directions lyon
emea 2023

# Dynamics 365 Sales Dynamics 365 BC

## Integration

## Deep Dive

Tom Kapitan, Fusion5
Marko Totovic, Quby Technology

01. – 03. 11. 2023, Lyon, France

directions lyon
emea 2023

## Tom Kapitan

- Working with Dynamics NAV/Business Central since 2014
- Blogger – Kepty.cz
- BC Open-Source programs
- MSDyn365 Sales integration, Telemetry
- Snr. Tech Consultant, Fusion5 Business Solutions Australia

FUSION5
Business Solutions

#MakingPotentialReality

## Marko Totovic

- Working with Dynamics CE/Power Platform since 2016
- Blogger – totovic.com
- Microsoft MVP for Business Applications
- Quby Technology Co-Founder

QUBY
TECHNOLOGY

## Agenda

All about Dynamics 365 Sales and Business Central!

- Overview
- Get most from the OOTB functionality
  - Configuration
  - Entity synchronization
  - Sales Orders (Legacy/Bi-directional)
- Power Automate for integration
- Customizations
  - Basic customizations (custom fields, tables)
  - Update jobs, why we have them
  - Top 25 events you should know
- Current limitations
- Future plans
- Q&A

slido

## Who are you?

ⓘ Start presenting to display the poll results on this slide.

slido

## Join at slido.com
## #1236981

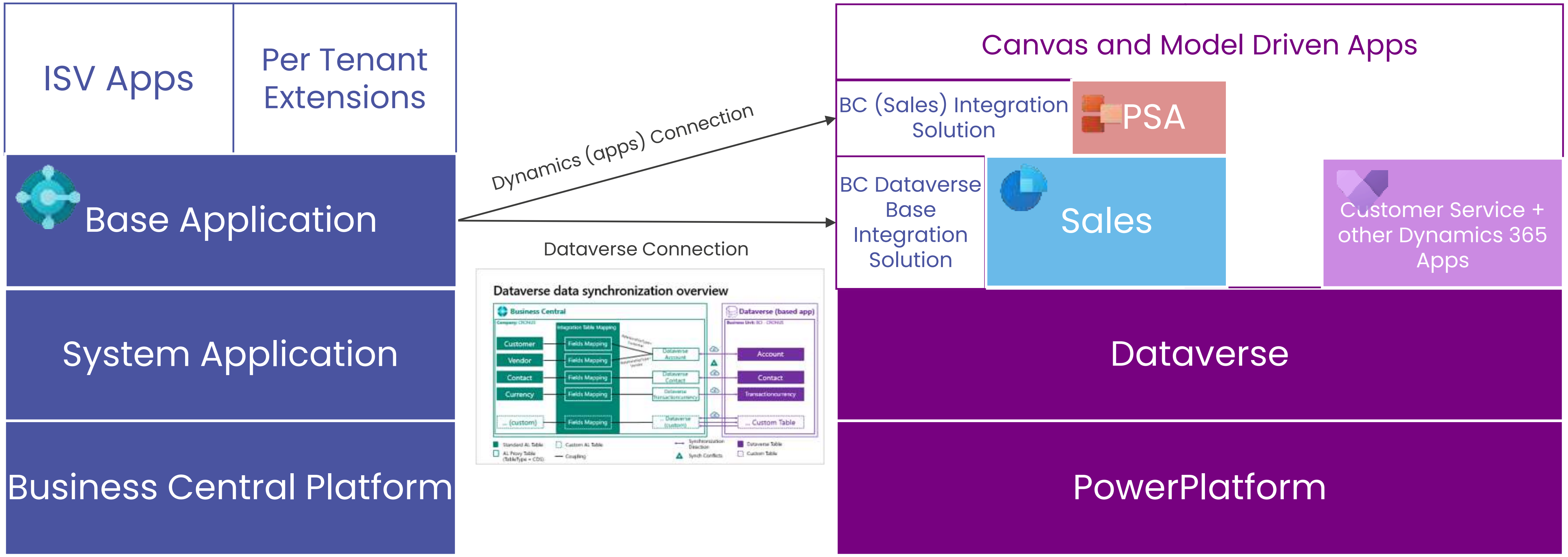
ⓘ Start presenting to display the joining instructions on this slide.

OOTB

# How data synchronization works

# How data synchronization works

To connect BC and Sales first you must connect it to Dataverse



OOTB

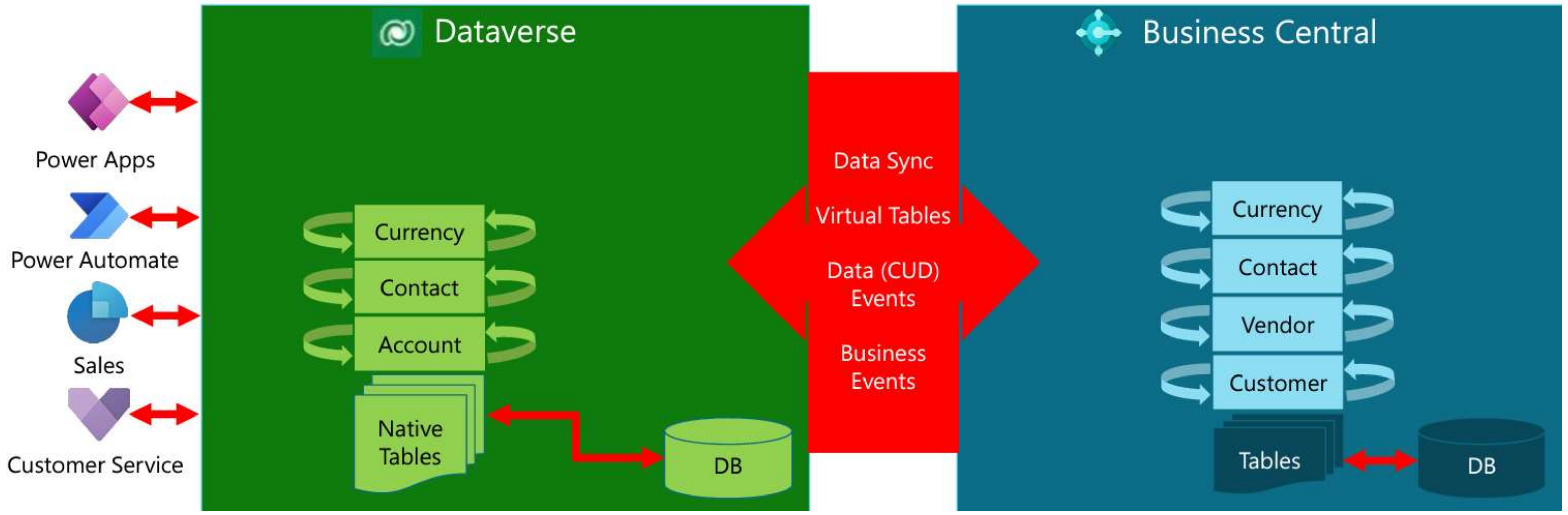
## Configuration

directions lyon emea 2023

## Connect BC to Dataverse

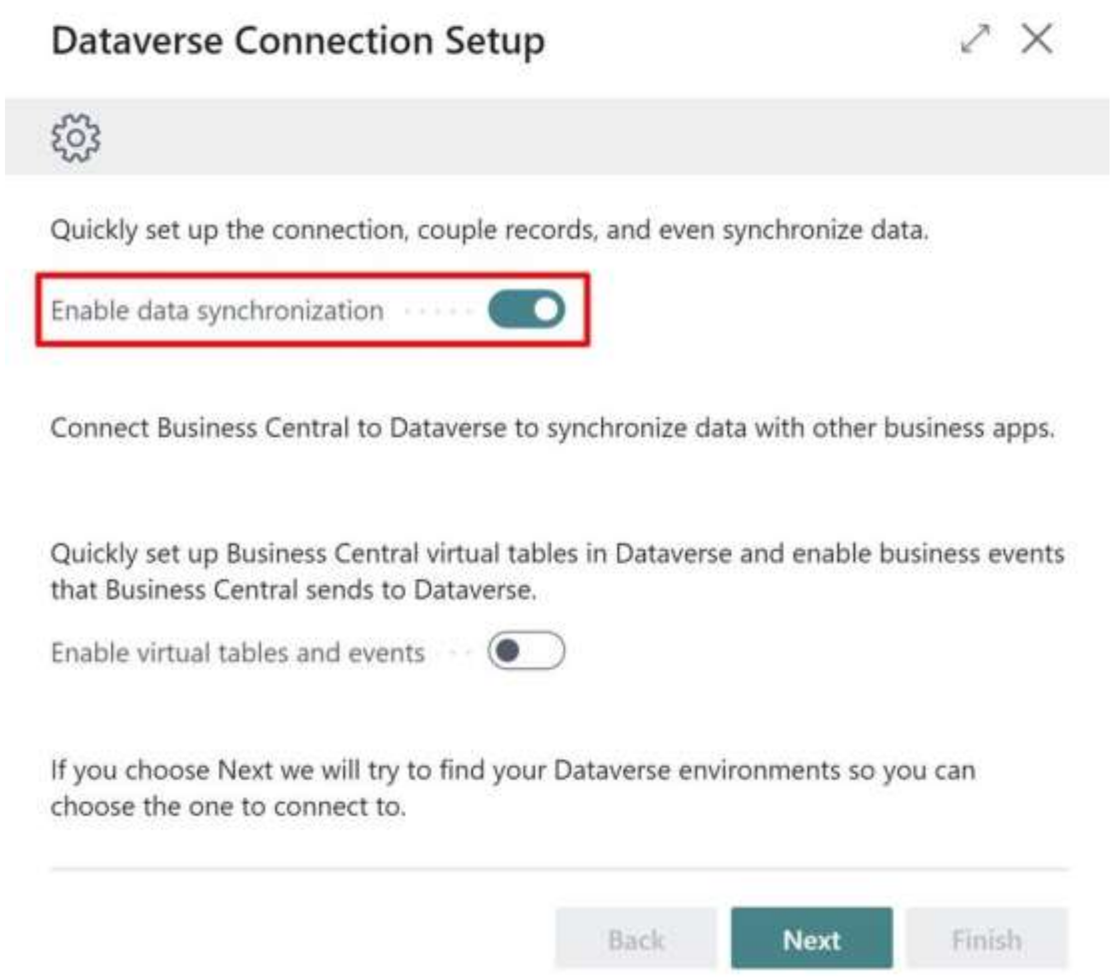
- Dataverse is a data storage management layer for Dynamics 365 Sales, Customer Service and Power Platform
  - It offers standard tables that are used in business scenarios, such as Account, Contact, and Currency
  - It implements business logic that enforces business rules validations process flows on data locally physically stored in those tables
  - Integrating with Dataverse enables Business Central to interact with other apps in its ecosystem on their overlapping non-overlapping data
  - There are four types of complementary interactions

## Connect BC to Dataverse



# Connect BC to Dynamics 365 Sales

directions lyon
emea 2023



Dataverse Connection Setup

Quickly set up the connection, couple records, and even synchronize data.

Enable data synchronization: On

Connect Business Central to Dataverse to synchronize data with other business apps.

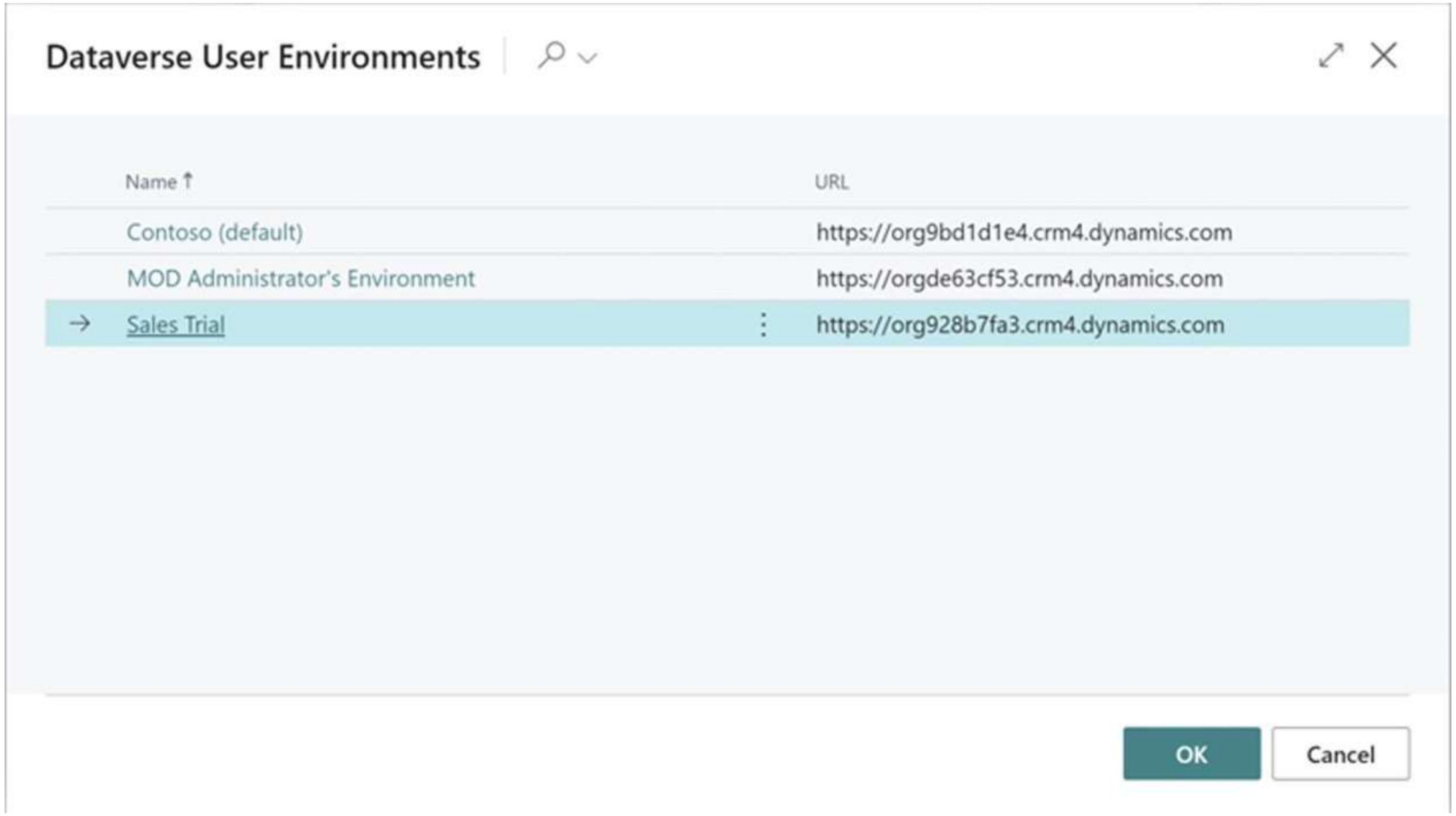
Quickly set up Business Central virtual tables in Dataverse and enable business events that Business Central sends to Dataverse.

Enable virtual tables and events: Off

If you choose Next we will try to find your Dataverse environments so you can choose the one to connect to.

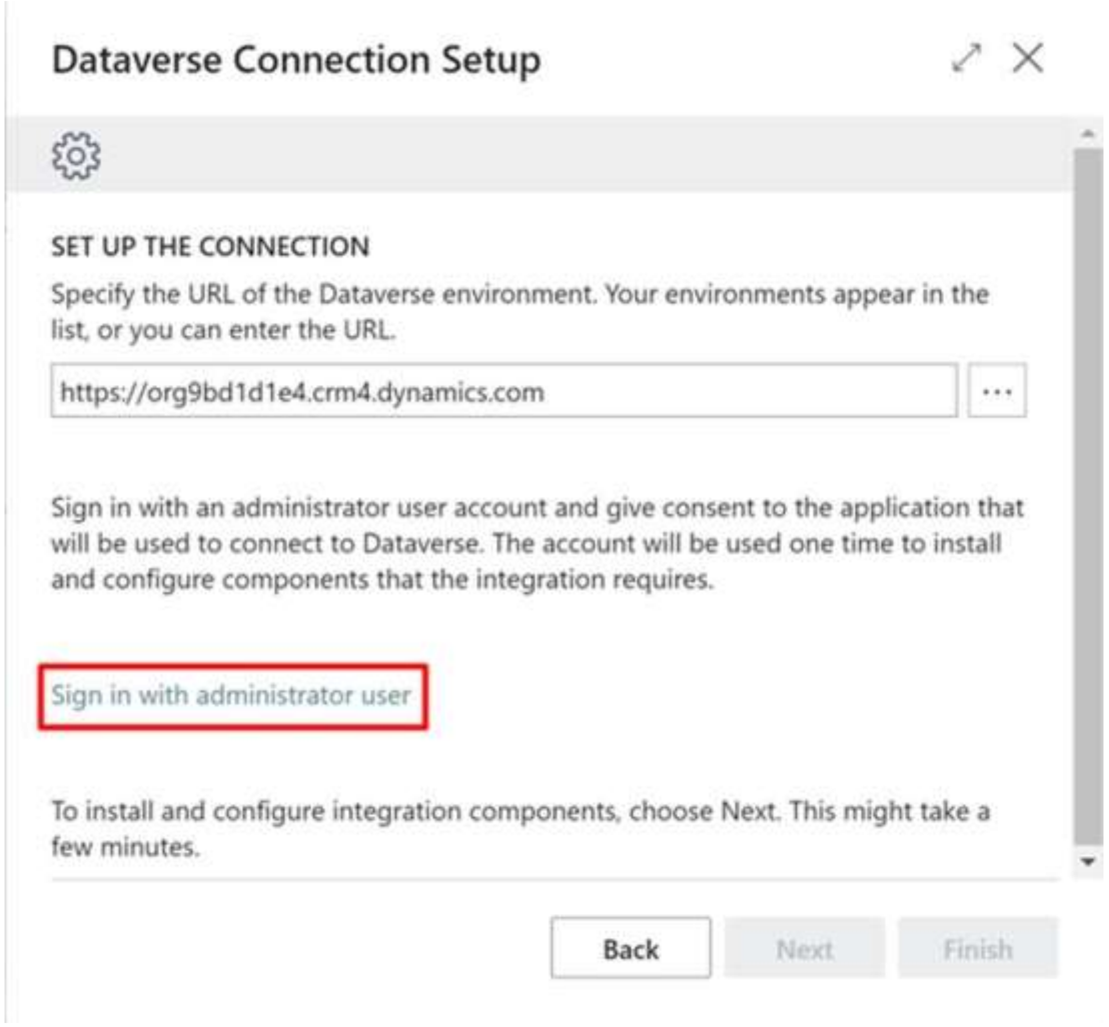
## Connect BC to Dynamics 365 Sales

directions lyon emea 2023


Dataverse User Environments

| Name | URL |
|---|---|
| Contoso (default) | https://org9bd1d1e4.crm4.dynamics.com |
| MOD Administrator's Environment | https://orgde63cf53.crm4.dynamics.com |
| Sales Trial | https://org928b7fa3.crm4.dynamics.com |

# Connect BC to Dynamics 365 Sales


Dataverse Connection Setup

SET UP THE CONNECTION

Specify the URL of the Dataverse environment. Your environments appear in the list, or you can enter the URL.

https://org9bd1d1e4.crm4.dynamics.com

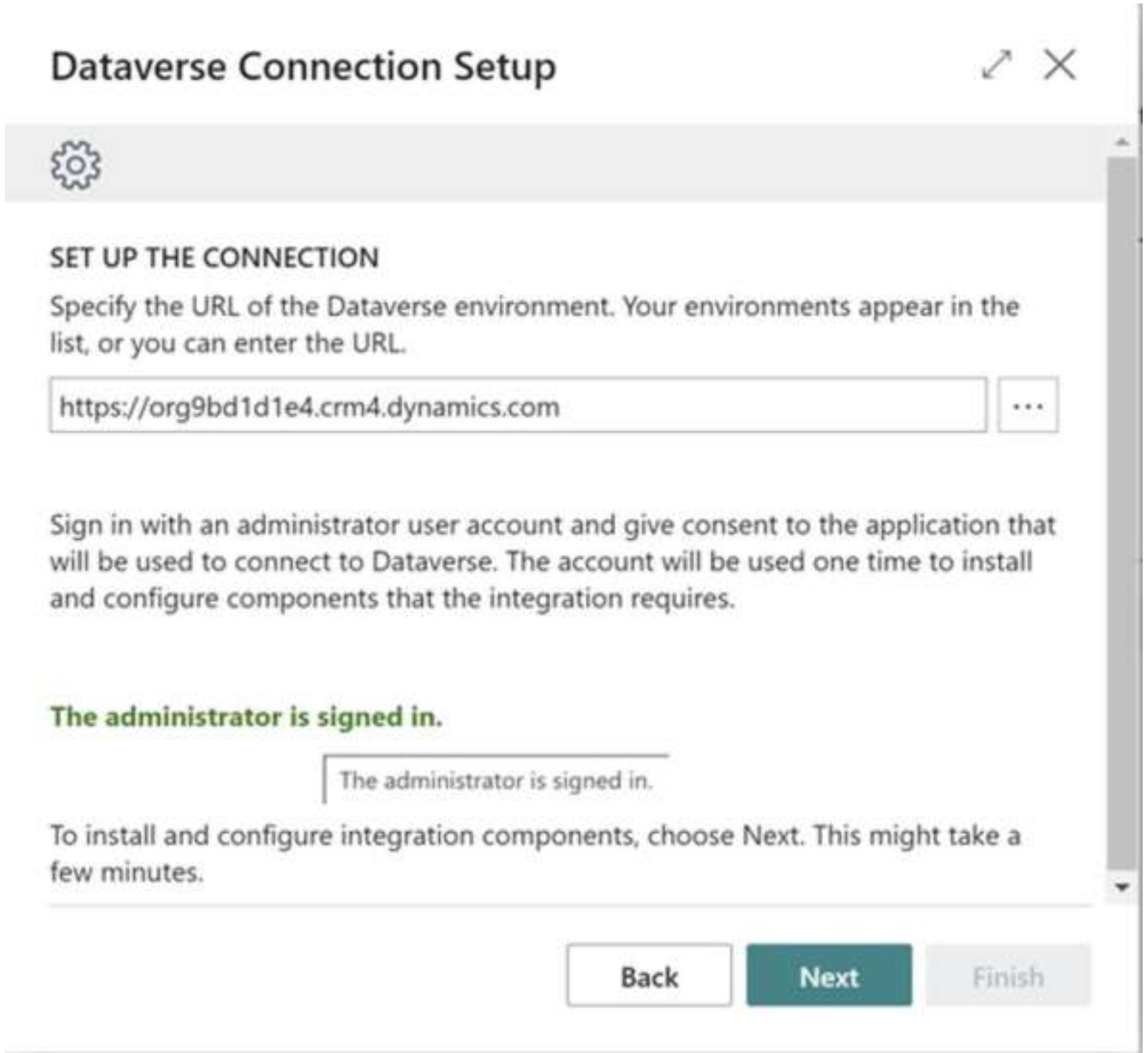
Sign in with an administrator user account and give consent to the application that will be used to connect to Dataverse. The account will be used one time to install and configure components that the integration requires.

Sign in with administrator user

To install and configure integration components, choose Next. This might take a few minutes.

# Connect BC to Dynamics 365 Sales

directions lyon emea 2023



Dataverse Connection Setup

SET UP THE CONNECTION

Specify the URL of the Dataverse environment. Your environments appear in the list, or you can enter the URL.

https://org9bd1d1e4.crm4.dynamics.com

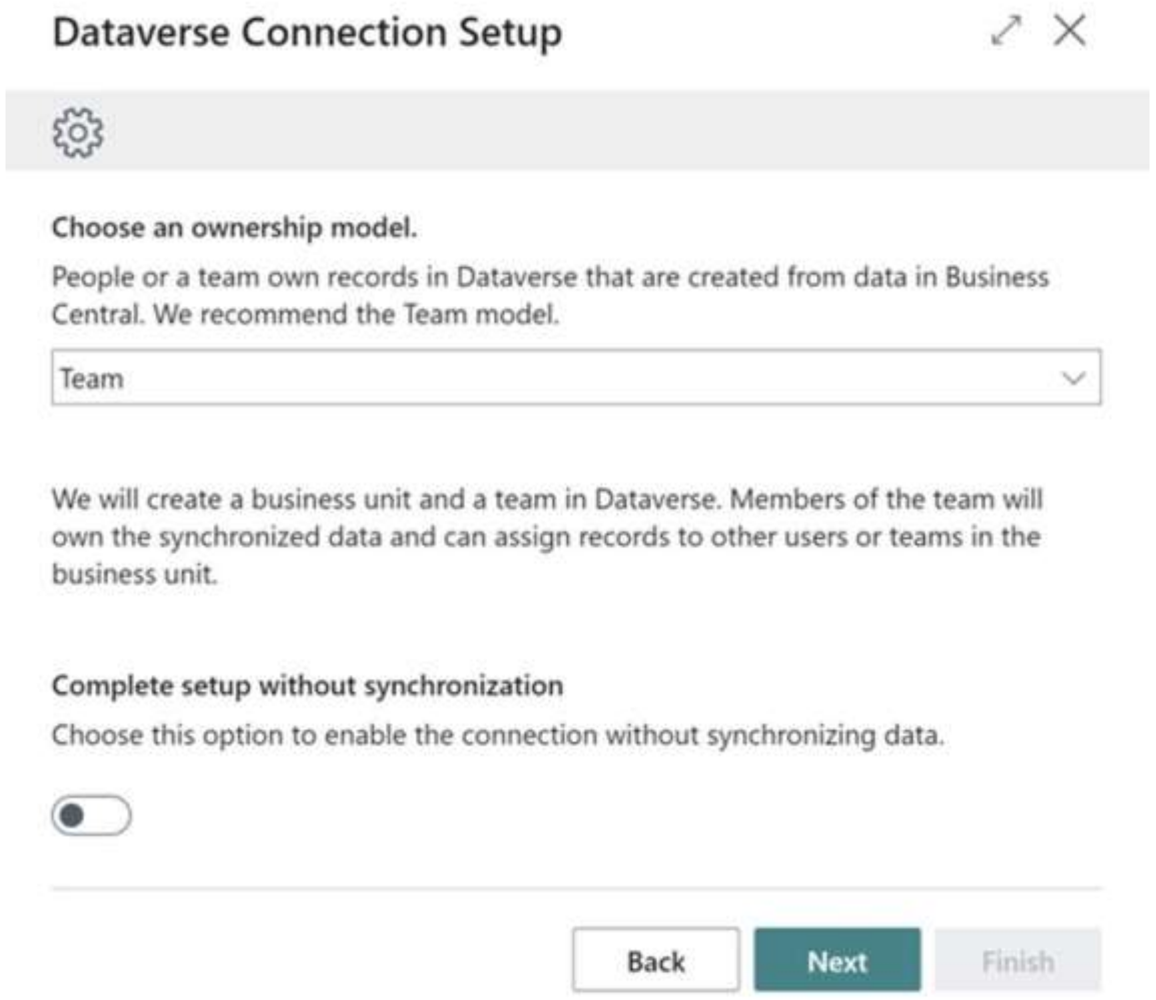
Sign in with an administrator user account and give consent to the application that will be used to connect to Dataverse. The account will be used one time to install and configure components that the integration requires.

The administrator is signed in.

The administrator is signed in.

To install and configure integration components, choose Next. This might take a few minutes.

## Connect BC to Dynamics 365 Sales


Dataverse Connection Setup

**Choose an ownership model.**

People or a team own records in Dataverse that are created from data in Business Central. We recommend the Team model.

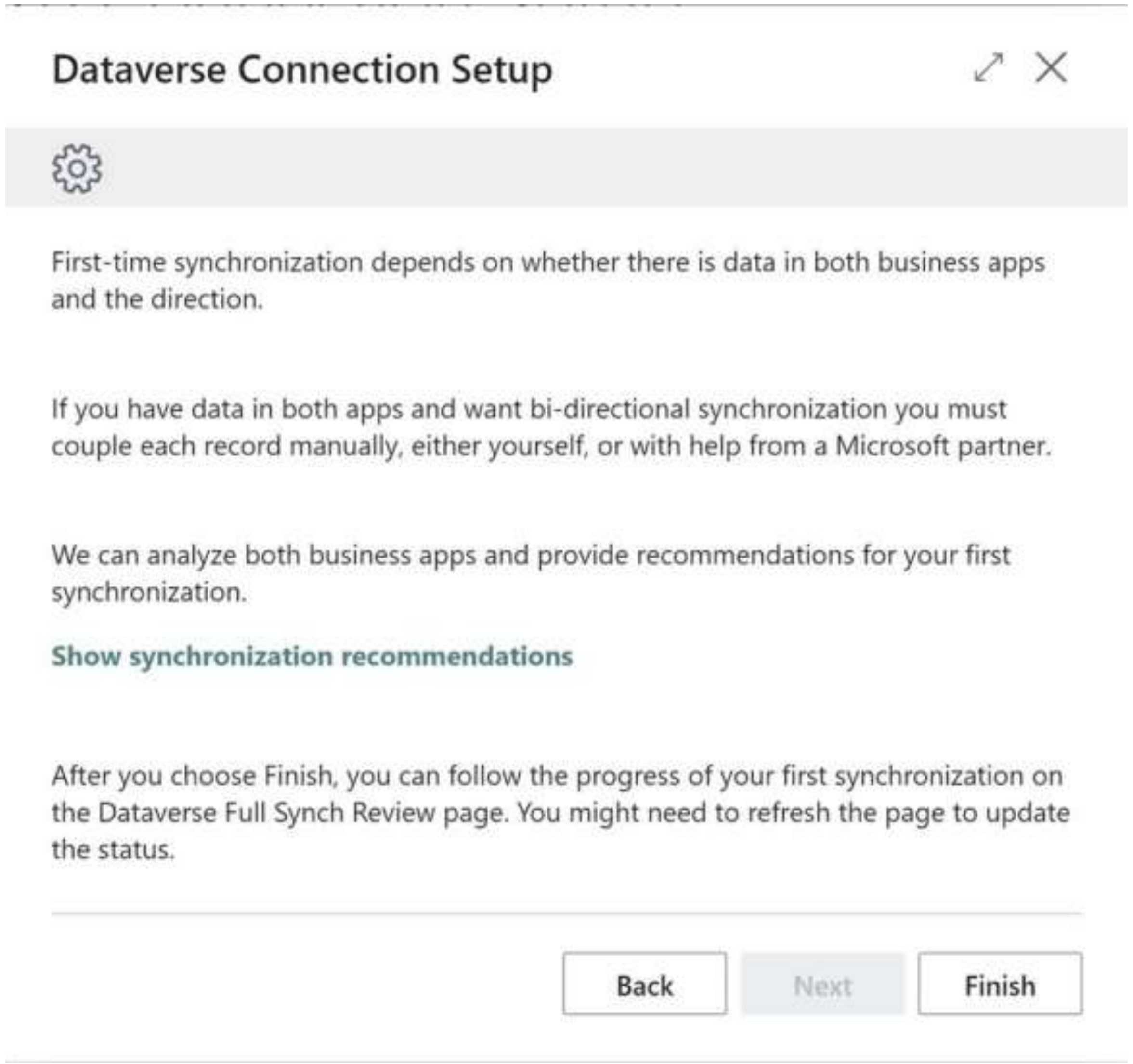
Team

We will create a business unit and a team in Dataverse. Members of the team will own the synchronized data and can assign records to other users or teams in the business unit.

**Complete setup without synchronization**

Choose this option to enable the connection without synchronizing data.

## Connect BC to Dynamics 365 Sales



Dataverse Connection Setup

First-time synchronization depends on whether there is data in both business apps and the direction.

If you have data in both apps and want bi-directional synchronization you must couple each record manually, either yourself, or with help from a Microsoft partner.

We can analyze both business apps and provide recommendations for your first synchronization.

**Show synchronization recommendations**

After you choose Finish, you can follow the progress of your first synchronization on the Dataverse Full Synch Review page. You might need to refresh the page to update the status.

OOTB

# Entity synchronization

## Entity synchronization

- Tables in Dynamics 365 Sales, such as orders, are integrated with equivalent types of tables in Business Central
- To work with Dynamics 365 Sales data you set up links, called couplings, between tables in Business Central and Dynamics 365 Sales.

# Entity synchronization

## Tables and direction of synchronization

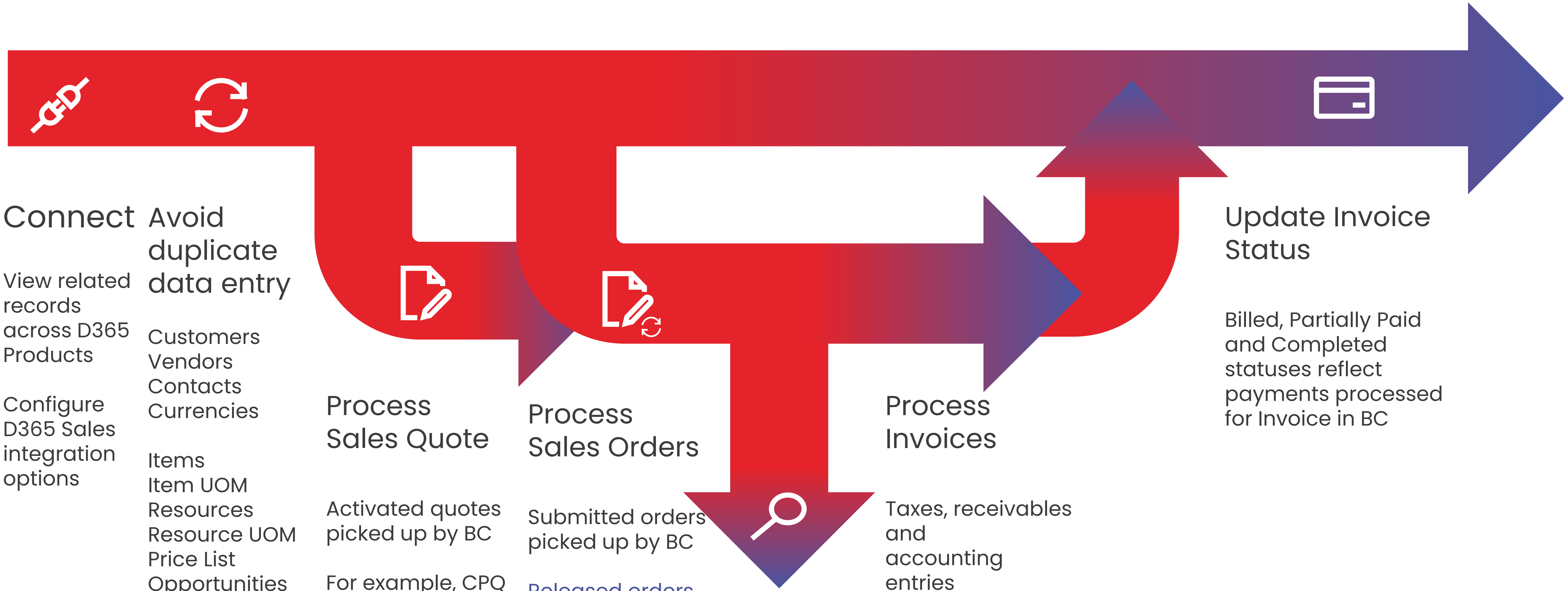
| Business Central | Dynamics 365 Sales | Synchronization Direction |
|---|---|---|
| Unit of Measure | Unit Group | Business Central -> Dynamics 365 Sales |
| Item | Product | Business Central -> Dynamics 365 Sales and Dynamics 365 Sales -> Business Central |
| Resource | Product | Business Central -> Dynamics 365 Sales and Dynamics 365 Sales -> Business Central |
| Item Unit of Measure | CRM UOM | Business Central -> Dynamics 365 Sales |
| Resource Unit of Measure | CRM UOM | Business Central -> Dynamics 365 Sales |
| Unit Group | CRM Uomschedule | Business Central -> Dynamics 365 Sales |
| Customer Price Group | Price List | Business Central -> Dynamics 365 Sales |
| Sales Price | Product Price List | Business Central -> Dynamics 365 Sales |
| Opportunity | Opportunity | Business Central -> Dataverse and Dynamics 365 Sales -> Business Central |
| Sales Invoice Header | Invoice | Business Central -> Dynamics 365 Sales |
| Sales Invoice Line | Invoice Product | Business Central -> Dynamics 365 Sales |
| Sales Order Header | Sales Order | Business Central -> Dynamics 365 Sales and Dynamics 365 Sales -> Business Central<br><br>To synchronize in both directions, you must turn on the **Bidirectional Synch of Sales Orders** toggle on the **Dynamics 365 Connection Setup** page. |
| Sales Order Notes | Sales Order Notes | Business Central -> Dynamics 365 Sales and Dynamics 365 Sales -> Business Central |

OOTB

## Sales Orders

directions lyon emea 2023

## Sales Quote & Sales Order Synchronization



### Connect

View related records across D365 Products

Configure D365 Sales integration options

### Avoid duplicate data entry

Customers
Vendors
Contacts
Currencies

Items
Item UOM
Resources
Resource UOM
Price List
Opportunities

### Process Sales Quote

Activated quotes picked up by BC

For example, CPQ solution built in Business Central

### Process Sales Orders

Submitted orders picked up by BC

Released orders can be sent to D365 Sales*

For example, CPQ solution built for D365 Sales

### View Item Availability

Based on inventory in BC

### Process Invoices

Taxes, receivables and accounting entries handled by BC

### Update Invoice Status

Billed, Partially Paid and Completed statuses reflect payments processed for Invoice in BC

slido

## Sales Order integration

ⓘ Start presenting to display the poll results on this slide.

slido

## Did you have any issues? If yes, what were the main problems?

ⓘ Start presenting to display the poll results on this slide.

slido

## Audience Q&A Session

ⓘ Start presenting to display the audience questions on this slide.

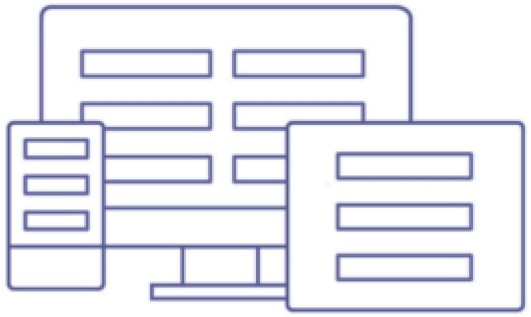
## Customizations

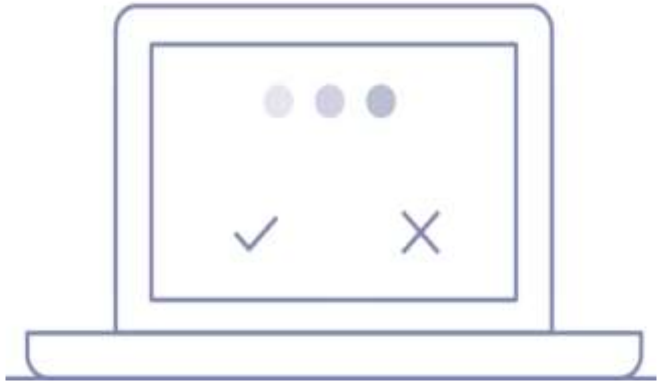
## Power Automate

## What is Power Automate

directions lyon emea 2023

Power Automate is a service for automating workflows across apps and services





Connect to data & systems you're already using; create the data you need

Create workflows using triggers & actions without code or scripts

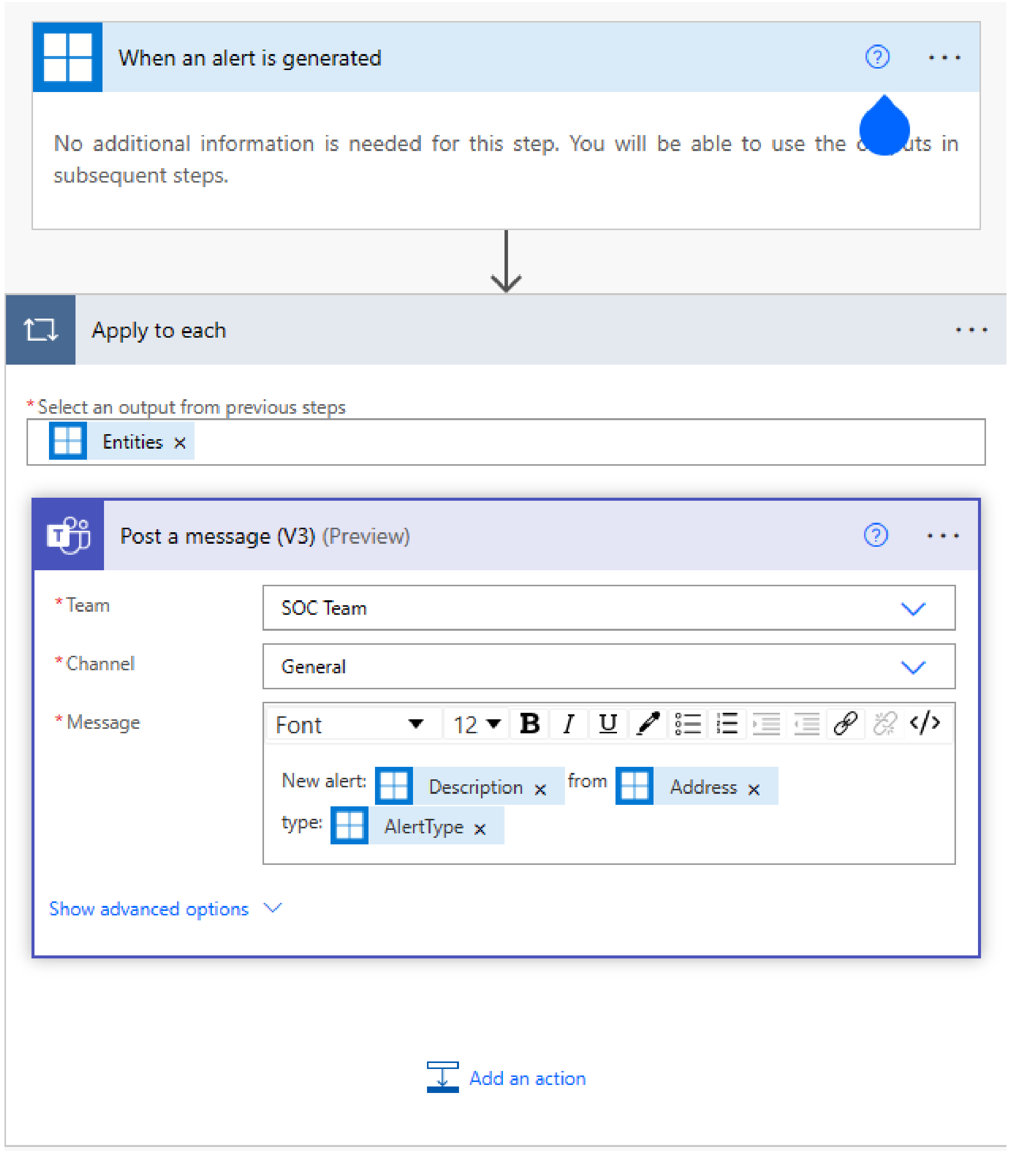
Edit flows on web and mobile

Approve requests or manage them on web and mobile

# What is Power Automate

directions lyon emea 2023

- Power Automate is an online workflow service that automates actions across the most common apps and services
- Power Automate is part of the Microsoft Power Platform



When an alert is generated

No additional information is needed for this step. You will be able to use the outputs in subsequent steps.

Apply to each

* Select an output from previous steps: Entities ×

Post a message (V3) (Preview)

* Team: SOC Team

* Channel: General

* Message: New alert: Description × from Address × type: AlertType ×

# What makes up a Power Automate?

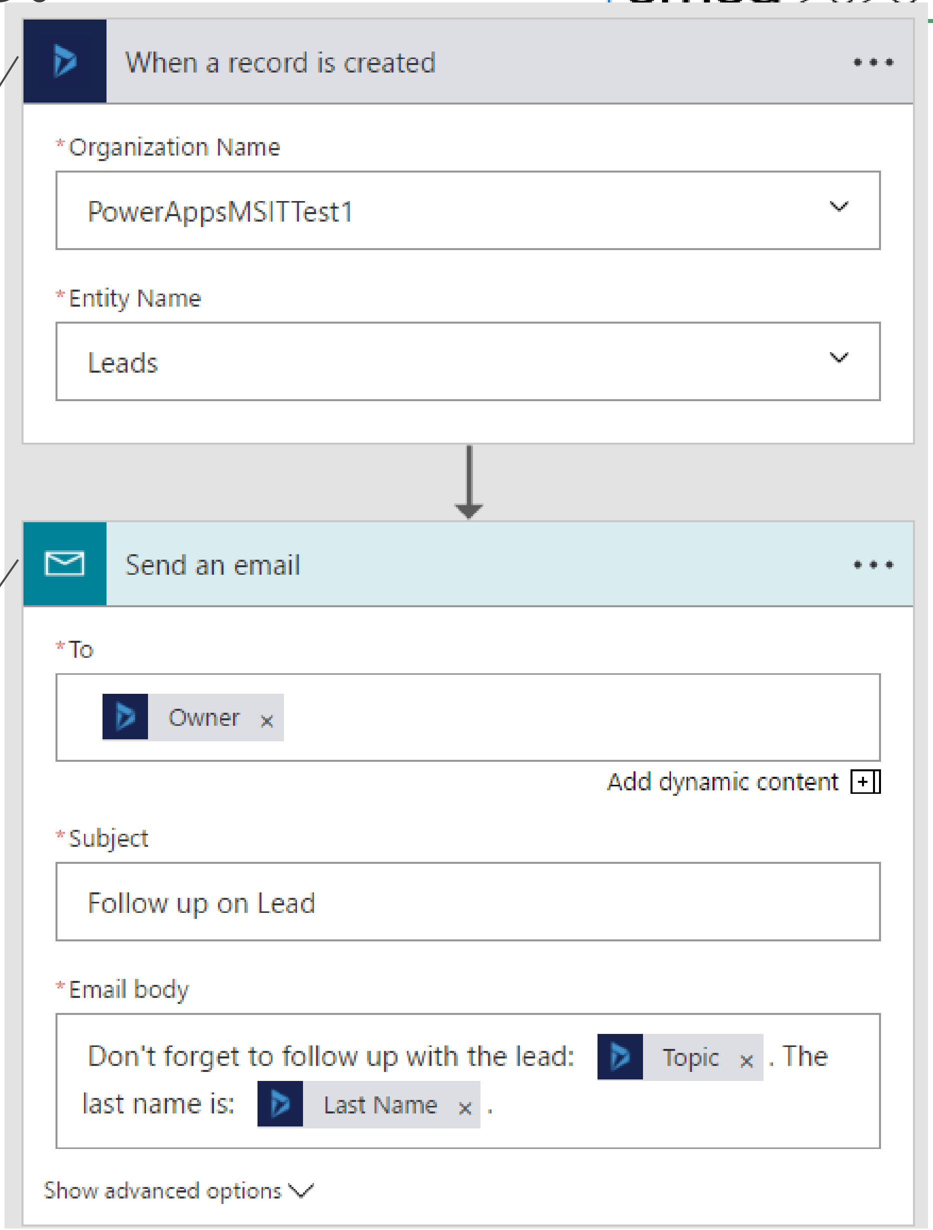
Example: Notification Flow

Trigger–the event that kicks off the flow:

- Manual via button or PowerApps
- On a schedule
- On an event in the cloud

Actions–what the flow does

*Uses data from the trigger*



When a record is created

*Organization Name
PowerAppsMSITTest1

*Entity Name
Leads

Send an email

*To
Owner

Add dynamic content

*Subject
Follow up on Lead

*Email body
Don't forget to follow up with the lead: Topic . The last name is: Last Name .

Show advanced options

## What value Power Automate brings to you

- Common scenarios and capabilities of Power Automate are:
- Automating repetitive tasks like moving data from one system to another.
- Guiding a user through a process so they can complete the different stages.
- Connecting to external data sources via one of the hundreds of connectors or directly via an API.
- Automating desktop-based processes with robotic process automation (RPA) capabilities.

Customizations

directions lyon emea 2023

## Power Automate

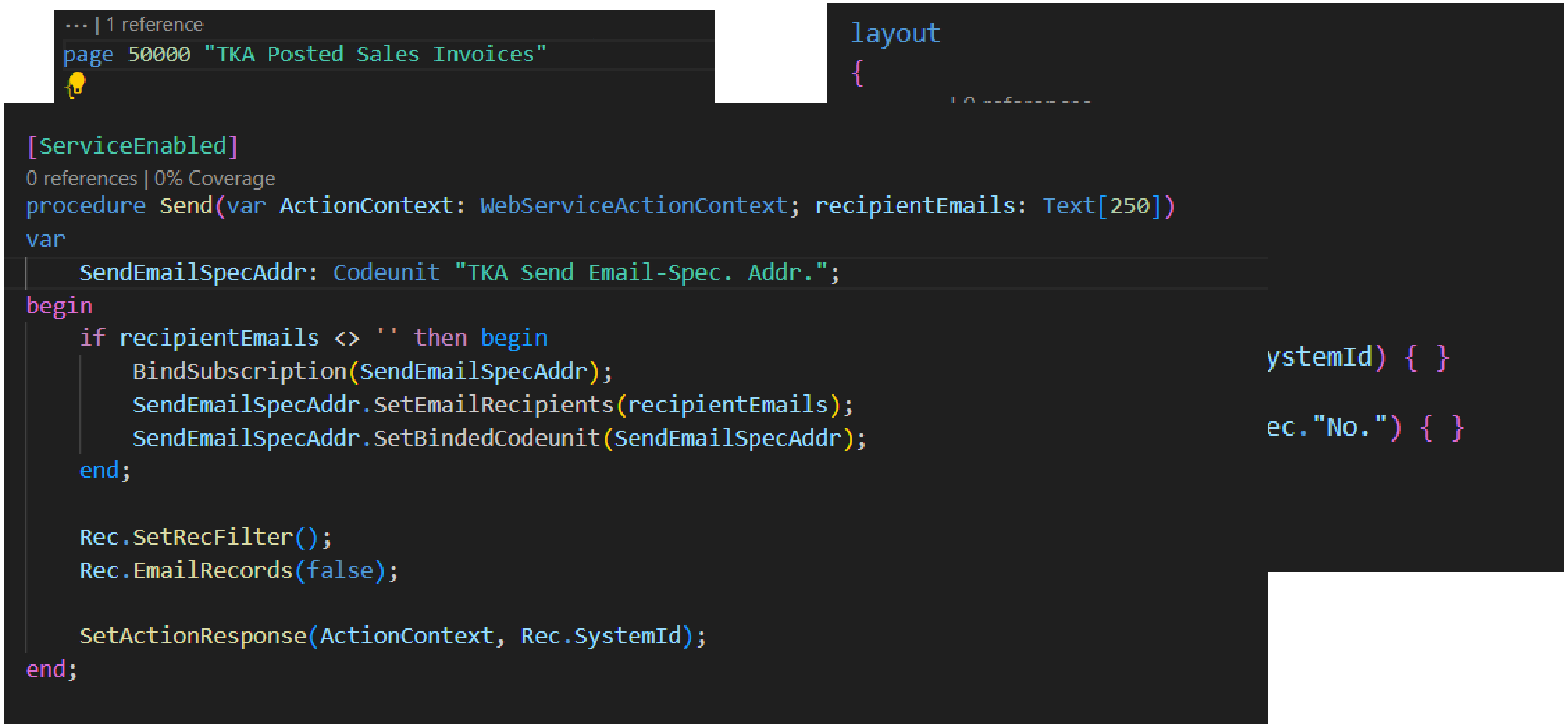
What can we do?

directions lyon emea 2023

DEMO TIME

## Power Automate

What can we do – anything! Let's send an invoice from CRM.



```
··· | 1 reference
page 50000 "TKA Posted Sales Invoices"
```

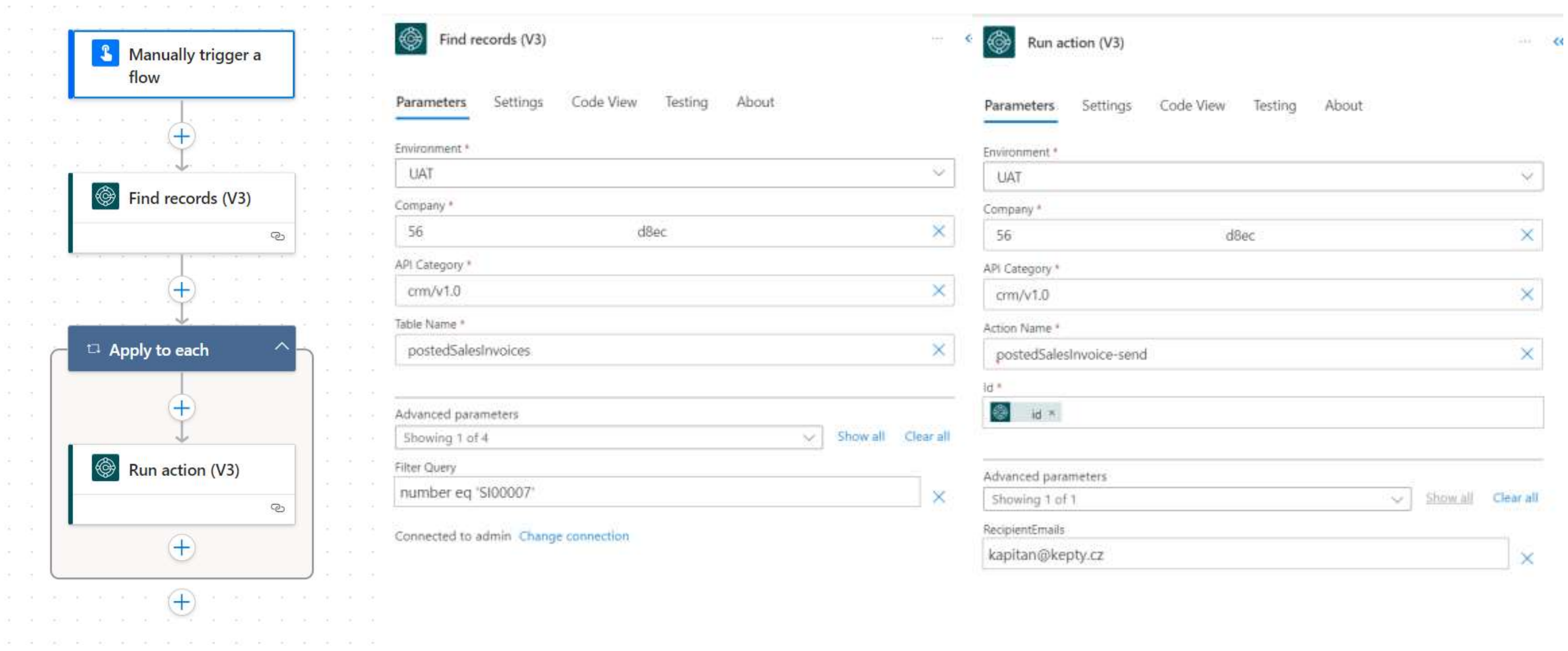
```
layout
{
```

```
ystemId) { }

ec."No.") { }
```

```
[ServiceEnabled]
0 references | 0% Coverage
procedure Send(var ActionContext: WebServiceActionContext; recipientEmails: Text[250])
var
    SendEmailSpecAddr: Codeunit "TKA Send Email-Spec. Addr.";
begin
    if recipientEmails <> '' then begin
        BindSubscription(SendEmailSpecAddr);
        SendEmailSpecAddr.SetEmailRecipients(recipientEmails);
        SendEmailSpecAddr.SetBindedCodeunit(SendEmailSpecAddr);
    end;

    Rec.SetRecFilter();
    Rec.EmailRecords(false);

    SetActionResponse(ActionContext, Rec.SystemId);
end;
```

## Power Automate

What can we do – anything! Let's send an invoice from CRM.



Find records (V3)

Parameters Settings Code View Testing About

Environment *
UAT

Company *
56 d8ec

API Category *
crm/v1.0

Table Name *
postedSalesInvoices

Advanced parameters
Showing 1 of 4 Show all Clear all

Filter Query
number eq 'SI00007'

Connected to admin Change connection

Run action (V3)

Parameters Settings Code View Testing About

Environment *
UAT

Company *
56 d8ec

API Category *
crm/v1.0

Action Name *
postedSalesInvoice-send

Id *
id

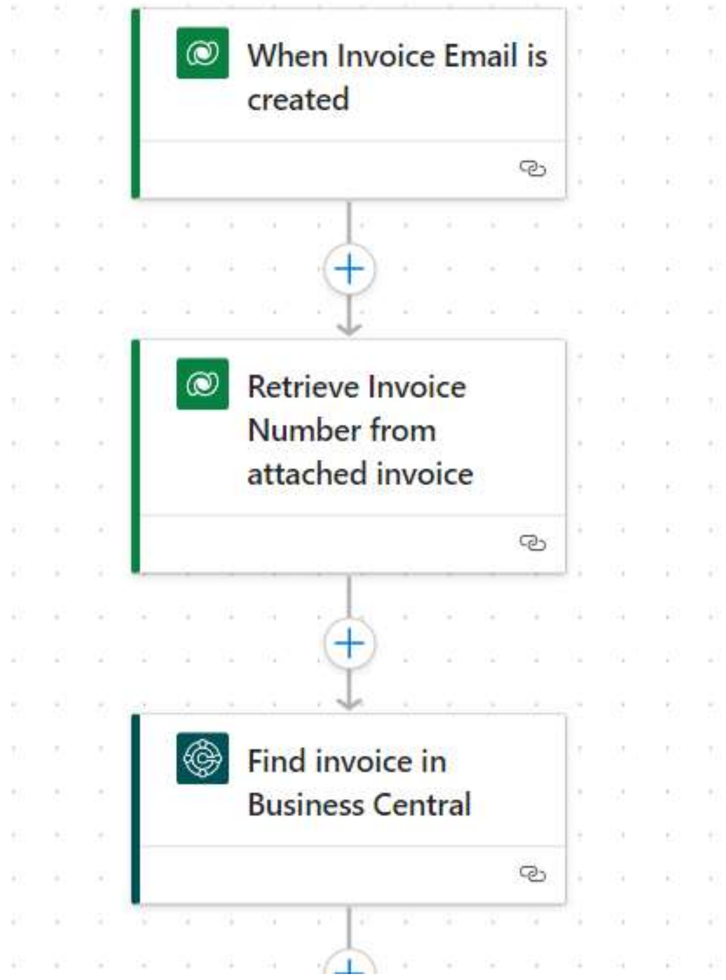
Advanced parameters
Showing 1 of 1 Show all Clear all

RecipientEmails
kapitan@kepty.cz

## Power Automate

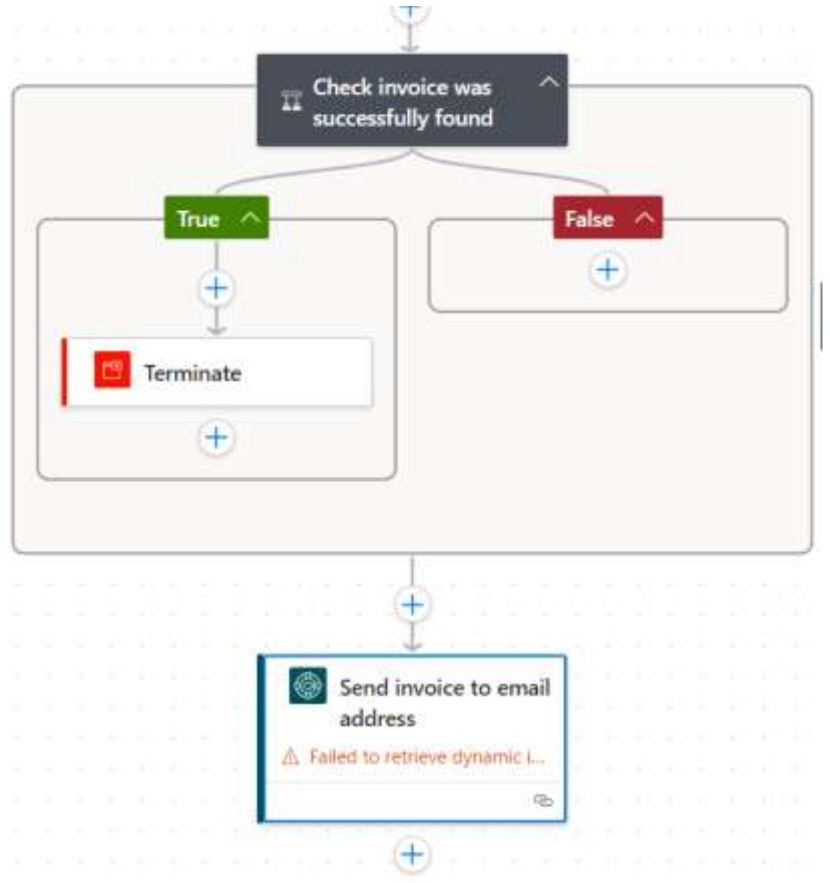
What can we do – anything! Let's send an invoice from CRM.

directions lyon emea 2023


When Invoice Email is created

Retrieve Invoice Number from attached invoice

Find invoice in Business Central


Check invoice was successfully found

True

Terminate

False

Send invoice to email address

⚠ Failed to retrieve dynamic i...

slido

## What are your ideas when to use the Power Automate to integrate CRM and BC?

ⓘ Start presenting to display the poll results on this slide.

directions lyon
emea 2023

## Customizations

slido

## How many D365BC + Sales (incl. other Field Service etc) integrations have you done?

ⓘ Start presenting to display the poll results on this slide.

slido

## Which entities/tables/parts of the integration did you customize in the past?

ⓘ Start presenting to display the poll results on this slide.

directions lyon emea 2023

Customizations

## Custom Table & Fields

directions lyon emea 2023

# Create integration table

## Using AL Table Proxy Generator tool (altpgen)

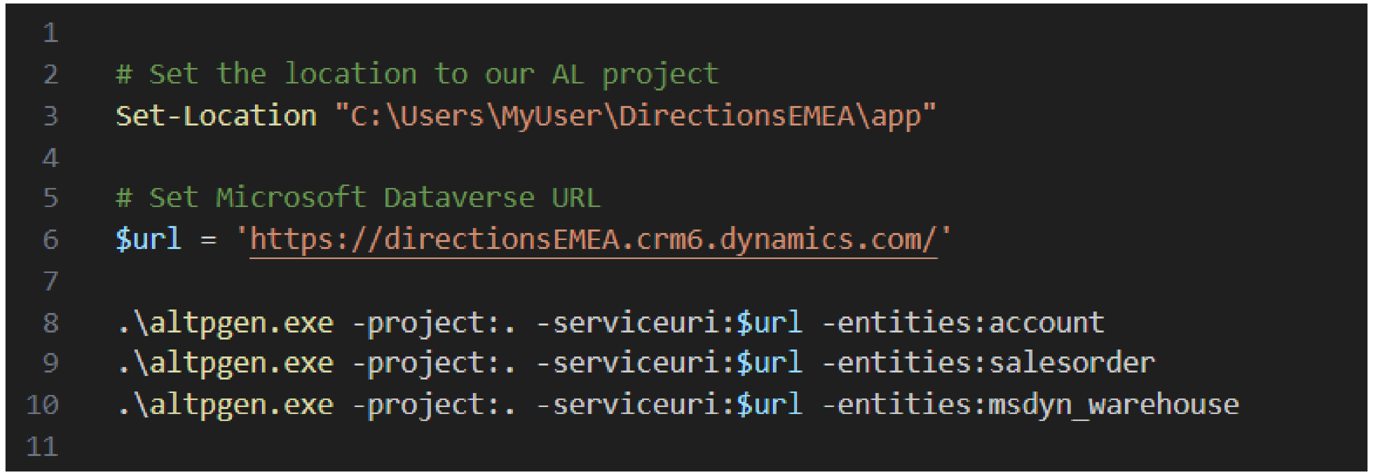
Table (and Table extension) can be generated automatically.

Required tool (altpgen) can be found

- C:\Users\<username>\.vscode\extensions\ms-dynamics-smb.al-<ALExtensionVersion>\bin\altpgen.exe

### Important Parameters

- **Project**: AL Project location
- **ServiceUri**: Microsoft Dataverse URL
- **Entities**: Name(s) of required entities
- **PackageCachePath**: AL Project cache location



```

# Set the location to our AL project
Set-Location "C:\Users\MyUser\DirectionsEMEA\app"

# Set Microsoft Dataverse URL
$url = 'https://directionsEMEA.crm6.dynamics.com/'

.\altpgen.exe -project:. -serviceuri:$url -entities:account
.\altpgen.exe -project:. -serviceuri:$url -entities:salesorder
.\altpgen.exe -project:. -serviceuri:$url -entities:msdyn_warehouse

```

More details
[Customizing an Integration with Microsoft Dataverse - BC| Microsoft Learn]()

## Scenario (example)

Add custom Table and Field

**Requirements**

We want to see and be able to change Item Category assigned to Item record in CRM.

**Design**

Add “Product Category” to “Product” CRM entity.
Add “Product Category” as a custom CRM entity.

**Changes**

New field, new table, synchronization logic for the newly created field, synchronization logic for newly created table

# Add custom table and field

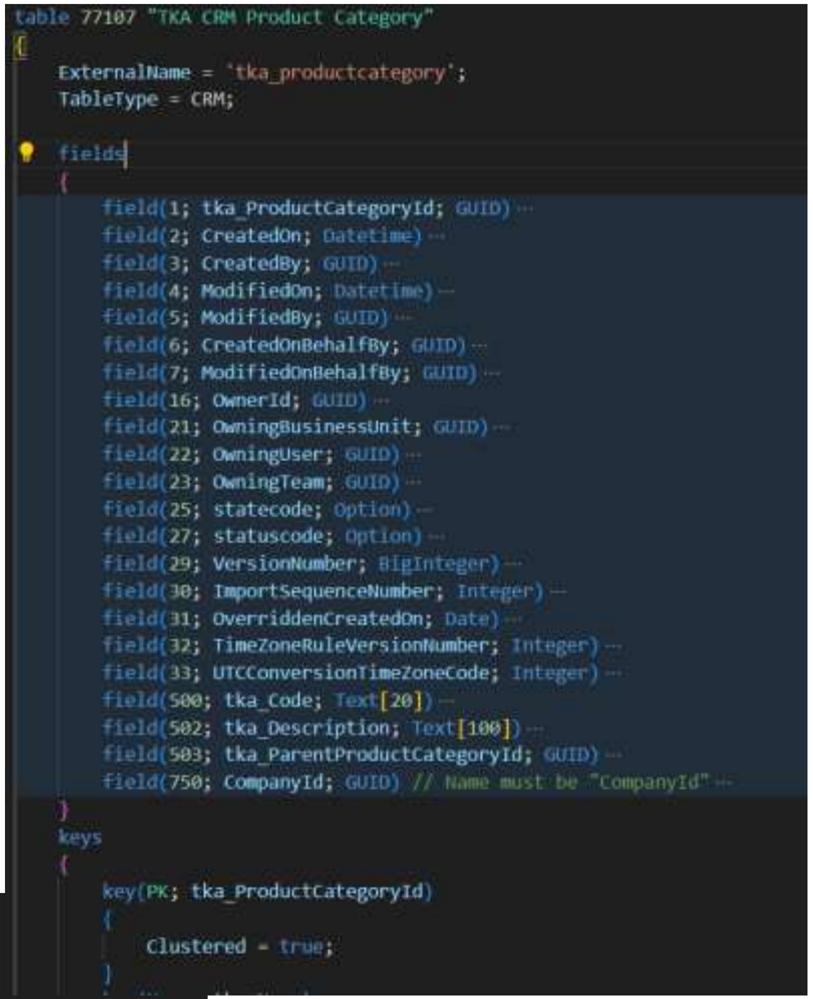
Structure

**In CRM**

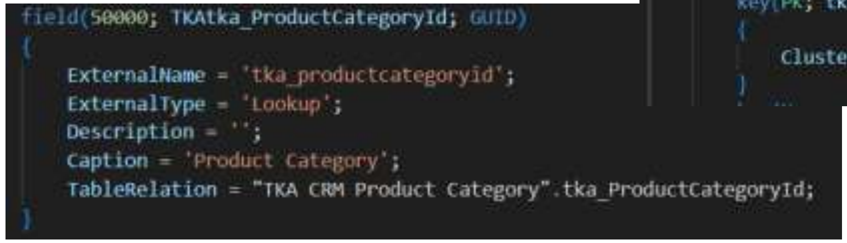
1) Create the new entity “Product Category” in CRM
2) Create the new field “Product Category Id” in CRM and add a relation to “Product Category” entity.

**In BC**

1) Run **altpgen.exe** for both
    1) “Product Category” entity to generate “Product Category” table
    2) “Product” entity to generate “Product” table extension



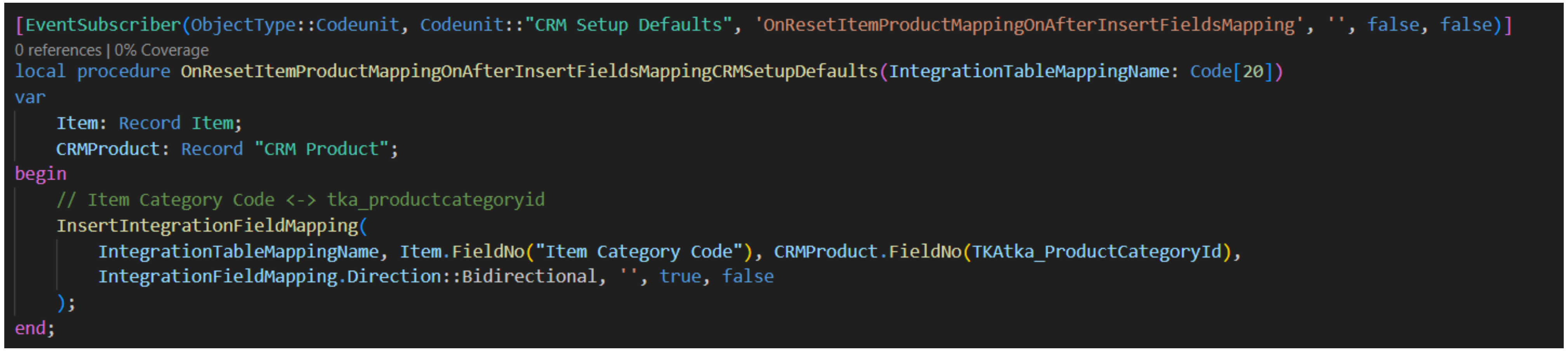
```
table 77107 "TKA CRM Product Category"
{
    ExternalName = 'tka_productcategory';
    TableType = CRM;

    fields
    {
        field(1; tka_ProductCategoryId; GUID)···
        field(2; CreatedOn; Datetime)···
        field(3; CreatedBy; GUID)···
        field(4; ModifiedOn; Datetime)···
        field(5; ModifiedBy; GUID)···
        field(6; CreatedOnBehalfBy; GUID)···
        field(7; ModifiedOnBehalfBy; GUID)···
        field(16; OwnerId; GUID)···
        field(21; OwningBusinessUnit; GUID)···
        field(22; OwningUser; GUID)···
        field(23; OwningTeam; GUID)···
        field(25; statecode; Option)···
        field(27; statuscode; Option)···
        field(29; VersionNumber; BigInteger)···
        field(30; ImportSequenceNumber; Integer)···
        field(31; OverriddenCreatedOn; Date)···
        field(32; TimeZoneRuleVersionNumber; Integer)···
        field(33; UTCConversionTimeZoneCode; Integer)···
        field(500; tka_Code; Text[20])···
        field(502; tka_Description; Text[100])···
        field(503; tka_ParentProductCategoryId; GUID)···
        field(750; CompanyId; GUID) // Name must be "CompanyId"···
    }
    keys
    {
        key(PK; tka_ProductCategoryId)
        {
            Clustered = true;
        }
```



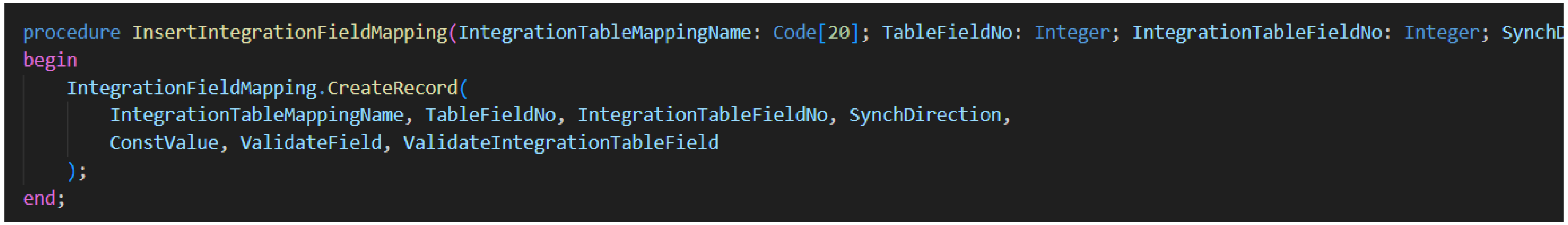
```
field(50000; TKAtka_ProductCategoryId; GUID)
{
    ExternalName = 'tka_productcategoryid';
    ExternalType = 'Lookup';
    Description = '';
    Caption = 'Product Category';
    TableRelation = "TKA CRM Product Category".tka_ProductCategoryId;
}
```

# Add custom table and field

Product field mapping logic



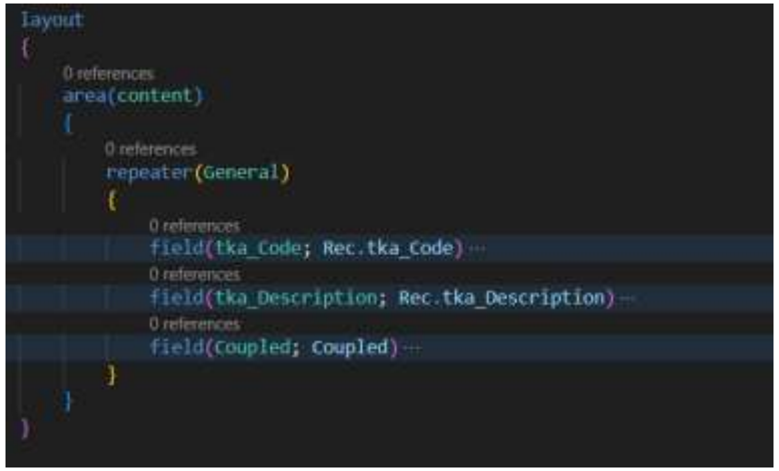
```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"CRM Setup Defaults", 'OnResetItemProductMappingOnAfterInsertFieldsMapping', '', false, false)]
0 references | 0% Coverage
local procedure OnResetItemProductMappingOnAfterInsertFieldsMappingCRMSetupDefaults(IntegrationTableMappingName: Code[20])
var
    Item: Record Item;
    CRMProduct: Record "CRM Product";
begin
    // Item Category Code <-> tka_productcategoryid
    InsertIntegrationFieldMapping(
        IntegrationTableMappingName, Item.FieldNo("Item Category Code"), CRMProduct.FieldNo(TKAtka_ProductCategoryId),
        IntegrationFieldMapping.Direction::Bidirectional, '', true, false
    );
end;
```



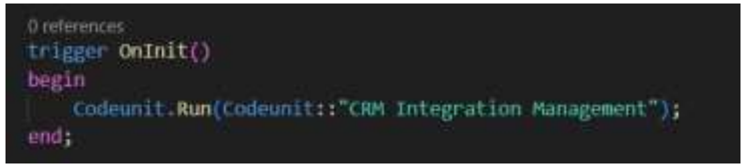
```
procedure InsertIntegrationFieldMapping(IntegrationTableMappingName: Code[20]; TableFieldNo: Integer; IntegrationTableFieldNo: Integer; SynchD
begin
    IntegrationFieldMapping.CreateRecord(
        IntegrationTableMappingName, TableFieldNo, IntegrationTableFieldNo, SynchDirection,
        ConstValue, ValidateField, ValidateIntegrationTableField
    );
end;
```

# Add custom table and field

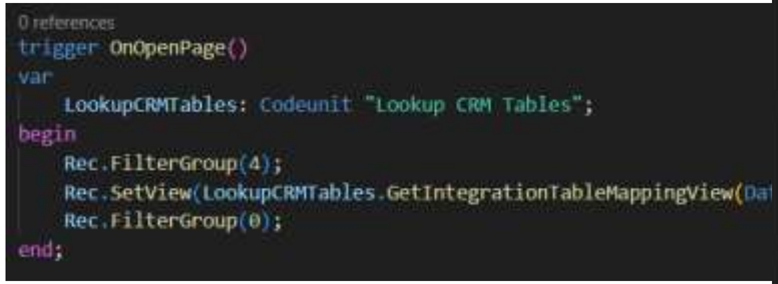
CRM Product Category page



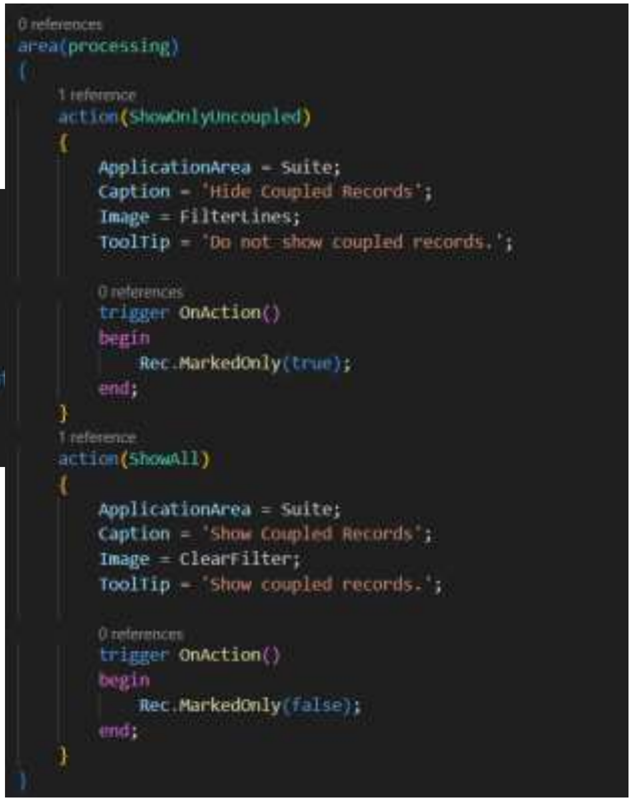
```
layout
{
    0 references
    area(content)
    {
        0 references
        repeater(General)
        {
            0 references
            field(tka_Code; Rec.tka_Code) ···
            0 references
            field(tka_Description; Rec.tka_Description) ···
            0 references
            field(Coupled; Coupled) ···
        }
    }
}
```



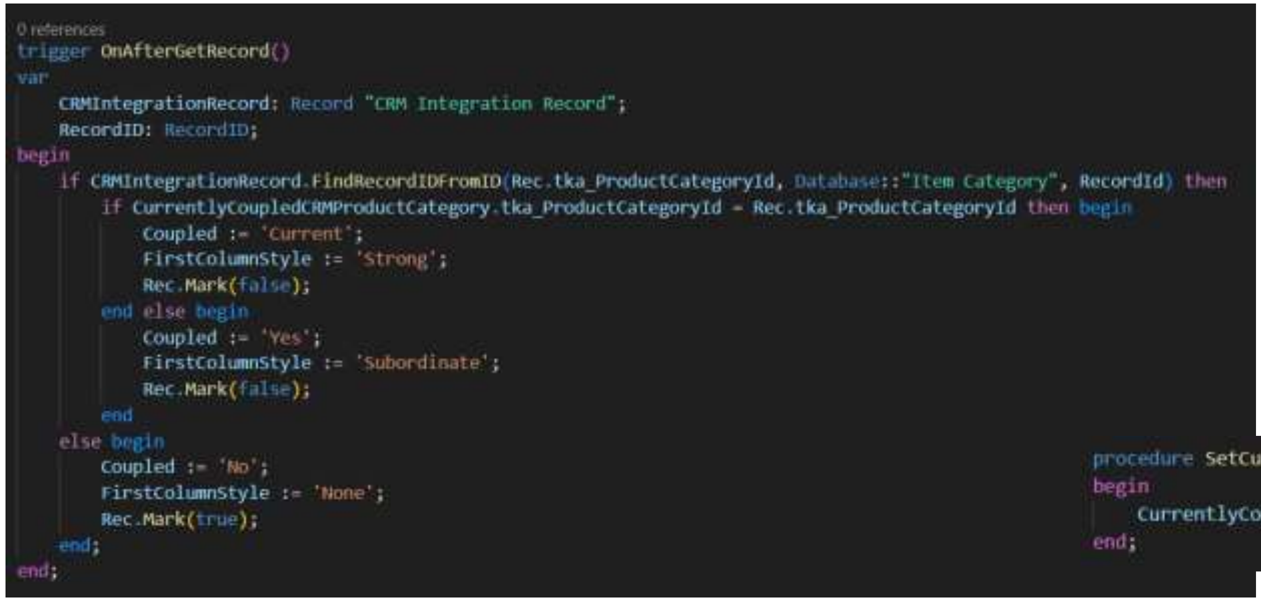
```
0 references
trigger OnInit()
begin
    Codeunit.Run(Codeunit::"CRM Integration Management");
end;
```



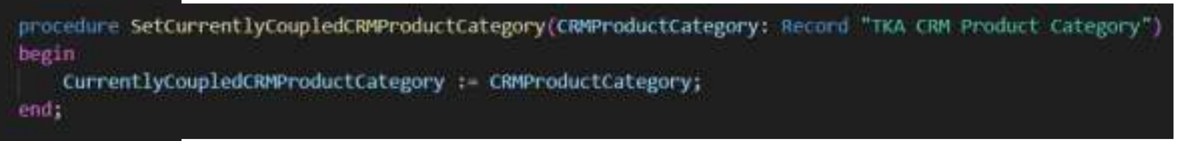
```
0 references
trigger OnOpenPage()
var
    LookupCRMTables: Codeunit "Lookup CRM Tables";
begin
    Rec.FilterGroup(4);
    Rec.SetView(LookupCRMTables.GetIntegrationTableMappingView(Da
    Rec.FilterGroup(0);
end;
```



```
0 references
area(processing)
{
    1 reference
    action(ShowOnlyUncoupled)
    {
        ApplicationArea = Suite;
        Caption = 'Hide Coupled Records';
        Image = FilterLines;
        ToolTip = 'Do not show coupled records.';

        0 references
        trigger OnAction()
        begin
            Rec.MarkedOnly(true);
        end;
    }
    1 reference
    action(ShowAll)
    {
        ApplicationArea = Suite;
        Caption = 'Show Coupled Records';
        Image = ClearFilter;
        ToolTip = 'Show coupled records.';

        0 references
        trigger OnAction()
        begin
            Rec.MarkedOnly(false);
        end;
    }
}
```



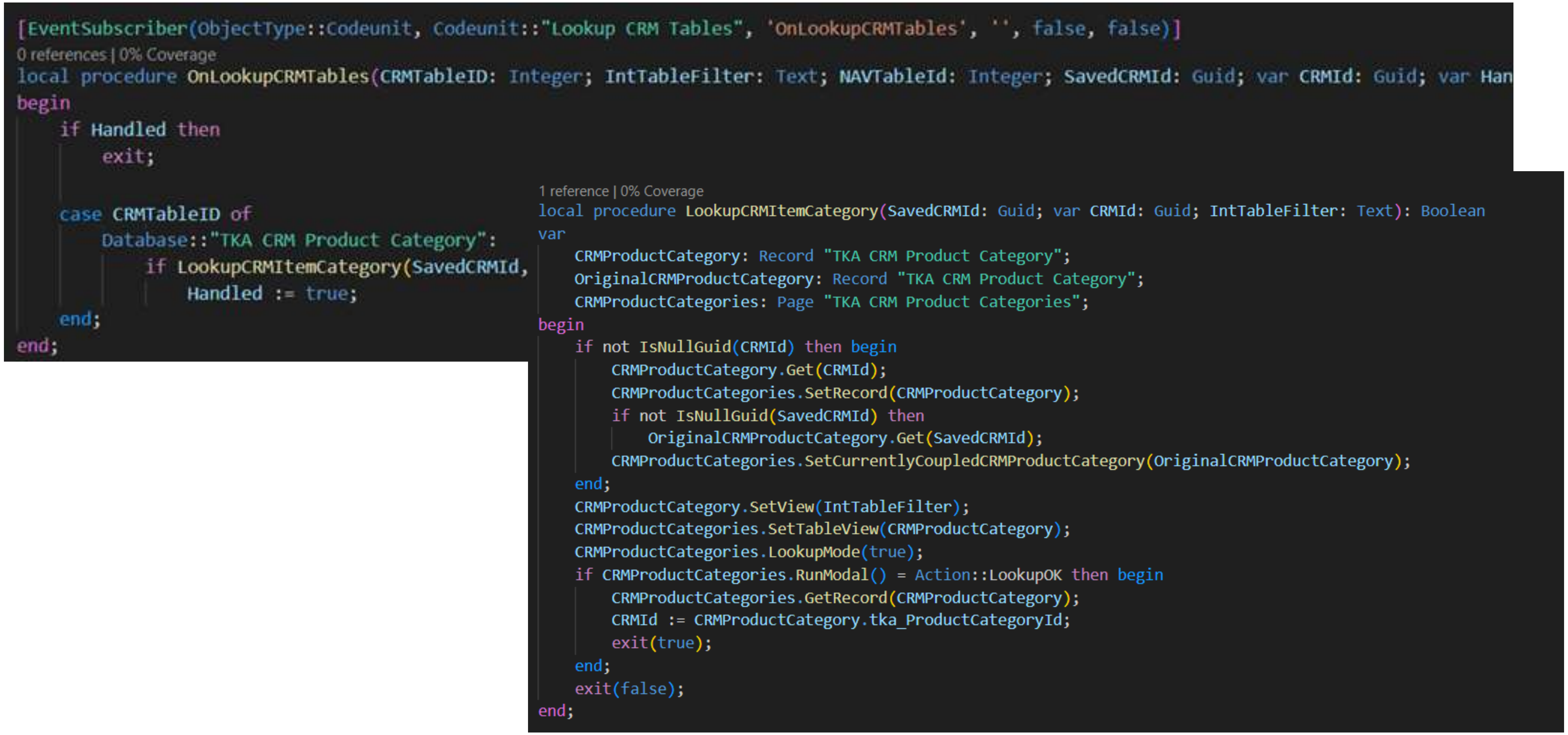
```
0 references
trigger OnAfterGetRecord()
var
    CRMIntegrationRecord: Record "CRM Integration Record";
    RecordID: RecordID;
begin
    if CRMIntegrationRecord.FindRecordIDFromID(Rec.tka_ProductCategoryId, Database::"Item Category", RecordId) then
        if CurrentlyCoupledCRMProductCategory.tka_ProductCategoryId = Rec.tka_ProductCategoryId then begin
            Coupled := 'Current';
            FirstColumnStyle := 'Strong';
            Rec.Mark(false);
        end else begin
            Coupled := 'Yes';
            FirstColumnStyle := 'Subordinate';
            Rec.Mark(false);
        end
    else begin
        Coupled := 'No';
        FirstColumnStyle := 'None';
        Rec.Mark(true);
    end;
end;
```



```
procedure SetCurrentlyCoupledCRMProductCategory(CRMProductCategory: Record "TKA CRM Product Category")
begin
    CurrentlyCoupledCRMProductCategory := CRMProductCategory;
end;
```

# Add custom table and field

Lookup CRM Table

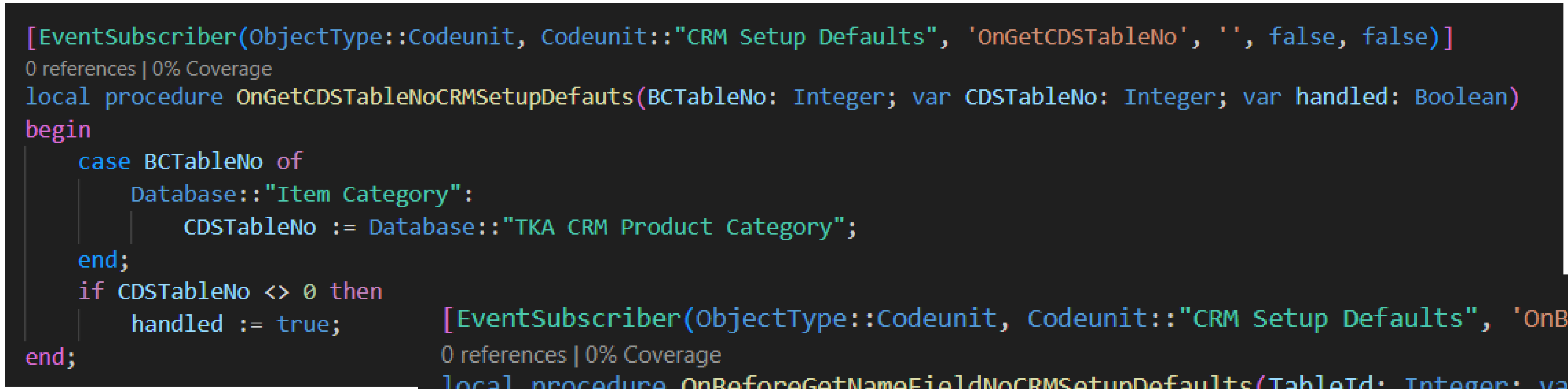


```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"Lookup CRM Tables", 'OnLookupCRMTables', '', false, false)]
0 references | 0% Coverage
local procedure OnLookupCRMTables(CRMTableID: Integer; IntTableFilter: Text; NAVTableId: Integer; SavedCRMId: Guid; var CRMId: Guid; var Han
begin
    if Handled then
        exit;

    case CRMTableID of
        Database::"TKA CRM Product Category":
            if LookupCRMItemCategory(SavedCRMId,
                Handled := true;
    end;
end;
```

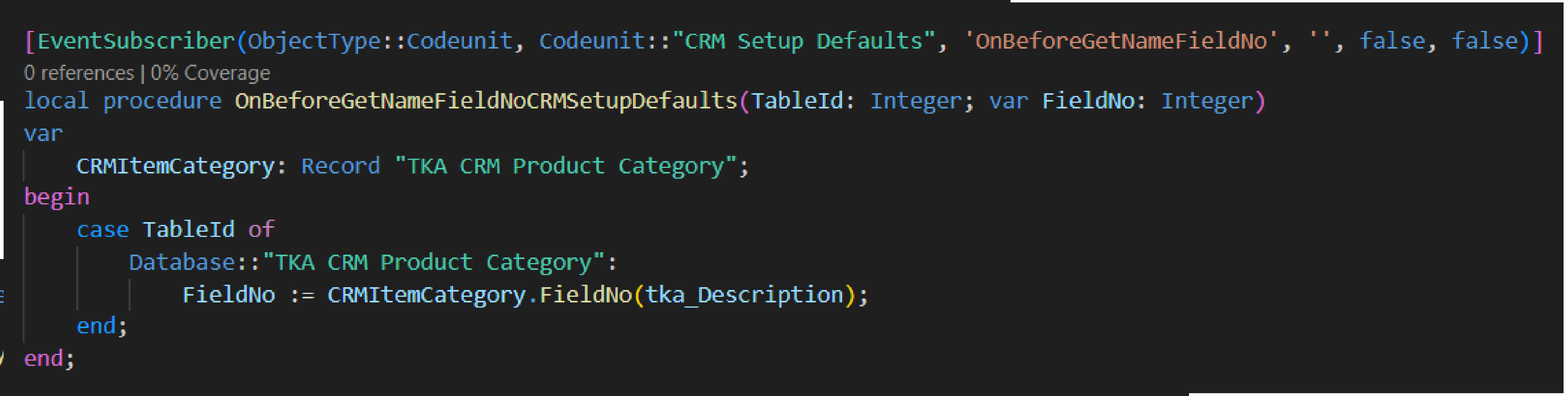
```
1 reference | 0% Coverage
local procedure LookupCRMItemCategory(SavedCRMId: Guid; var CRMId: Guid; IntTableFilter: Text): Boolean
var
    CRMProductCategory: Record "TKA CRM Product Category";
    OriginalCRMProductCategory: Record "TKA CRM Product Category";
    CRMProductCategories: Page "TKA CRM Product Categories";
begin
    if not IsNullGuid(CRMId) then begin
        CRMProductCategory.Get(CRMId);
        CRMProductCategories.SetRecord(CRMProductCategory);
        if not IsNullGuid(SavedCRMId) then
            OriginalCRMProductCategory.Get(SavedCRMId);
        CRMProductCategories.SetCurrentlyCoupledCRMProductCategory(OriginalCRMProductCategory);
    end;
    CRMProductCategory.SetView(IntTableFilter);
    CRMProductCategories.SetTableView(CRMProductCategory);
    CRMProductCategories.LookupMode(true);
    if CRMProductCategories.RunModal() = Action::LookupOK then begin
        CRMProductCategories.GetRecord(CRMProductCategory);
        CRMId := CRMProductCategory.tka_ProductCategoryId;
        exit(true);
    end;
    exit(false);
end;
```

## Add custom table and field

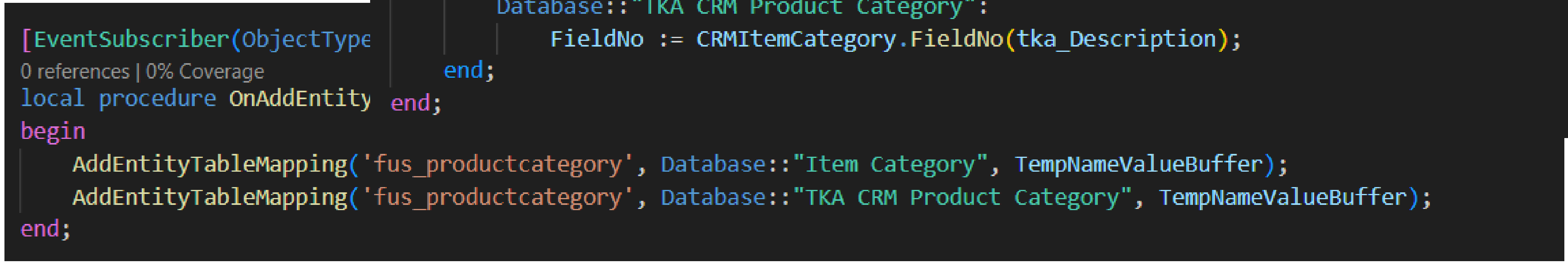
Lookup CRM Table



```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"CRM Setup Defaults", 'OnGetCDSTableNo', '', false, false)]
0 references | 0% Coverage
local procedure OnGetCDSTableNoCRMSetupDefauts(BCTableNo: Integer; var CDSTableNo: Integer; var handled: Boolean)
begin
    case BCTableNo of
        Database::"Item Category":
            CDSTableNo := Database::"TKA CRM Product Category";
    end;
    if CDSTableNo <> 0 then
        handled := true;
end;
```



```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"CRM Setup Defaults", 'OnBeforeGetNameFieldNo', '', false, false)]
0 references | 0% Coverage
local procedure OnBeforeGetNameFieldNoCRMSetupDefaults(TableId: Integer; var FieldNo: Integer)
var
    CRMItemCategory: Record "TKA CRM Product Category";
begin
    case TableId of
        Database::"TKA CRM Product Category":
            FieldNo := CRMItemCategory.FieldNo(tka_Description);
    end;
end;
```

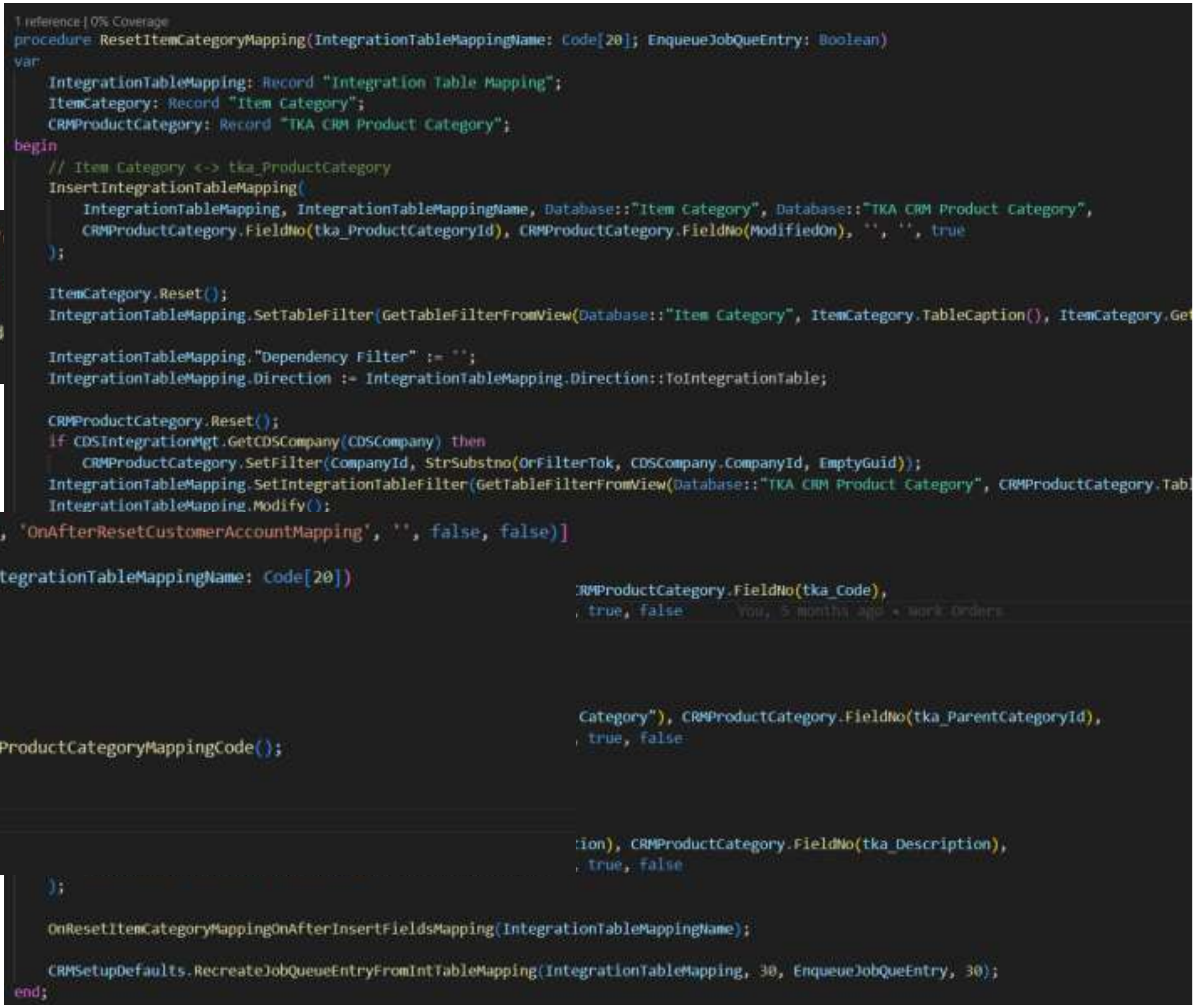


```
[EventSubscriber(ObjectType
0 references | 0% Coverage
local procedure OnAddEntity
begin
    AddEntityTableMapping('fus_productcategory', Database::"Item Category", TempNameValueBuffer);
    AddEntityTableMapping('fus_productcategory', Database::"TKA CRM Product Category", TempNameValueBuffer);
end;
```

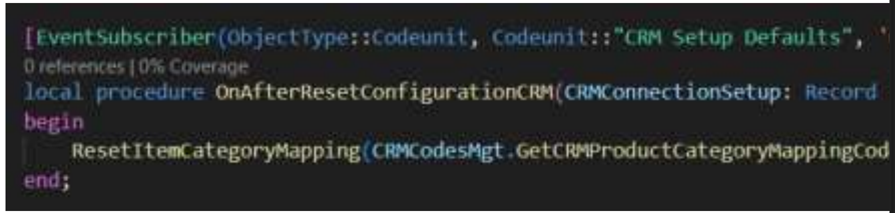
# Add custom table and field

## Table Mapping

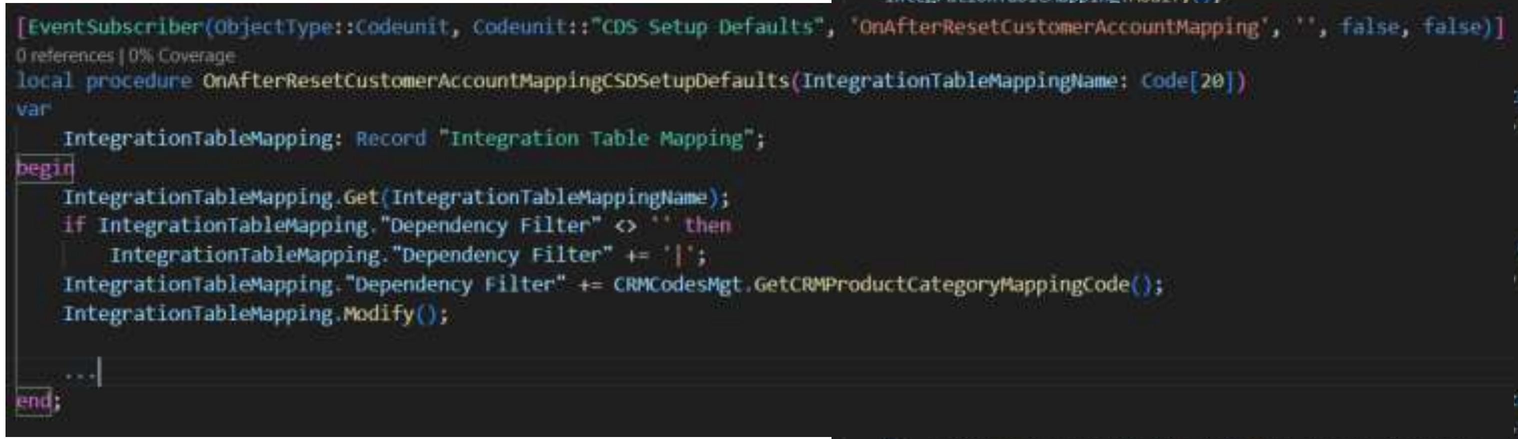
directions lyon emea 2023



```
1 reference | 0% Coverage
procedure ResetItemCategoryMapping(IntegrationTableMappingName: Code[20]; EnqueueJobQueEntry: Boolean)
var
    IntegrationTableMapping: Record "Integration Table Mapping";
    ItemCategory: Record "Item Category";
    CRMProductCategory: Record "TKA CRM Product Category";
begin
    // Item Category <-> tka_ProductCategory
    InsertIntegrationTableMapping(
        IntegrationTableMapping, IntegrationTableMappingName, Database::"Item Category", Database::"TKA CRM Product Category",
        CRMProductCategory.FieldNo(tka_ProductCategoryId), CRMProductCategory.FieldNo(ModifiedOn), '', '', true
    );

    ItemCategory.Reset();
    IntegrationTableMapping.SetTableFilter(GetTableFilterFromView(Database::"Item Category", ItemCategory.TableCaption(), ItemCategory.Get

    IntegrationTableMapping."Dependency Filter" := '';
    IntegrationTableMapping.Direction := IntegrationTableMapping.Direction::ToIntegrationTable;

    CRMProductCategory.Reset();
    if CDSIntegrationMgt.GetCDSCompany(CDSCompany) then
        CRMProductCategory.SetFilter(CompanyId, StrSubstno(OrFilterTok, CDSCompany.CompanyId, EmptyGuid));
    IntegrationTableMapping.SetIntegrationTableFilter(GetTableFilterFromView(Database::"TKA CRM Product Category", CRMProductCategory.Tabl
    IntegrationTableMapping.Modify();

RMProductCategory.FieldNo(tka_Code),
, true, false      You, 5 months ago • Work Orders

Category"), CRMProductCategory.FieldNo(tka_ParentCategoryId),
, true, false

ion), CRMProductCategory.FieldNo(tka_Description),
, true, false
    );

    OnResetItemCategoryMappingOnAfterInsertFieldsMapping(IntegrationTableMappingName);

    CRMSetupDefaults.RecreateJobQueueEntryFromIntTableMapping(IntegrationTableMapping, 30, EnqueueJobQueEntry, 30);
end;
```



```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"CRM Setup Defaults", '
0 references | 0% Coverage
local procedure OnAfterResetConfigurationCRM(CRMConnectionSetup: Record
begin
    ResetItemCategoryMapping(CRMCodesMgt.GetCRMProductCategoryMappingCod
end;
```



```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"CDS Setup Defaults", 'OnAfterResetCustomerAccountMapping', '', false, false)]
0 references | 0% Coverage
local procedure OnAfterResetCustomerAccountMappingCSDSetupDefaults(IntegrationTableMappingName: Code[20])
var
    IntegrationTableMapping: Record "Integration Table Mapping";
begin
    IntegrationTableMapping.Get(IntegrationTableMappingName);
    if IntegrationTableMapping."Dependency Filter" <> '' then
        IntegrationTableMapping."Dependency Filter" += '|';
    IntegrationTableMapping."Dependency Filter" += CRMCodesMgt.GetCRMProductCategoryMappingCode();
    IntegrationTableMapping.Modify();

    ...
end;
```

Customizations

## JQ Update jobs

## Update Jobs

Codeunit 5350, CRM Statistics Job

### Two responsibilities

**Update Customer Statistics**

- For newly synchronized customers
  - CRM Integration Records with Skipped = false, Statistics Uploaded = false
- For already updated customers who
  - Have Sales Lines/Service Lines modified since CRMSynchStatus."Cust. Statistics Synch. Time"
  - Have Customer Ledger Entries modified since CRMSynchStatus."Cust. Statistics Synch. Time"

**Update Status of Invoices**

- For detailed customer ledger entries  with entry no. > CRMSynchStatus."Last Update Invoice Entry No."

Status (State/Status)

- Sales Invoice Header.Canceled -> Canceled/Canceled
- Customer Ledger Entry.Remaining Amount = 0 -> Paid/Complete
- Customer Ledger Entry.Remaining Amount <> Customer Ledger Entry.Amount  -> Paid/Partial
- Else -> Active/Billed

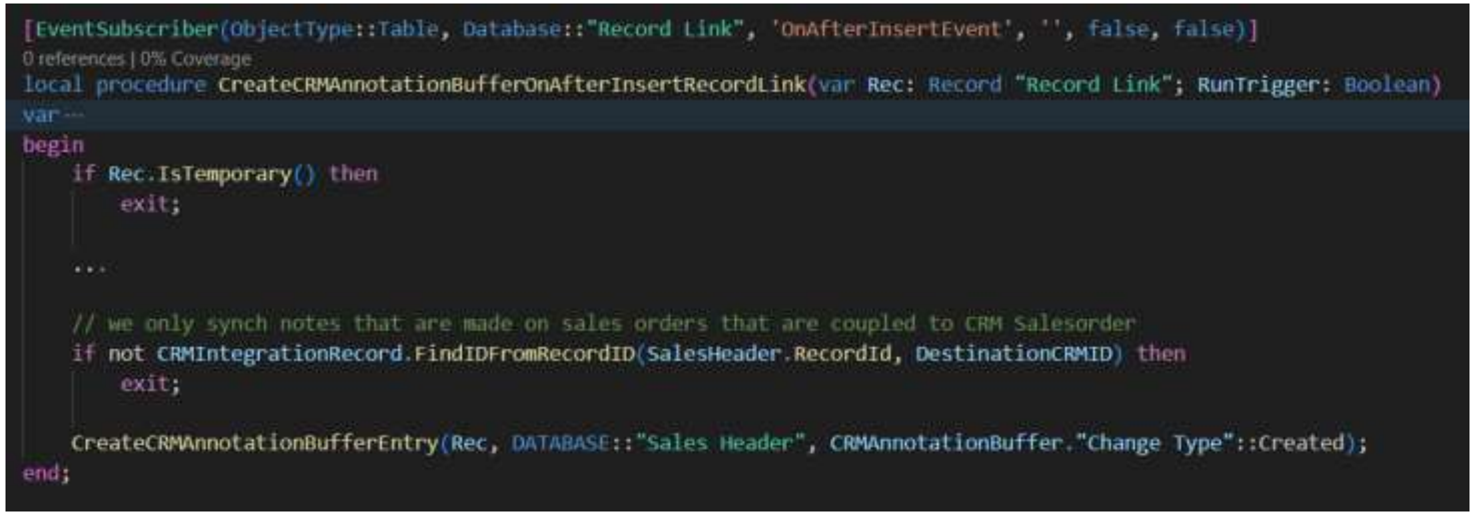
## Update Jobs

Codeunit 5355, CRM Notes Synch Job

### One responsibility

**Synchronize BC "Record Links" with CE "annotation" (Note)**

- From Record Links to Annotations
  - When the record link is created in BC, it's stored in "CRM Annotation Buffer".
  - Once created as annotation in CRM, the buffer record is removed.
- From Annotations to Record Links
  - For annotations with CreatedOn > max(CRMAnnotationCoupling."CRM Created On")
  - For annotations with LastModifiedOn > max(CRMAnnotationCoupling."CRM Modified On")



```
[EventSubscriber(ObjectType::Table, Database::"Record Link", 'OnAfterInsertEvent', '', false, false)]
0 references | 0% Coverage
local procedure CreateCRMAnnotationBufferOnAfterInsertRecordLink(var Rec: Record "Record Link"; RunTrigger: Boolean)
var···
begin
    if Rec.IsTemporary() then
        exit;

    ...

    // we only synch notes that are made on sales orders that are coupled to CRM Salesorder
    if not CRMIntegrationRecord.FindIDFromRecordID(SalesHeader.RecordId, DestinationCRMID) then
        exit;

    CreateCRMAnnotationBufferEntry(Rec, DATABASE::"Sales Header", CRMAnnotationBuffer."Change Type"::Created);
end;
```

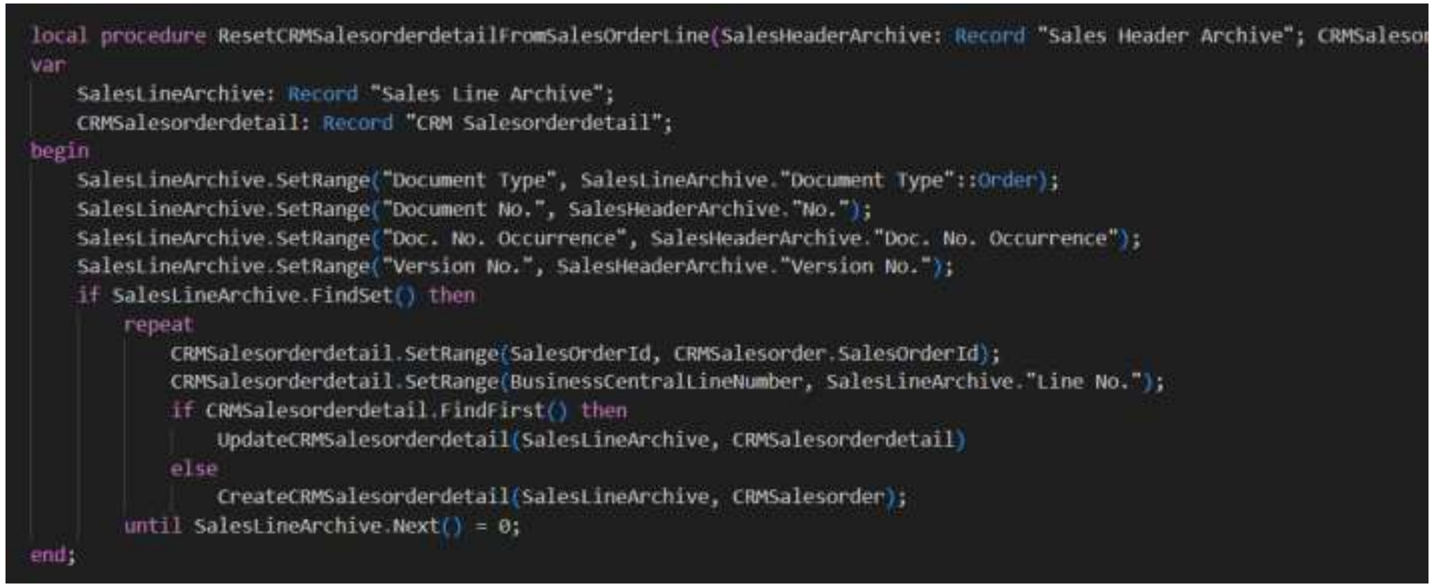
## Update Jobs

Codeunit 5366, CRM Archived Sales Orders Job

### One responsibility

**Update CRM Sales Order and Sales Order Detail when the order in BC is archived**

- CRM Integration Record
  - "Archived Sales Order" = true
  - "Archived Sales Order Updated" = false
- Updates CRM Sales Order and Order details with information from archived sales order
- Set the state and status of the CRM Order to Invoiced/Invoiced if the SalesHeaderArchive.Invoice is true.



```
local procedure ResetCRMSalesorderdetailFromSalesOrderLine(SalesHeaderArchive: Record "Sales Header Archive"; CRMSalesor
var
    SalesLineArchive: Record "Sales Line Archive";
    CRMSalesorderdetail: Record "CRM Salesorderdetail";
begin
    SalesLineArchive.SetRange("Document Type", SalesLineArchive."Document Type"::Order);
    SalesLineArchive.SetRange("Document No.", SalesHeaderArchive."No.");
    SalesLineArchive.SetRange("Doc. No. Occurrence", SalesHeaderArchive."Doc. No. Occurrence");
    SalesLineArchive.SetRange("Version No.", SalesHeaderArchive."Version No.");
    if SalesLineArchive.FindSet() then
        repeat
            CRMSalesorderdetail.SetRange(SalesOrderId, CRMSalesorder.SalesOrderId);
            CRMSalesorderdetail.SetRange(BusinessCentralLineNumber, SalesLineArchive."Line No.");
            if CRMSalesorderdetail.FindFirst() then
                UpdateCRMSalesorderdetail(SalesLineArchive, CRMSalesorderdetail)
            else
                CreateCRMSalesorderdetail(SalesLineArchive, CRMSalesorder);
        until SalesLineArchive.Next() = 0;
end;
```

directions lyon emea 2023

## Events

## Events

Basic integration events

[Customizing an Integration with Microsoft Dataverse - Business Central | Microsoft Learn]

**Codeunit 5345 "Integration Rec. Synch. Invoke"**

- OnBeforeTransferRecordFields / OnAfterTransferRecordFields

Init record before other fields are transferred
- Document Type, Line No.
- Change the record Status etc.

Update records after all fields are transferred
- Update additional fields, such as UoM
- Run validation for additional fields (payment terms)

# Events

Basic integration events

[Customizing an Integration with Microsoft Dataverse - Business Central | Microsoft Learn]

**Codeunit 5345 "Integration Rec. Synch. Invoke"**

- OnBeforeTransferRecordFields / OnAfterTransferRecordFields
- OnBeforeInsertRecord / OnAfterInsertRecord

Init record before the record is inserted

- CompanyId, OwnerId
- Complex logic that must happen once for every rec

Update records after all fields are transferred

- Change the record Status etc.
- Run the synchronization for related records (lines etc.)

# Events

## Basic integration events

[Customizing an Integration with Microsoft Dataverse - Business Central | Microsoft Learn]

### Codeunit 5345 "Integration Rec. Synch. Invoke"

- OnBeforeTransferRecordFields / OnAfterTransferRecordFields
- OnBeforeInsertRecord / OnAfterInsertRecord
- OnBeforeModifyRecord / OnAfterModifyRecord

Init record before the record is inserted

- CompanyId
- Complex logic that must happen once for every rec with every modification

Update records after all fields are transferred

- Change the record Status etc.
- Run the synchronization for related records (lines etc.)

## Events

Basic integration events

Customizing an Integration with Microsoft Dataverse - Business Central | Microsoft Learn

**Codeunit 5345 "Integration Rec. Synch. Invoke"**

- OnBeforeTransferRecordFields / OnAfterTransferRecordFields
- OnBeforeInsertRecord / OnAfterInsertRecord
- OnBeforeModifyRecord / OnAfterModifyRecord

**Codeunit 5336 "Integration Record Synch."**

- OnTransferFieldData

**Event we use the most often**

Implement custom data transfer logic.

- Hard-mapped enums (Customer/Vendor Blocked Status, …)
- Complex mapping or lookups (Bill-to Customer; Country / Region, …)

# Events

Integration events – not described on MS Learn

**Codeunit 5345 "Integration Rec. Synch. Invoke"**

- OnWasModifiedAfterLastSynch
- OnAfterUnchangedRecordHandled

Is Record Changed?
- Custom condition for the change.
- Check other entities.

What should happen when the record is not changed?
- Check the child entities (when the header is not changed, lines could still have changes)

## Events

Integration events – not described on MS Learn

### Codeunit 5345 "Integration Rec. Synch. Invoke"

- OnWasModifiedAfterLastSynch
- OnAfterUnchangedRecordHandled

### Codeunit 5340 "CRM Integration Table Synch."

- OnQueryPostFilterIgnoreRecord
- OnLoadCRMOption

Should the record be ignored as it was already processed?

- Archived (=posted) documents.

Load CRM Option

- Init the temp table used for Enum/Option mapping

## Events

Integration events – not described on MS Learn

### Codeunit 5345 "Integration Rec. Synch. Invoke"

- OnWasModifiedAfterLastSynch
- OnAfterUnchangedRecordHandled

### Codeunit 5340 "CRM Integration Table Synch."

- OnQueryPostFilterIgnoreRecord
- OnLoadCRMOption

### Codeunit 5338 "Integration Record Management"

- OnIsIntegrationRecordSkipped

Should the record be skipped?

- Similar usage to OnQueryPostFilterIgnoreRecord
- Ignored records are not visible (that the record was ignored)
- Skipped records are marked as skipped (synch log)

## Events

Integration events – not described on MS Learn

**Codeunit 5345 "Integration Rec. Synch. Invoke"**

- OnWasModifiedAfterLastSynch
- OnAfterUnchangedRecordHandled

**Codeunit 5340 "CRM Integration Table Synch."**

- OnQueryPostFilterIgnoreRecord
- OnLoadCRMOption

**Codeunit 5338 "Integration Record Management"**

- OnIsIntegrationRecordSkipped

**codeunit 5330 "CRM Integration Management"**

- OnIsCRMIntegrationRecord
- OnGetTableIdFromCRMOption

OnIsCRMIntegrationRecord
- Tables listed in "Integration Table Mapping" are marked as integration record automatically.
- Use for tables not directly linked (f.e. archived documents)

CRM Option to Table
- Specifies table for CRM option mapping

## Events

Integration events – not described on MS Learn (continue)

**Codeunit 5332 "Lookup CRM Tables"**

- OnLookupCRMOption
- OnLookupCRMTables

Lookup implementation for CRM tables and options.

## Events

Integration events – not described on MS Learn (continue)

**Codeunit 5332 "Lookup CRM Tables"**

- OnLookupCRMOption
- OnLookupCRMTables

**Codeunit 5343 "CRM Sales Order to Sales Order"**

- OnBeforeGetCRMAccountOfCRMSalesOrder

Different way how the customer in sales order should be found?

Use this event.

## Events

Integration events – not described on MS Learn (continue)

**Codeunit 5332 "Lookup CRM Tables"**

- OnLookupCRMOption
- OnLookupCRMTables

**Codeunit 5343 "CRM Sales Order to Sales Order"**

- OnBeforeGetCRMAccountOfCRMSalesOrder

**Codeunit 5341 "CRM Int. Table. Subscriber"**

- OnFindNewValueForCoupledRecordPK

Allows to change how the related record is found.

- Use when the related record can't be found using the OOTB logic using the first key field.
- Usage
    - Item Variants
    - Bill-to Customer (lookup standard customer if empty)

directions lyon emea 2023

## Events

Integration events – not described on MS Learn (continue)

**Codeunit 5332 "Lookup CRM Tables"**

- OnLookupCRMOption
- OnLookupCRMTables

**Codeunit 5343 "CRM Sales Order to Sales Order"**

- OnBeforeGetCRMAccountOfCRMSalesOrder

**Codeunit 5341 "CRM Int. Table. Subscriber"**

- OnFindNewValueForCoupledRecordPK

**Codeunit 5357 "Int. Rec. Uncouple Invoke"**

- OnAfterUncoupleRecord

Use to Uncouple other tables

- When Sales Order is uncoupled, uncouple the sales order lines

# Events

Integration events – not described on MS Learn (last one, FINALLY)

**Codeunit 5342 "CRM Synch. Helper"**

- OnConvertOptionToTableOnBeforeSetRangeForIntegrationFieldID

CRM Option to Table
- Specifies CRM option mapping field

# Events

Integration events – not described on MS Learn (continue 2)

**Codeunit 5342 "CRM Synch. Helper"**

- OnConvertOptionToTableOnBeforeSetRangeForIntegrationFieldID

- OnBeforeCalculateActualStatusCode
- OnUpdateCRMInvoiceStatusFromEntryOnBeforeCheckFieldsChanged
- OnUpdateCRMInvoiceStatusFromEntryOnBeforeModify
- OnCancelCRMInvoiceOnBeforeCheckFieldsChanged
- OnCancelCRMInvoiceOnBeforeModifyCRMInvoice

(new in 22.4+5)

Use for custom logic for CRM invoice Status/State and related fields

- Custom CRM Invoice State/Status conditions
- Remaining Amount for CRM invoice

slido

## Have you ever used any of these events?

ⓘ Start presenting to display the poll results on this slide.

directions lyon
emea 2023

## Limitations

## Limitations for customizations

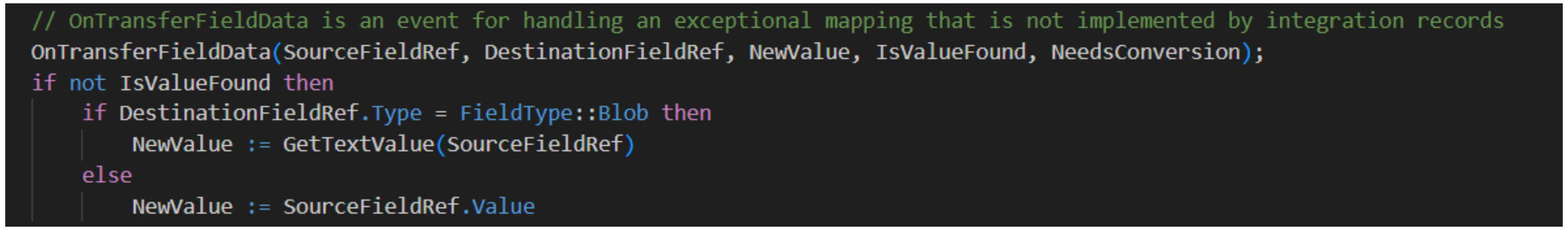
BLOB to Text

The Blob to text transformation (regardless of the direction) is not currently supported.

**REASON**

Never been tested before.

**Will be updated in upcoming versions**



```
// OnTransferFieldData is an event for handling an exceptional mapping that is not implemented by integration records
OnTransferFieldData(SourceFieldRef, DestinationFieldRef, NewValue, IsValueFound, NeedsConversion);
if not IsValueFound then
    if DestinationFieldRef.Type = FieldType::Blob then
        NewValue := GetTextValue(SourceFieldRef)
    else
        NewValue := SourceFieldRef.Value
```

directions lyon emea 2023

## Limitations for customizations

### Integrate two CRM tables with one BC table

Some actions/processes (actions available on BC record's pages, …) are not working if one BC table is integrated with multiple CRM entities.

**REASON**

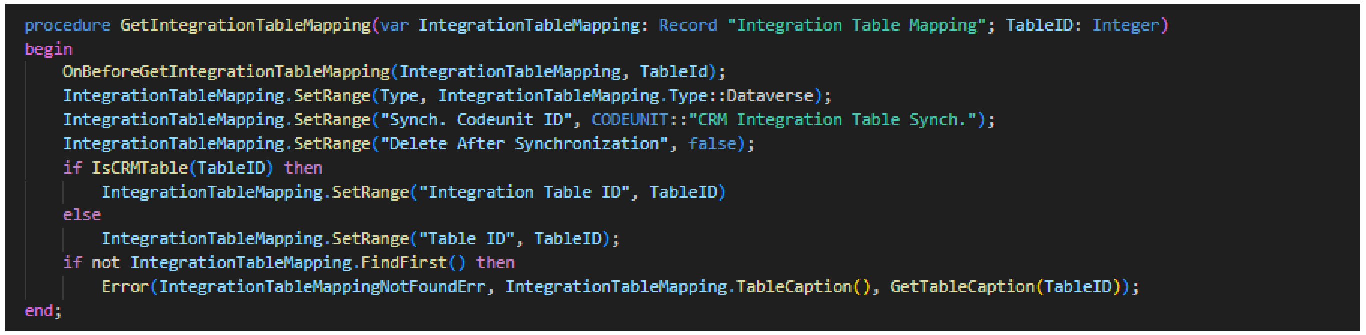
The publisher OnBeforeGetIntegrationTableMapping() provides only source (BC) table number.

**CHANGE**

New procedure (and publisher) with RecordRef instead of table number. With RecordRef, customizations can choose proper CRM table based on the values in the record.

**Will be updated in upcoming versions**

**Open Source Contribution!**



```
procedure GetIntegrationTableMapping(var IntegrationTableMapping: Record "Integration Table Mapping"; TableID: Integer)
begin
    OnBeforeGetIntegrationTableMapping(IntegrationTableMapping, TableId);
    IntegrationTableMapping.SetRange(Type, IntegrationTableMapping.Type::Dataverse);
    IntegrationTableMapping.SetRange("Synch. Codeunit ID", CODEUNIT::"CRM Integration Table Synch.");
    IntegrationTableMapping.SetRange("Delete After Synchronization", false);
    if IsCRMTable(TableID) then
        IntegrationTableMapping.SetRange("Integration Table ID", TableID)
    else
        IntegrationTableMapping.SetRange("Table ID", TableID);
    if not IntegrationTableMapping.FindFirst() then
        Error(IntegrationTableMappingNotFoundErr, IntegrationTableMapping.TableCaption(), GetTableCaption(TableID));
end;
```

MSDyn365BC App Platform Contribution program, Issue #475

## Limitations for customizations

### Coupled to Dataverse for standard tables

“Coupled to Dataverse” for OOTB tables added by customization is not fully supported

**REASON**

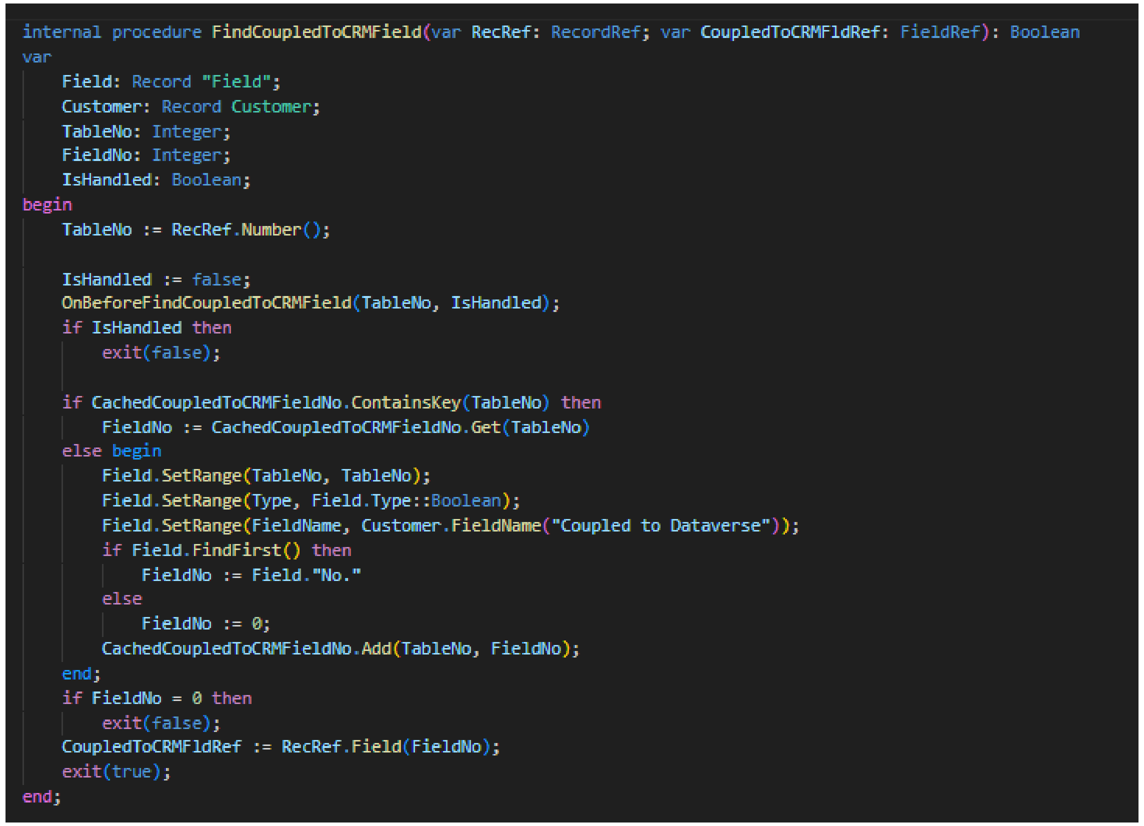
The system search for field with exact name “**Coupled to Dataverse**”. Custom fields have usually affix.

**CHANGE**

If field is not found, new logic with SetFilter(*%1*) will be called to search for the field with an affix.

**Will be updated in upcoming versions**

**Open Source Contribution!**



```
internal procedure FindCoupledToCRMField(var RecRef: RecordRef; var CoupledToCRMFldRef: FieldRef): Boolean
var
    Field: Record "Field";
    Customer: Record Customer;
    TableNo: Integer;
    FieldNo: Integer;
    IsHandled: Boolean;
begin
    TableNo := RecRef.Number();

    IsHandled := false;
    OnBeforeFindCoupledToCRMField(TableNo, IsHandled);
    if IsHandled then
        exit(false);

    if CachedCoupledToCRMFieldNo.ContainsKey(TableNo) then
        FieldNo := CachedCoupledToCRMFieldNo.Get(TableNo)
    else begin
        Field.SetRange(TableNo, TableNo);
        Field.SetRange(Type, Field.Type::Boolean);
        Field.SetRange(FieldName, Customer.FieldName("Coupled to Dataverse"));
        if Field.FindFirst() then
            FieldNo := Field."No."
        else
            FieldNo := 0;
        CachedCoupledToCRMFieldNo.Add(TableNo, FieldNo);
    end;
    if FieldNo = 0 then
        exit(false);
    CoupledToCRMFldRef := RecRef.Field(FieldNo);
    exit(true);
end;
```

MSDyn365BC App Platform Contribution program, Issue #475

slido

## Do you know about any other limitation?

ⓘ Start presenting to display the poll results on this slide.

Future

# What is planned

## What is planned

Generic Table/Field mapping

“Coupled to Dataverse” for OOTB tables added by customization is not fully supported

**Current state**

Any new field mapping must be done by a developer, even when the field exists in the BC proxy table.

**Future state**

The fields without any logic that has matching data type can be added by a user (if the field exists in the BC proxy table).

If the table does not exist, developer is needed only to add the field to proxy table (can be done with altpgen).

**Open Source Contribution!**

MSDyn365BC App Platform Contribution program
Issue #400

## What is planned

Field Service integration

**MICROSOFT ROUNDTABLE: FIELD SERVICE INTEGRATION SCENARIOS**

Date: 02-11-2023 | From: 11:15 to 12:00 | Room: Gratte Ciel 3

Well, you've missed it…

[Directions EMEA 2023 - Microsoft Roundtable: Field Service integration scenarios (directions4partners.com)]

slido

## Audience Q&A Session

ⓘ Start presenting to display the audience questions on this slide.

directions lyon emea 2023

## Give us Feedback!

and help us improve!

Please rate my session in the Conference App!

and leave a constructive comment.

Win a 100€ gift card

one ticket for every session feedback!

**Thank you !**