Microsoft Dynamics 365

Introducing Contoso Demo tool

Agenda

- Current demo data/tool

- Introducing Contoso

- How to build demo data module?

- What's next?

Current demo data/tool

Available demo data

- Setup
- Evaluation (SaaS)
- Extended (On-premises)

Create New Company

Specify some basic information

Enter a name for the company.

[   ]

Select the data and setup to get started.

| Production - Setup Data Only ⌄ |

Evaluation - Sample Data
Production - Setup Data Only
Advanced Evaluation - Complete Sample Data
Create New - No Data

Create a company with the Essential functionality scope containing data and setup, such as a chart of accounts and payment methods ready for use by companies with standard processes. Set up your own items and customers, and start posting right away.

Back   Next   Finish

Extracting subset of demo data

Evaluation package	Export	Import	Export final package
· Identify which tables contain demo data for the feature you want to present	· Export related tables as a rapidstart package	· Import the package to another company.	· Export final package containing different subset of demo data

Introducing Contoso

What is Contoso?

Contoso Coffee is a fictitious company that produces consumer and commercial coffee makers.

Maintained by Microsoft to demonstrate how Microsoft Dynamics 365 products and services can support prospective customers

Contoso Modules

Available

- Common
- Warehousing
- Manufacturing
- Service

Future

· We are working on releasing more modules

Designing Contoso

Designing Contoso

Modules

· Split demo data into modules
· Modules must be scenario-based.
· Modules should work together and separately

Dependency

· Your demo data module can have dependency on other modules(s)

Data layers

· Split data layers:
→ · Setup
→ · Master
→ · Transactional
→ · Historical

Interface

Speed up productivity

· Simplify creating a demo data module
· Tool orchestrates configuring and running the modules
· Simplify localization & Extensibility

High level architecture

The tool – By Microsoft

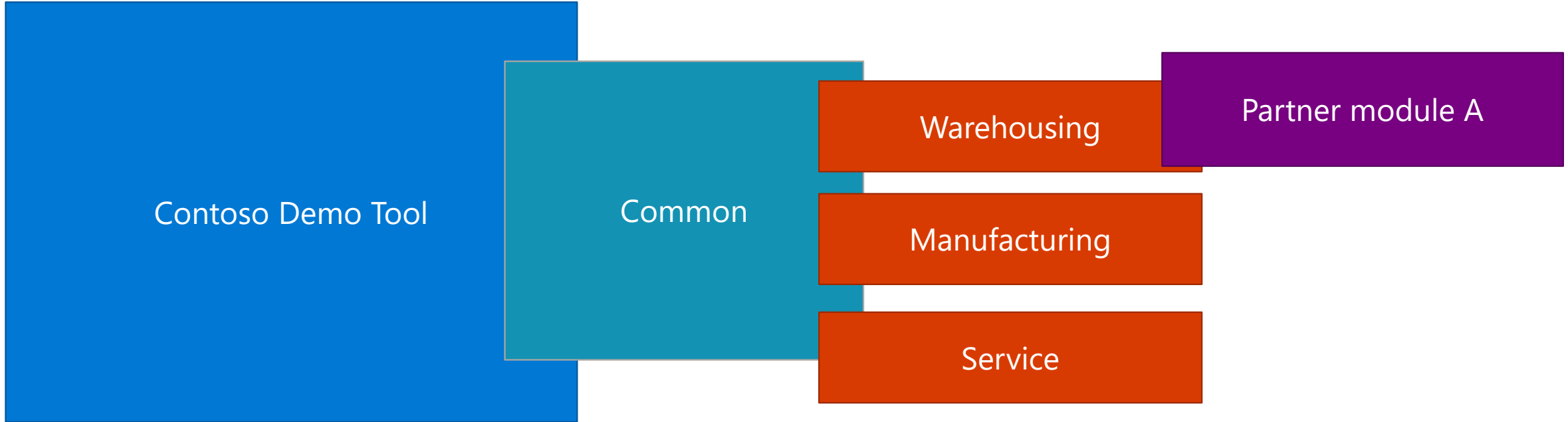The data – By Microsoft & Partners

Contoso Demo Tool

Common

Warehousing

Partner module A

Manufacturing

Service

Dependency between demo data modules

How to build demo data module?

Create your extension and uptake dependency

```json
  "id": "e0cbd2c6-b4ed-47ae-b6bb-cb007b3a6cef",
  "name": "My Contoso Demo Data Module",
  "publisher": "Microsoft Demo",
  "version": "1.0.0.0",
  "brief": "My Contoso Demo Data Module",
  "description": "My Contoso Demo Data Module",
  "privacyStatement": "https://go.microsoft.com/fwlink/?linkid=724009",
  "EULA": "https://go.microsoft.com/fwlink/?linkid=2009120",
  "help": "https://go.microsoft.com/fwlink/?linkid=2187180",
  "url": "https://go.microsoft.com/fwlink/?linkid=724011",
  "logo": "./ExtensionLogo.png",
  "dependencies": [
    {
      "id": "5a0b41e9-7a42-4123-d521-2265186cfb31",
      "name": "Contoso Coffee Demo Dataset",
      "publisher": "Microsoft",
      "version": "23.0.0.0"
    }
  ],
  "platform": "23.0.0.0",
  "application": "23.0.0.0",
  "idRanges": [
    {
      "from": 50100,
      "to": 50149
    }
  ],
  "resourceExposurePolicy": {
    "allowDebugging": true,
    "allowDownloadingSource": true,
    "includeSourceInSymbolFile": true
  },
  "features": [
    "NoImplicitWith"
```

Extend the Enum

```
1    enumextension 50100 "My Contoso Demo Data Module" extends "Contoso Demo Data Module"
2    {
     0 references
3        value(50100; "My Contoso Demo Data Module")
4        {
5            Implementation = "Contoso Demo Data Module" = "My Contoso Demo Data Module";
6        }
7    }
```

Implement the interface

```al
codeunit 50100 "My Contoso Demo Data Module" implements "Contoso Demo Data Module"
{
    0 references
    procedure RunConfigurationPage();
    begin
        Page.Run(Page::"My Demo Data Configuration");
    end;

    0 references
    procedure GetDependencies() Dependencies: List of [enum "Contoso Demo Data Module"];
    begin
        Dependencies.Add(Enum::"Contoso Demo Data Module"::"Common Module");
    end;

    0 references
    procedure CreateSetupData();
    begin
        Codeunit.Run(Codeunit::"Create My Demo No Series");
        Codeunit.Run(Codeunit::"Create My Demo GL Accounts");
        Codeunit.Run(Codeunit::"Create My Demo Posting Group");
        Codeunit.Run(Codeunit::"Create My Demo Posting Setup");
    end;

    0 references
    procedure CreateMasterData();
    begin
        Codeunit.Run(Codeunit::"Create My Demo Customers");
        Codeunit.Run(Codeunit::"Create My Demo Items");
    end;

    0 references
    procedure CreateTransactionalData();
    begin
        Codeunit.Run(Codeunit::"Create My Demo Sales Orders");
    end;

    0 references
    procedure CreateHistoricalData();
    begin
        Codeunit.Run(Codeunit::"Create My Demo Sales Invoices");
    end;
```

Utilize the helper codeunits and dependencies

```
1   codeunit 50102 "Create My Demo Posting Setup"
2   {
3
        0 references
4       trigger OnRun()
5       var
6           ContosoPostingSetup: Codeunit "Contoso Posting Setup";
7           CommonPostingGroup: Codeunit "Create Common Posting Group";
8           CommonGLAccount: Codeunit "Create Common GL Account";
9       begin
10          ContosoPostingSetup.InsertGeneralPostingSetup(CommonPostingGroup.EU(), CommonPostingGroup.Retail(), CommonGLAc
11      end;
12  }
```

Quick Recap

1. Create your extension and uptake dependency
2. Extend the Enum
3. Implement the interface

What's next?

On-demand demo data

- More Modules
- Move away from rapid start package
- Improve company creation wizard

Open source

s Public

equests **10**   💬 Discussions   ▶ Actions   📖 Wiki   🛡 Security   📈 Insights

ALAppExtensions / **Apps** / **W1** / **ContosoCoffeeDemoDataset** / **app** /   ⧉

🧑 **mazhelez** Bump version to 22.5 (**#24449**) ✓

Name	Last commi
📁 ..	
📁 Manufacturing	Update wi
📁 Permissions	Update wi
📁 Shared	Update wi
📁 Warehousing	Uptake ch
📄 ExtensionLogo.png	Update to
📄 app.json	Bump vers

Thank you