# Office AMS: Modify host web lists at list creation time

|  |  |
| --- | --- |
| Summary: | Applies to: |
| This scenario shows how you can modify a list created in the host web at list creation time | * Office 365 Multi-Tenant (MT) * Office 365 Dedicated (D) * SharePoint 2013 on-premises |
| Solution: | Core.EventReceiversBasedModifications, version 1.0 |
| Author: | Bert Jansen (Microsoft) |
| //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  // THIS CODE IS PROVIDED \*AS IS\* WITHOUT WARRANTY OF  // ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING ANY  // IMPLIED WARRANTIES OF FITNESS FOR A PARTICULAR  // PURPOSE, MERCHANTABILITY, OR NON-INFRINGEMENT.  //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* | |

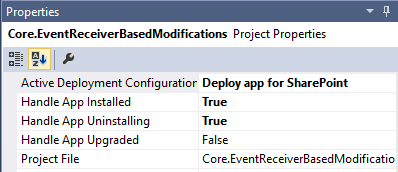
# General comments

This example shows how you can modify lists when they’re created in the host web. Whenever a user creates a new list in the host web a ListAdded remote event receiver gets fired and in this remote event receiver you can modify the list to your needs. Typical changes would be enabling versioning or adding a content type to the list, but in reality anything that can be done via CSOM can be used.

As we’re hooking up the list to the host web we’ll need to programmatically hookup the event receiver. In the sample we’ve opted to do this as part of an app: when the app is installed an AppInstalled event will fire and we’ll use this event to hookup the ListAdded event.

## Ensuring APpInstalled and APPUnInstalling events fire

The app events are set as properties of the SharePoint project:



## Hooking up the ListAdded event

The remote event receiver that gets executed on app install will dynamically add the ListAdded event receiver. Below code snippet shows how this is done:

bool rerExists = false;

cc.Load(cc.Web.EventReceivers);

cc.ExecuteQuery();

foreach (var rer in cc.Web.EventReceivers)

{

if (rer.ReceiverName == RECEIVER\_NAME)

{

rerExists = true;

}

}

if (!rerExists)

{

EventReceiverDefinitionCreationInformation receiver = new EventReceiverDefinitionCreationInformation();

receiver.EventType = EventReceiverType.ListAdded;

//Get WCF URL where this message was handled

OperationContext op = OperationContext.Current;

Message msg = op.RequestContext.RequestMessage;

receiver.ReceiverUrl = msg.Headers.To.ToString();

receiver.ReceiverName = RECEIVER\_NAME;

receiver.Synchronization = EventReceiverSynchronization.Synchronous;

cc.Web.EventReceivers.Add(receiver);

cc.ExecuteQuery();

}

## CUSTOMIZING the added lists

When the ListAdded event handler is firing then the following code is executed. This is just a simple sample that leverages Office AMS Core methods to set versioning, but in reality you can do any kind of manipulation that you might need.

private void HandleListAdded(SPRemoteEventProperties properties)

{

using (ClientContext cc = TokenHelper.CreateRemoteEventReceiverClientContext(properties))

{

if (cc != null)

{

try

{

if (properties.ListEventProperties.TemplateId == (int)ListTemplateType.DocumentLibrary)

{

//set versioning

cc.Web.GetListByTitle(properties.ListEventProperties.ListTitle).UpdateListVersioning(true, true);

}

}

catch (Exception ex)

{

System.Diagnostics.Trace.WriteLine(ex.Message);

}

}

}

}

}

## Dealing with uninstall

When the app is uninstalled we’re also removing the event receiver. In order to make this work during debugging you’ll need to ensure that you navigate to the “Apps in testing” library and use the remove option on the app. This remove will trigger the app uninstalling event with the proper permissions to remove the created remote event handler. If you just close the browser or uninstall the app from the “site contents” then either the event receiver never fires or the event receivers runs with unsufficient permissions to remove the list added event receiver. The reason for this behavior is differences in app deployment when the app gets side loaded which is what Visual Studio does when you press F5.

Note:

If you’ve been experimenting a lot it often helps to test this sample in a clean developer site.