
Change Tracking MS Dynamics CRM

Contents

Introduction.....	1
Approach	3
Using Plugin	3
Further Improvements	4

Introduction

Dynamics CRM 2016

Applies To: Dynamics 365 (online), Dynamics 365 (on-premises)

In Microsoft Dynamics 365, An out of the box feature to capture the changes on records by the Users or System is achieved by enabling the "Audit" for an entity but there is no out of the box feature to track the changes done by System Administrators/System Customizers or anyone having access to make a change in the application.

There could be requirements where the business owners and IT Admins want to see the changes done in the application for a specific time period. It could be useful during a Live Site, Troubleshooting or fixing an issue. But, there is no audit available for such customization changes in MS CRM right now.

CRM stores the information about various components like Entities, Web resources (HTML, JavaScript etc), Workflows in its data base.

The extraction of this information and the triggering point to collect this information varies due to some restrictions for developers.

The approach followed for this solution is elaborated in this document.

What Change Tracking Achieves:

Component	Default Collection of Change tracking by D365 (Where and How)	Change Tracking Solution (How does it make it simpler)	Approach of Change Tracking solution.
Entity	No, CRM does not store the information about this when the component is modified or by whom it is modified	Change tracking, tracks whenever this component is modified & published and stores this information in a new entity	Using "Publish" message, we extract information of the modified user and time when an entity is modified and published.
Web Resources	CRM stores the information about the web resources in "Web resource" table but this is not visible in the application.	Change tracking captures the web resource details which includes modified user, modified time and resource type. whenever a web resource is modified & published, Change tracking stores this information in an entity called "Change Tracking"	Using "Publish" message, we extract the information of the modified user and time when a web resource is modified and published.

Plugin Assemblies	CRM stores the information about the assemblies in "pluginassembly" table but this is not visible in the application	Change tracking uses a batch job which executes in a specified time frame and captures the modified time and user and stores this information in a new entity.	Using a batch job which gets the data from "Pluginassembly" entity.
Workflows	CRM stores the information about the workflows in "Workflow" table but this is not visible to the user in the application.	Change tracking uses a batch job which executes in a specified time frame and captures the modified time and user and stores this information in a new entity.	Using a batch which gets the data from "workflow" entity.

Approach

Using Plugin

For Entities & Web resources

- "Publish" Message:** A plugin can be registered for this message type which will be executed whenever someone publishes on entity or web resource like JavaScript/HTML/XML etc. This plugin also captures name of the component being published in XML format.

Limitation: This message type can capture the information of Entities & Web Resources. This message cannot capture the information about the workflows and plugin assemblies (dll files) as the method of updating these are different and are not "published"
- "PublishAll" Message:** A plugin can be registered for this message type which will be executed whenever someone publishes a group of components in a solution using "Publish all customizations" button.

Limitation: We will not have the information about the names of the components which are being published like we get in "publish" plugin. This only can capture limited information like which user published and when it was done.

For Plugin Assemblies (dll files) & Workflows:

The information about Plugin assemblies is stored in "pluginassembly" entity and Workflows is stored in "workflow" entity.

We cannot register any plugin on these entities to track the information about the modifications done as these are not available to developers in plugin registration tool. There is no trigger point for the code to execute when developer updates a dll or workflow.

We use a batch job which will run every hour and fetches data like "ModifiedOn", "ModifiedBy", "Name of the component" about the Plugin assemblies and workflows,

Limitation: This will only give information about the last modified before this batch job executes.

Further Improvements

- We can develop reports based on the data we capture and get the information about the modified components in the application between a given time period
- We can develop charts and Dash boards using the data according to the business requirements.