 200	Sets the fold in and out speed to be used on the Flash FrontEnd.

Example:

```
<customproperties>
  <customproperty name="foldInOutSpeedExternal" value="200" />
</customproperties>
```

7.9.4 menuScrollSpeedExternal

Custom Property: *menuScrollSpeedExternal*

FrontEnd Version: 2.0

Description: Sets the menu scroll speed to the Flash FrontEnd. This CustomProperty is valid on the project level.

Attributes:

NAME	OPTIONAL	POSSIBLE VALUES	DESCRIPTION
<i>Value</i>	No	Number less than 200	Sets the menu scroll speed to be used on the Flash FrontEnd.

Example:

```
<customproperties>
  <customproperty name="menuScrollSpeedExternal" value="200" />
</customproperties>
```