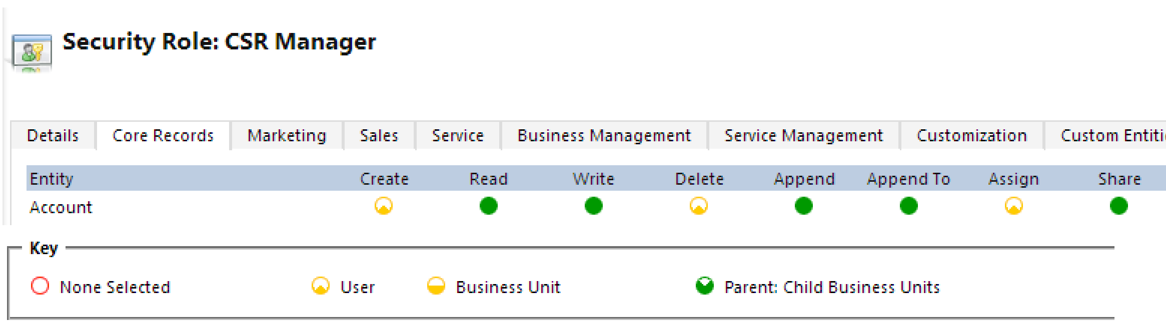
Improving Performance with Access Teams in Microsoft Dynamics CRM 2013

Introduction

Access Teams is a new addition in Microsoft Dynamics CRM 2013 that brings along a new way to define privileges and permissions for entities in CRM.

Prior to Microsoft Dynamics CRM 2013 (in CRM 2011), security roles were the only means by which privileges were defined for entities. These privileges— Create, Read, Write, Delete, Append, Append To, Assign, Share—were defined in configurations for Security Roles from a User level of access to an Organization level of access.



Once security roles had been configured, they were then assigned to various users or teams where the privileges defined in the security roles would get additively applied.

However, there are a couple of caveats with the previous implementation of security:

1. Security roles defined privileges for an entire entity.
2. Due to the cumulative effect of security roles, if a user has several roles and/or belongs to a team with a role, performance can suffer as CRM needs to process and calculate the final set of privileges. Furthermore, any time roles get added/removed from users frequently, the cache in CRM is flushed for the user and performance is slowed further.

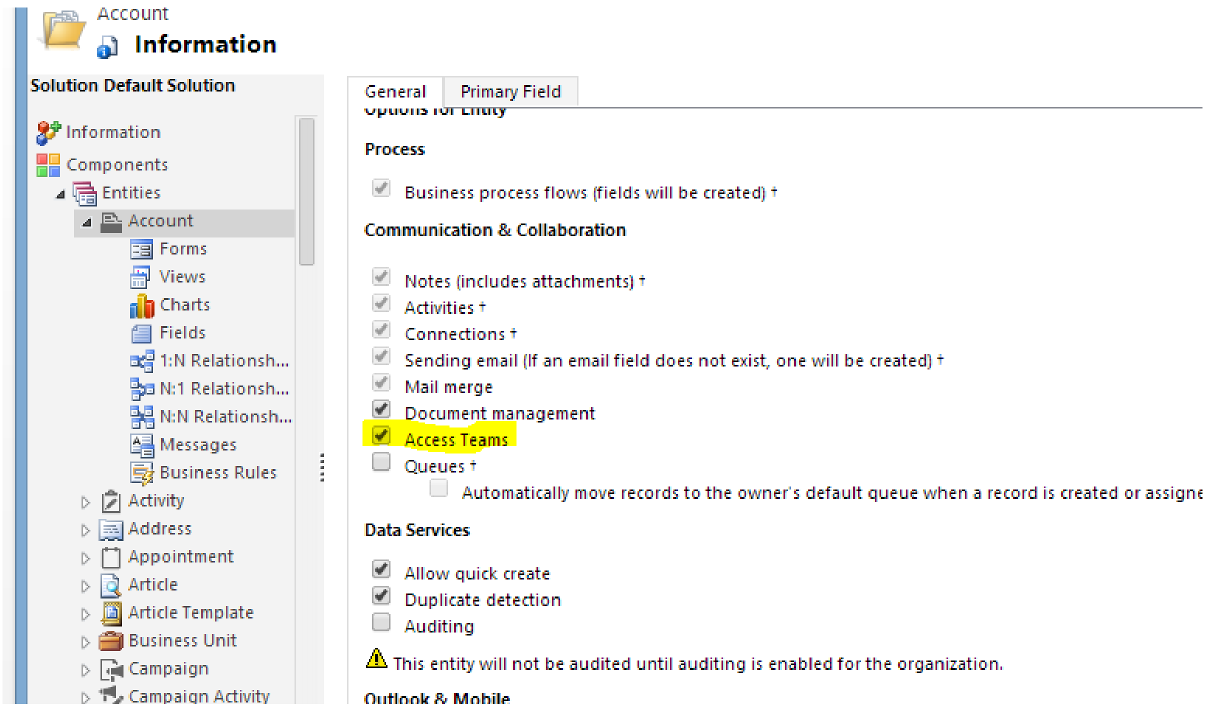
So, in CRM 2013 we have the introduction of **Access Teams** where privileges can get set for an individual record level, and since Access Teams don't have security roles associated with them, the performance implications involved with multiple security roles and changing of the security roles have been eliminated.

Setting up Access Teams

Access Teams are easily configurable out of the box. Here are the steps to set up an Access Team for a record:

Step 1

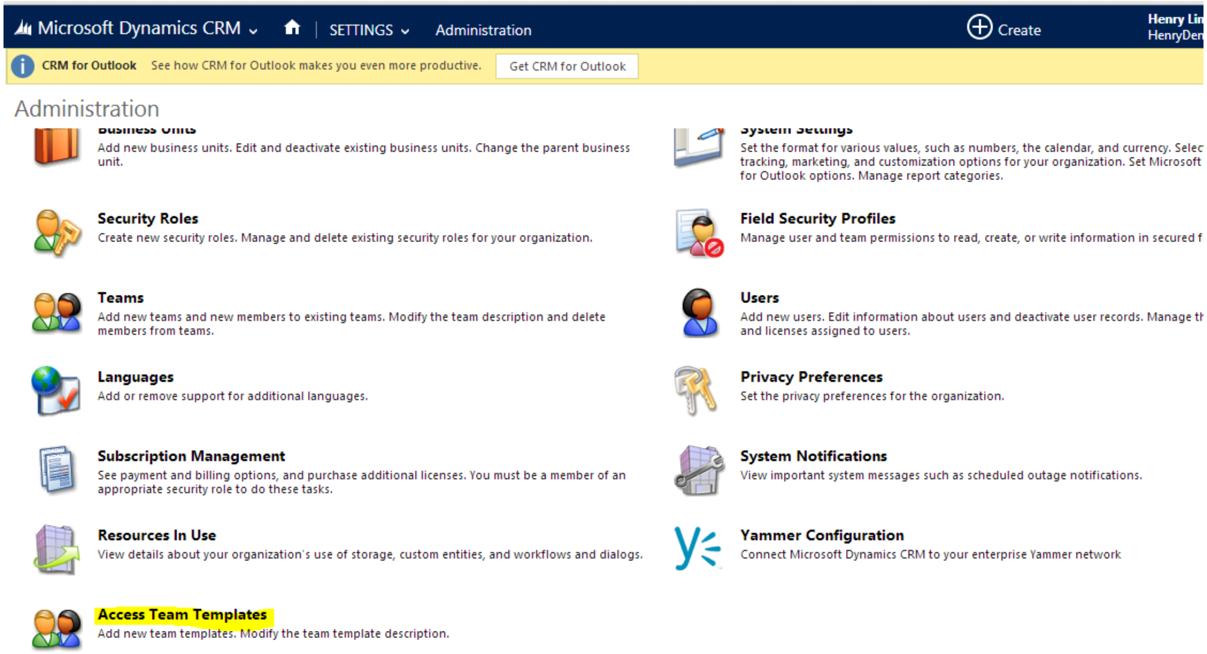
Enable the entity for Access Teams:



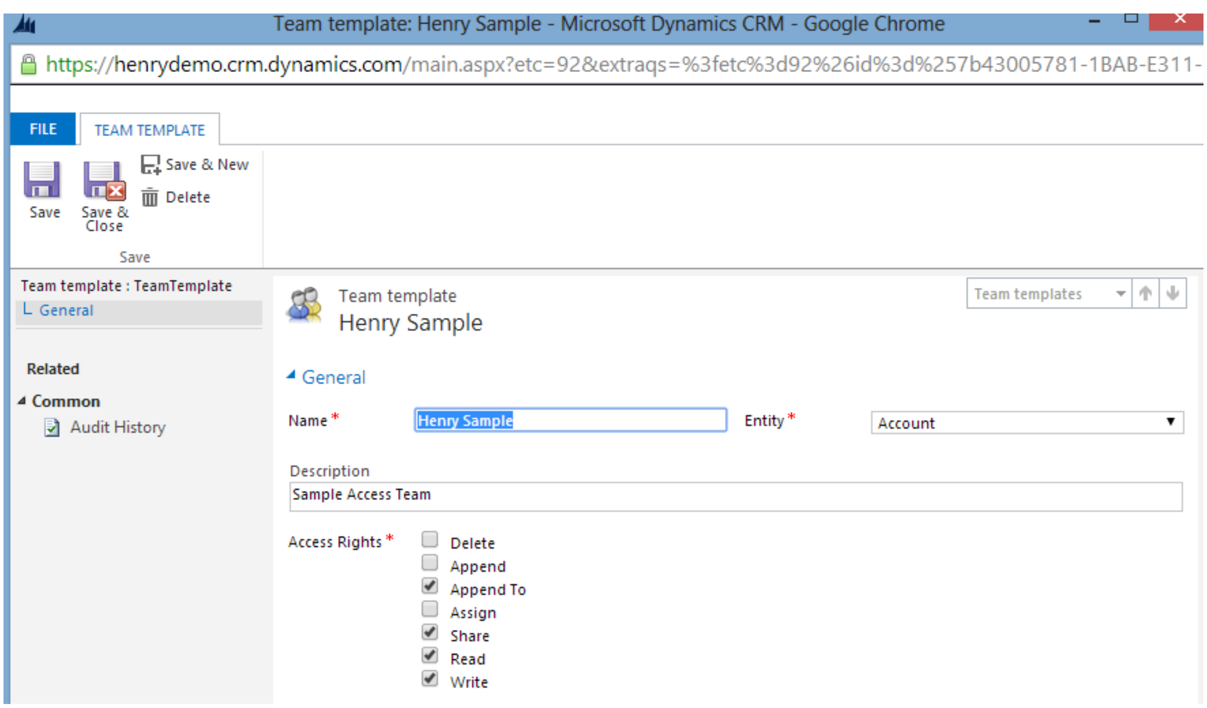
In Settings -> Customizations, select the checkbox for Access Teams (highlighted in screenshot above)

Step 2.

Set up the Access Team Template for the entity's Access Team:

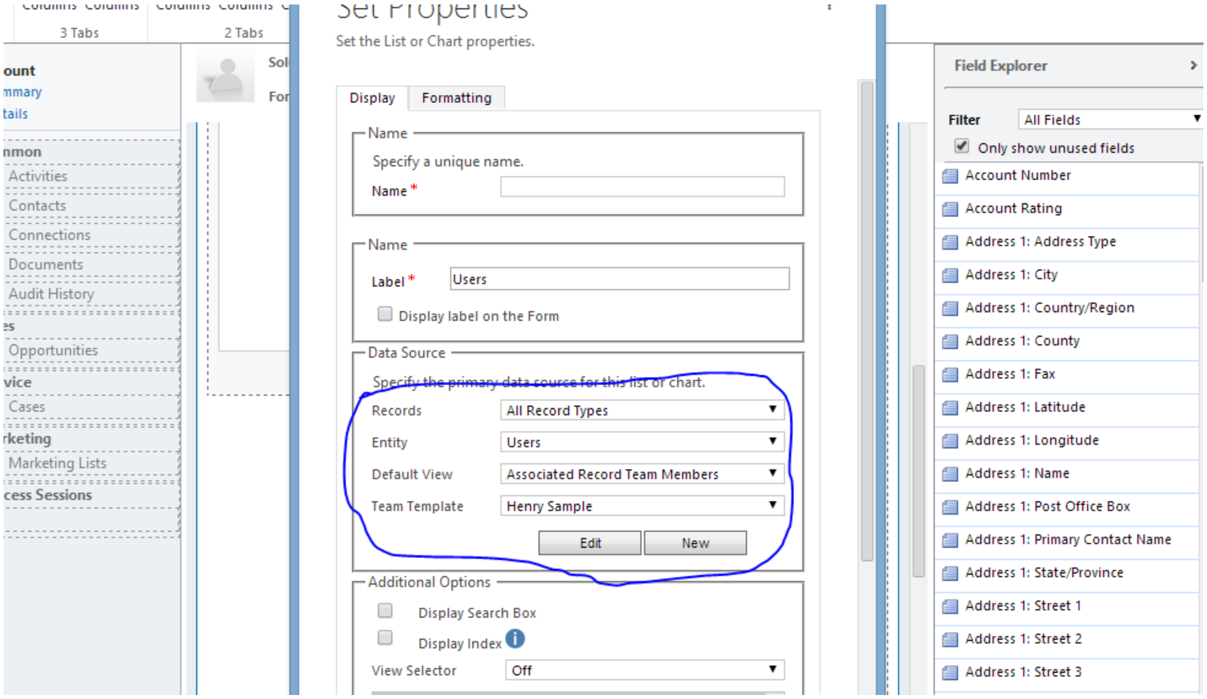


In Settings -> Administration, click open **Access Team Templates** and create a new Access Team. In addition, select the privileges in Access Rights that you would like to assign for the **Access Team:**



Step 3.

Add a Sub-Grid for Users on the form for the entity that you wish to enable Access Teams for:

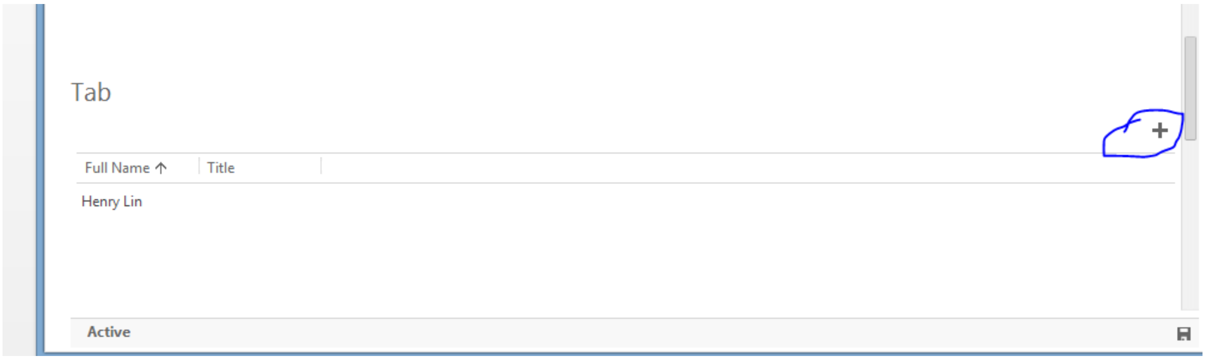


Make sure to select:

All Record Types for Records, **Users** for entity, **Associated Record Team Members** for Default View; this selection will pop up the team template that you wish to use and select your Team Template.

Step 4.

On any given record, add Users to the sub-grid to grant them privileges based on the Team Template specified in the Sub-Grid:



Step 5.

Alternatively, a new message has been added to the SDK—**AddUserToRecordTeamRequest**—which can programmatically add a user to an access team—you just need to specify the GUID's of the Record, the User you want to add, and the Access Team Template you want to add to, also add a reference to **Microsoft.Crm.Sdk.Proxy**:

```
AddUserToRecordTeamRequest adduser = new AddUserToRecordTeamRequest()
{
Record = recordID,
SystemUserId = sysUserID,
TeamTemplateId = teamTemplateID
};
// Execute the request.
AddUserToRecordTeamResponse response =
(AddUserToRecordTeamResponse)service.Execute(adduser);
```