

Contoso Demo Tool workshop

Haoran Sun
Wael AbuSeada



Who are we?



Haoran Sun

**Software Engineer
@Microsoft**



Predrag Maricic

**Principal engineering
manager @Microsoft**



Wael AbuSeada

**"Legendary" Software
Engineer @ Microsoft**

Agenda

- Introduce Contoso Coffee
 - Intro/Modules
- How to build your module
- How to switch an existing package to module.
- Setup dev and access to prod env
- Work time ~25 min
 - Your own module
 - Break up Evaluation module
 - Assembly
- Q/A

How to build your own module

Take dependency

app.json

```
"dependencies": [  
  {  
    "id": "5a0b41e9-7a42-4123-d521-2265186cfb31",  
    "name": "Contoso Coffee Demo Dataset",  
    "publisher": "Microsoft",  
    "version": "23.0.0.0"  
  },  
]
```

Extend the Enum

```
enumextension 50100 "My Module" extends "Contoso Demo Data Module"
```

```
{  
    value(50100; "My Module")  
    {  
        Implementation = "Contoso Demo Data Module" = "My Module";  
    }  
}
```


Implement the Interface

Interface "Contoso Demo Data Module"

```
{  
    procedure RunConfigurationPage();  
  
    procedure GetDependencies() Dependencies: List of [Enum "Contoso Demo Data Module"];  
  
    procedure CreateSetupData();  
    procedure CreateMasterData();  
    procedure CreateTransactionalData();  
    procedure CreateHistoricalData();  
}
```

RunConfigurationPage

codeunit 5140 "Warehouse Module" implements "Contoso Demo Data Module"

```
{  
    .....  
  
    procedure RunConfigurationPage()  
    begin  
        Page.Run(Page::"Warehouse Module Setup");  
    end;  
  
    .....  
}
```




Name

Common Module

Manufacturing Module

Warehouse Module

Service Module



✓ Saved



Warehouse Module Setup

Locations >

Master Data

Customer No.



Item 2 No.



Vendor No.



Item 3 No.



Item 1 No.



Configuration Page

Add Dependency

codeunit 5140 "Warehouse Module" implements "Contoso Demo Data Module"

```
{  
    .....  
  
    procedure GetDependencies() Dependencies: List of [Enum "Contoso Demo Data Module"]  
    begin  
        Dependencies.Add(Enum::"Contoso Demo Data Module"::"Common Module");  
    end;  
  
    .....  
}
```

Data Levels

codeunit 5140 "Warehouse Module" implements "Contoso Demo Data Module"

```
procedure CreateSetupData()
begin
    WarehouseModuleSetup.InitRecord();
    Codeunit.Run(Codeunit::"Create Whse Put Away Template");
    Codeunit.Run(Codeunit::"Create Whse Location");
    Codeunit.Run(Codeunit::"Create Whse No Series");
    Codeunit.Run(Codeunit::"Create Whse Posting Setup");
    Codeunit.Run(Codeunit::"Create Whse Inventory Setup");
    Codeunit.Run(Codeunit::"Create Whse Item Category");
end;

procedure CreateMasterData()
begin
end;
















procedure CreateTransactionalData()
begin
end;

procedure CreateHistoricalData()
begin
end;
```

Helper Codeunits

Contoso Helpers

Contents History

Name ↑
 ContosoCountryOrRegion.Codeunit.al
 ContosoCustomerVendor.Codeunit.al
 ContosoGLAccount.Codeunit.al
 ContosoGLAccount.Table.al
 ContosoItem.Codeunit.al
 ContosoManufacturing.Codeunit.al
 ContosoNoSeries.Codeunit.al
 ContosoPostingGroup.Codeunit.al
 ContosoPostingSetup.Codeunit.al
 ContosoPurchase.Codeunit.al
 ContosoResource.Codeunit.al
 ContosoSales.Codeunit.al
 ContosoService.Codeunit.al
 ContosoUnitOfMeasure.Codeunit.al
 ContosoUtilities.Codeunit.al
ContosoWarehouse.Codeunit.al

Helper Codeunits

codeunit 5129 "Contoso Unit of Measure"

```
procedure InsertUnitOfMeasure(UnitOfMeasureCode: Code[10]; Description: Text[10]; InternationalStandardCode: Code[10])
var
    UnitOfMeasure: Record "Unit of Measure";
    Exists: Boolean;
begin
    if UnitOfMeasure.Get(UnitOfMeasureCode) then begin
        Exists := true;

        if not OverwriteData then
            exit;
    end;

    UnitOfMeasure.Validate(Code, UnitOfMeasureCode);
    UnitOfMeasure.Validate(Description, Description);
    UnitOfMeasure.Validate("International Standard Code", InternationalStandardCode);

    if Exists then
        UnitOfMeasure.Modify(true)
    else
        UnitOfMeasure.Insert(true);
end;
```

Localization

codeunit 10856 "Common Module FR"

```
[EventSubscriber(ObjectType::Codeunit, Codeunit::"Contoso Demo Tool", 'OnAfterGeneratingDemoData', '', false, false)]
local procedure VATPostingSetup(Module: Enum "Contoso Demo Data Module"; ContosoDemoDataLevel: Enum "Contoso Demo Data Level")
var
    LocalStandardVATPercentage: Decimal;
begin
    if Module = Enum::"Contoso Demo Data Module"::"Common Module" then
        if ContosoDemoDataLevel = Enum::"Contoso Demo Data Level"::"Setup Data" then begin
            LocalStandardVATPercentage := 20;
            ContosoPostingSetup.SetOverwriteData(true);
            ContosoPostingSetup.InsertVATPostingSetup(CommonPostingGroup.Domestic(), ..., LocalStandardVATPercentage, ...
            ContosoPostingSetup.SetOverwriteData(false);
        end;
    end;
```

From the lab: Convert packages to code

Get started with setup dev environment

Setup Dev environment

You will need:

1. Laptop
2. VSCode
3. Updated AL compiler in VSCode

Setup Dev environment

- Login to admin center using:
 - Admin center:
<https://businesscentral.dynamics.com/9dad436-6f60-4172-a9d2-43c91c24ecb1/admin>



Setup Dev environment

- Assign environment to yourself by renaming it.
- ZCust4_20231031_134052

Database ▾Support ▾Update Settings ▾CopyRestoreRenameDeleteRefresh

roduction

Application Family
Business Central

Country/region
US

Azure Region
East US 2

Application Insights Connection String [\(?\)](#)
Not Set ([Define](#))

Security Group
Not Set ([Define](#))

Access with Microsoft
Off ([Modify](#))

[ynamics.com/9dad436-6f60-4172-a9d2-43c91c24ecb1/Production](#)

Platform Version
23.0

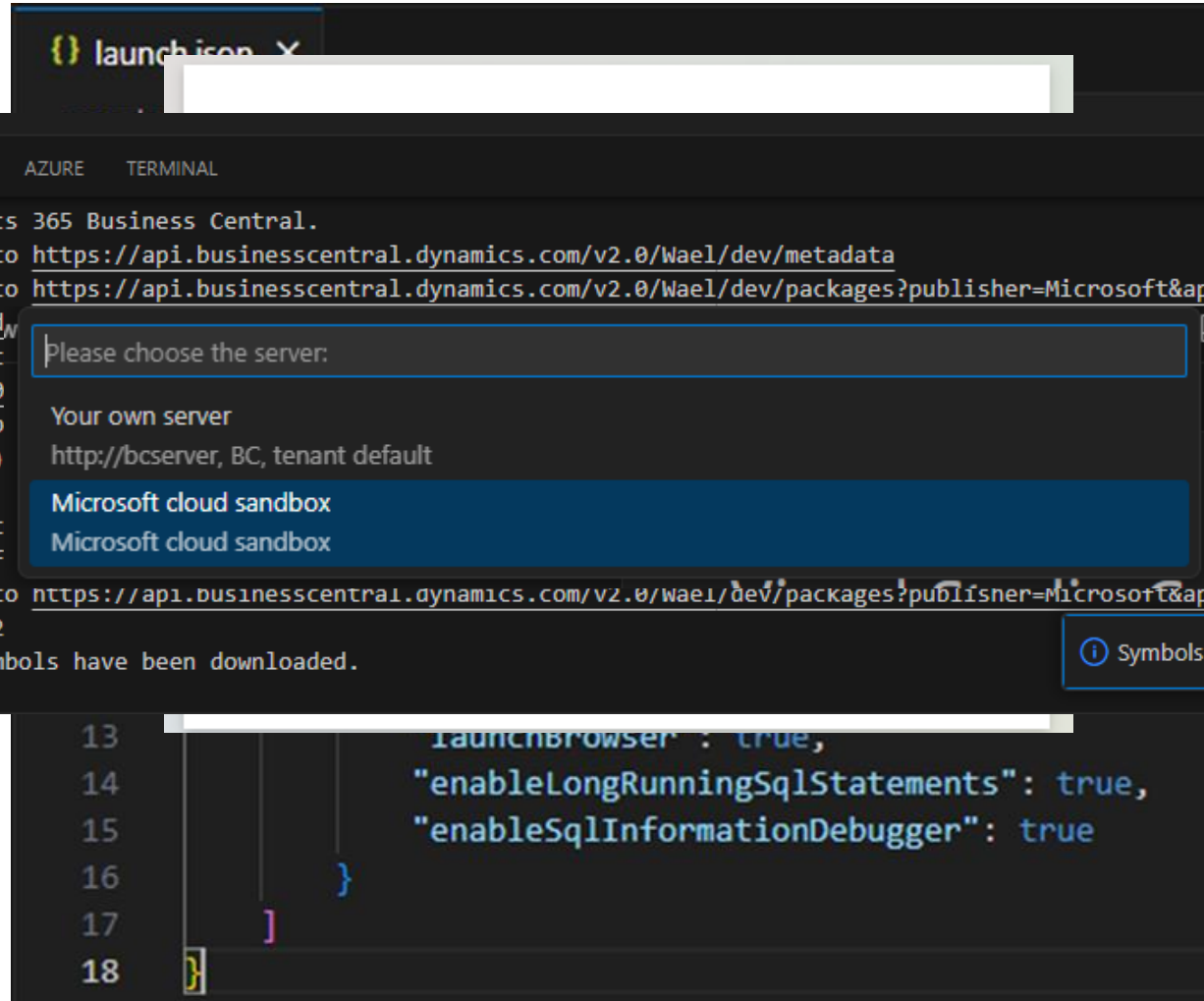
Available Update Version

Update Scheduling A

Update Window (UTC-06:00) [\(?\)](#)
09:00 PM - 04:00 AM ([Modify](#))

Update Rollout State [\(?\)](#)

Debug sandbox



The screenshot shows the Visual Studio Code interface. The top bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, PORTS, AZURE, and TERMINAL. The OUTPUT tab is active, displaying a log of system messages. A modal dialog is open in the center, asking the user to choose a server. The log messages show the process of targeting Dynamics 365 Business Central, sending requests to the API, and downloading reference symbols. Below the log, a code editor shows a configuration file with settings for launching the browser and enabling SQL debugging.

```
[2023-10-20 13:13:07.48] Targeting Dynamics 365 Business Central.  
[2023-10-20 13:13:07.50] Sending request to https://api.businesscentral.dynamics.com/v2.0/Wael/dev/metadata  
[2023-10-20 13:13:09.00] Sending request to https://api.businesscentral.dynamics.com/v2.0/Wael/dev/packages?publisher=Microsoft&appName=System&versionText=1.0.0.0&  
appId=8874ed3a-0643-4247-9ced-7a7002f7135d  
[2023-10-20 13:13:09.00] Sending request to https://api.businesscentral.dynamics.com/v2.0/Wael/dev/packages?publisher=Microsoft&appName=Application&versionText=23.0.0.0&  
appId=00000000-0000-0000-0000-000000000000  
[2023-10-20 13:13:11.03] The following dep  
System Application by Microsoft (23.0.0.0)  
Base Application by Microsoft (23.0.0.0)  
[2023-10-20 13:13:11.03] Sending request to https://api.businesscentral.dynamics.com/v2.0/Wael/dev/packages?publisher=Microsoft&appName=Base Application&versionText=23.0.0.0&  
appId=63ca2fa4-4f03-4f2b-a480-172fef340d3f  
[2023-10-20 13:13:11.03] Sending request to https://api.businesscentral.dynamics.com/v2.0/Wael/dev/packages?publisher=Microsoft&appName=Base Application&versionText=23.0.0.0&  
appId=437dbf0e-84ff-417a-965d-ed2bb9650972  
[2023-10-20 13:13:29.48] All reference symbols have been downloaded.
```

Please choose the server:

- Your own server
http://bcserver, BC, tenant default
- Microsoft cloud sandbox**
- Microsoft cloud sandbox

Symbols have been downloaded.

```
13 launchBrowser : true,  
14     "enableLongRunningSqlStatements": true,  
15     "enableSqlInformationDebugger": true  
16 }  
17 ]  
18 }
```

Workshop example

Modules to work on: Assembly

- Use tool as starting point:
 - Create Conf Package ASSEM, add following tables with fields and filters (to limit amount of exported data)
 - Optional: Export to excel to validate that all good.
 - Run tool to get 5 files. Make sure files can compile

Table	Fields	Filters
27 Item	1 No., 3 Description, 8 Base UoM, 10 Type, 18 Unit Price, 22 Unit Cost, 5419 Replenishment System	No. = 1925-W 1929-W 1920-S 1968-S
90 BOM Component	1 Parent Item No., 2 Line No., 3 Type, 4 No., 8 Quantity Per	Parent Item No. = 1925-W 1929-W; Type = Item
308 No. Series	All fields will be fine	Code = A-*
309 No Series Line	All fields will be fine	Series Code = A-*
905 Assembly Setup	All fields will be fine	

- Folder structure concept:
 - Move No Series, No Series Lines and Assembly setup codeunits to Setup folder
 - Move Items and BOM Components to Master data
- Patterns:
 - Align Setup and No series with similar approach for Service management or warehouse (merge files?)
 - Align Item and Bom components with other modules (use helper to create items), benefit from posting setup, price adjustment.
 - Plus (open contribution PR): Introduce helper to create BOM Components
- Module definition
 - ?? Interface implementation, module registration (not sure what we need to do, but result – user can see module in UI and run it)
 - For each testing, recommendation is to create copy of My Company (MyTest1, MyTest2,...)

- Setup and open balances:

- Create setup table/page with Main Location field, make sure it is initialized with MAIN (check concept in manufacturing, service)
- Create item journal batch/template/lines for components (1920-S|1968-S). Use Location from Setup (main), qty hardcoded 100. See examples in Service and Manufacturing modules.

- Documents:

- Introduce helper to create Assembly Order and Assembly Order lines. (Transaction folder?)
- Create assemble order for item 1925-W, qty 3, (location from Setup = main), see helpers for sales orders.

Working time

Give us Feedback!

and help us improve!

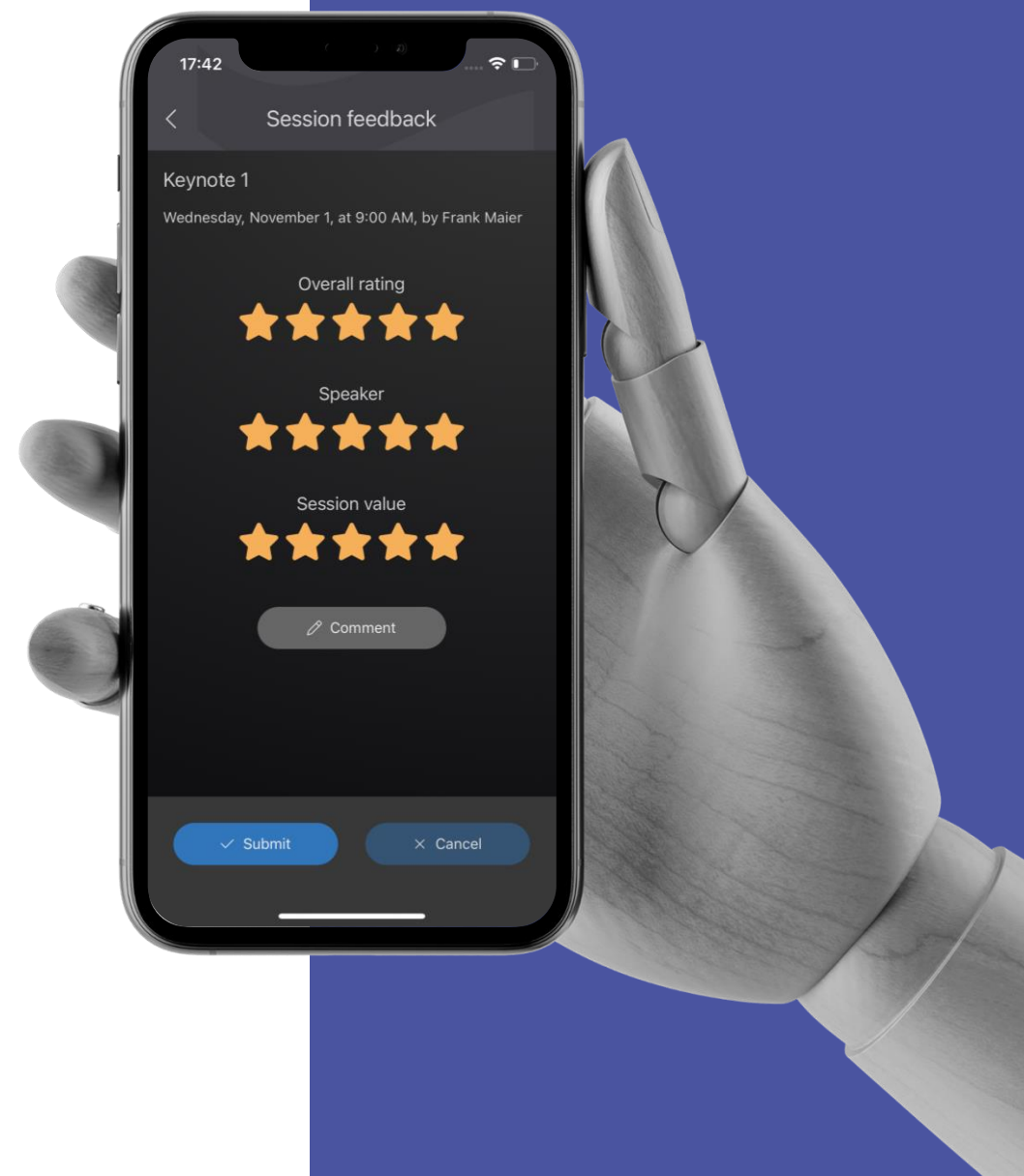
Please rate my session in the Conference App!

and leave a constructive comment.

Win a 100€ gift card

one ticket for every session feedback!

Thank you !



MERCI

TACK

ขอบคุณครับ

KIITOS

MULȚUMESC

DANKE

DANK U

GRACIAS

감사합니다

شكراً

DZIĘKUJĘ

OBRIGADO

THANK YOU

TEŞEKKÜRLER

ευχαριστώ

धन्यवाद

AČIŪ

תודה

Merci beaucoup!

நன்றி

多謝晒

PALDIES

ДЯКУЮ

ĎAKUJEM

متشكراً

TERIMA KASIH

TAK

HVALA

DANKE

DĚKUJI

谢谢

CÁM ƠN

DANKON

GRAZIE

KÖSZÖNÖM

БЛАГОДАРЯ

ありがとうございます