IMPLEMENTING A CUSTOM PRICING ENGINE WITHIN DYNAMICS 365 CUSTOMER ENGAGEMENT

Joe Griffin

https://crmchap.co.uk

joe@griffin.gb.net

joejgriffin

joejgriffin

JJGriffin

AGENDA

- Custom Pricing Plug-ins Overview
  - Why should they be used?
  - What can you use them for?
  - How do you get started with them?
- Let’s build!
  - The Scenario
  - Ready, Steady, Code!
- Closing Remarks

ABOUT ME

- Director @ SOLO Group:
  - SOLO Mailing Services
  - SOLO Cloud Solutions
- 4+ years experience working with Dynamics CRM 2015, 2016, 365 Customer Engagement & the Power Platform
- Also experienced in/worked with:
  - Microsoft Azure
  - Power BI
  - SQL Server (SSIS, SSRS etc.)
  - C# / Jscript
  - Azure DevOps
  - Project Management
- Currently involved as technical architect for a solution for the UK water industry, built using Power BI and Azure 👌
- CRMUG/D365UG NW Committee Member

Microsoft Certified Professional
MCSA
BI Reporting
Earned 2018
Microsoft Certified Professional
MCSE
Data Management and Analytics
Earned 2019
Microsoft Certified Professional
MCSE
Business Applications
Earned 2018
Microsoft Certified Professional
MCSA
Dynamics 365
Earned 2017
Microsoft Certified
Dynamics 365 Sales Functional Consultant
Associate
PRINCE2
Practitioner CPD
Scrum Alliance
CSM Certified

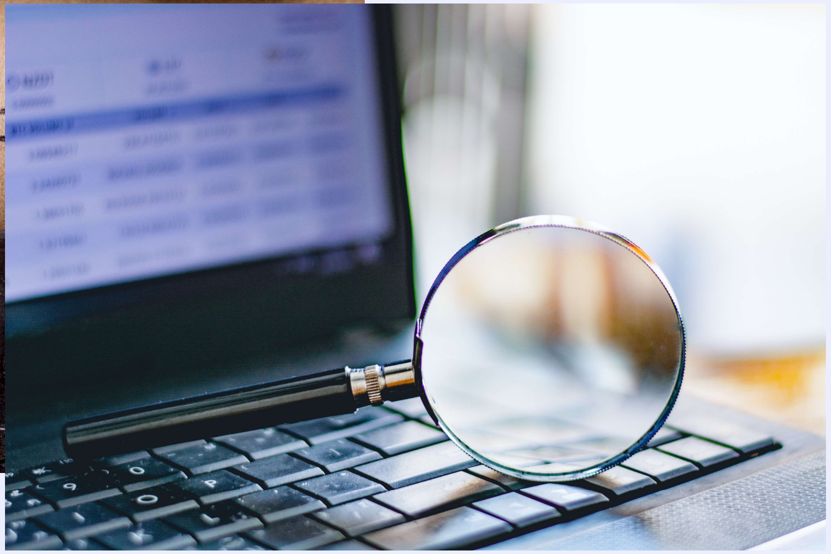
BUT ENOUGH ABOUT ME…

Tell me about you!

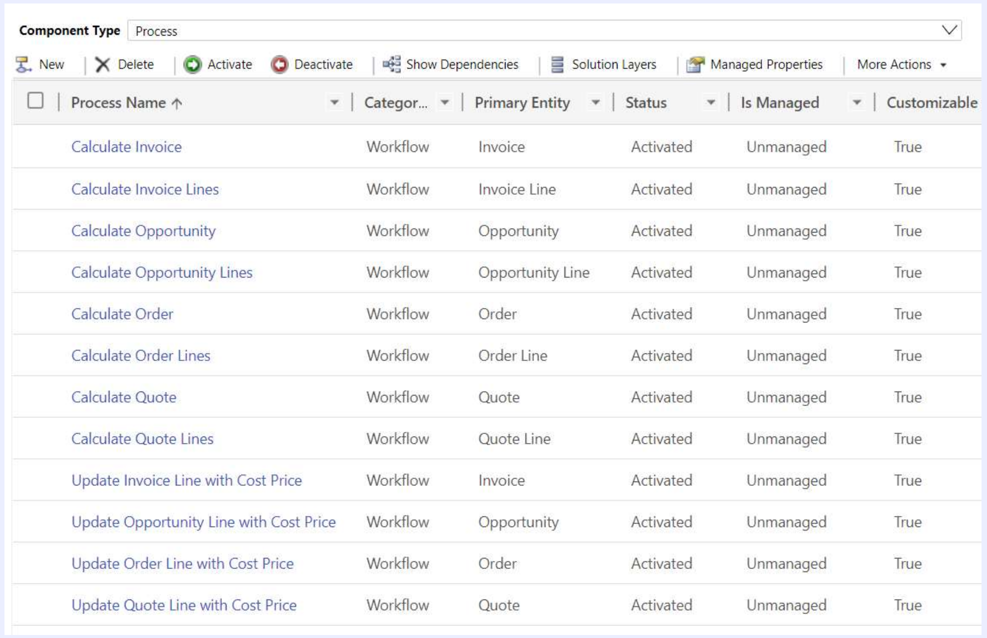
Are you:

- An experienced .NET Developer?
- A functional specialist/administrator?
- Wondering why I am talking about musical notes at a technical conference?
- In the wrong room?

STORY TIME



STORY TIME


Component Type: Process

New | Delete | Activate | Deactivate | Show Dependencies | Solution Layers | Managed Properties | More Actions

Process Name	Categor...	Primary Entity	Status	Is Managed	Customizable
Calculate Invoice	Workflow	Invoice	Activated	Unmanaged	True
Calculate Invoice Lines	Workflow	Invoice Line	Activated	Unmanaged	True
Calculate Opportunity	Workflow	Opportunity	Activated	Unmanaged	True
Calculate Opportunity Lines	Workflow	Opportunity Line	Activated	Unmanaged	True
Calculate Order	Workflow	Order	Activated	Unmanaged	True
Calculate Order Lines	Workflow	Order Line	Activated	Unmanaged	True
Calculate Quote	Workflow	Quote	Activated	Unmanaged	True
Calculate Quote Lines	Workflow	Quote Line	Activated	Unmanaged	True
Update Invoice Line with Cost Price	Workflow	Invoice	Activated	Unmanaged	True
Update Opportunity Line with Cost Price	Workflow	Opportunity	Activated	Unmanaged	True
Update Order Line with Cost Price	Workflow	Order	Activated	Unmanaged	True
Update Quote Line with Cost Price	Workflow	Quote	Activated	Unmanaged	True

X 4!!

SO WHAT ARE THE PROBLEMS HERE?

- Functional solutions are great, but not when they reverse engineer a system.
- Greater risk to the business, via a convoluted and poorly implemented solution.
- To ~~completely misquote~~ quote Albert Einstein:
  - *"If you can't express your business logic in less than 3 workflows, you don't understand it yourself!"*
- Poor appreciation of the capabilities within Customer Engagement.
  - Greatest sin of all ☺

CUSTOM PRICING OVERVIEW

- Provides a mechanism to completely replace the out of the box calculation engine within Customer Engagement.
- Developers disable out of the box calculations and then inject any custom logic as part of the **CalculatePrice** message.
- This message triggers whenever the following record types are retrieved, created or updated:
    - Opportunity (*opportunity*)
    - Opportunity Product (*opportunityproduct*)
    - Quote (*quote*)
    - Quote Product (*quotedetail*)
    - Order (*salesorder*)
    - Order Product (*salesorderdetail*)
    - Invoice (*invoice*)
    - Invoice Product (*invoicedetail*)
- Custom logic is built out using C# or Visual Basic .NET (VB.NET), using the SDK.

KEY BENEFITS

- Exposes the full capabilities within platforms SDK.
- Allows for practically *any* custom business logic to be applied for sales calculations.
- Developers can tailor business logic to apply to one, several or all sales entities at once.
- Effective way of isolating complex logic into a simple, containerised solution, that can be deployed to multiple environments with ease.
- Logic is applied at a platform level, thereby allowing any custom solutions (PowerApps etc.) to obey and enforce.

POTENTIAL USAGE SCENARIOS

- Query an external system/API to retrieve product pricing information and apply this within any sales calculations.
- Include custom fields within sales calculations.
- Display bespoke error messages to users if business conditions are violated.
- Compare before/after sales calculation values and trigger custom logic based on comparisons.
- In other words, the world is your oyster!

GETTING STARTED

- There's a few things you need to start building your own custom pricing solution:
  - Online Customer Engagement tenant or on-premise Dynamics CRM organisation
  - Visual Studio (any version from 2015 onwards)
  - Plug-in Registration Tool
- An awareness of how to work with the following tools is advantageous:
  - C# (or VB.NET for hard mode)
  - NuGet
  - A general awareness of the CRM/Customer Engagement SDK
  - Experience managing/deploying plug-ins via the Plug-in Registration Tool

We will cover all of this in today's session.

DEMO SCENARIO

- Our fictional organisation has been tasked with implementing a custom pricing solution, that achieves the following requirements:
    - Allow for Product Line item discounts to be applied, pre tax, by expressing a percentage at both Sales document (Opportunity, Quote etc.) & product level.
    - Calculates the freight amount for each product line item, based on the location of the parent sales document record.
    - Displays an error message to users if they attempt to sell a product at below cost value.
- All functional components of the solution (forms, fields etc.) have already been built out for us; all that's left is to start coding!

THE PROCESS

1. Review Environment
2. Create Visual Studio Project Setup/Configuration
3. Code* the Solution
4. Deploy
5. Test

*With some cheating involved

FOLLOW ALONG YOURSELF!

- A managed solution, containing the components we will work through, can be found on my GitHub page.
- Also contains a base Visual Studio solution file, that can be developed further, alongside detailed instructions.

https://github.com/JJGriffin/talk-assets/tree/master/D365CECustomPricing

LET'S CODE!

THINGS TO WATCH OUT FOR

- Checking the ParentContext and including a SharedVariable is a mandatory requirement, to prevent infinite loops.
- If you are using Project Service Automation (PSA), note that there will be some additional work involved to make any custom pricing solution compatible with this application.
  - e.g. PSA enforces a requirement that any totals for a product line item tally up with any related line item detail records.
- Be aware of potential issues if moving from on-premise to v9.x online when you have a custom pricing solution in place.
- Consider the impact a custom pricing solution will have in the context of the recent API limit changes.
- A custom pricing solution will <u>not</u>:
  - Let you perform calculations on custom entities.
  - Ignore any other general limitations concerning plug-ins.
  - Allow you to “pick and choose” which entities are enabled for custom pricing; it’s all or nothing!

WRAPPING UP

- For situations where an organisations Sales process doesn’t “fit” well into Customer Engagement, custom pricing is an invaluable feature to have available.
- Thanks to extensive code examples and instructions, it is relatively straightforward (with a bit of C# knowledge) to get up and running with a custom pricing solution.
- New API limits could cause a potential barrier in its future adoption.
- Does introduce a degree of complexity into your Customer Engagement deployment.

FURTHER READING

- Microsoft Docs Tutorials/Articles:
  - Use custom pricing for products
  - Sample: Calculate Price plug-in
  - Power Platform Requests limits and allocations
- Blog Posts:
  - A Few Observations on Using Custom Pricing Plugins Alongside Project Service Automation
  - Automatically Populate Extended Amount Field When Using Custom Pricing (Dynamics CRM/365 for Enterprise)
  - Implementing Custom Calculations for Sales Entities (Dynamics CRM/Dynamics 365 for Enterprise)
  - Mapping Product Attributes to Quote/Order/Invoice Line Items (Dynamics 365 Customer Engagement)

THANK YOU FOR LISTENING! ANY QUESTIONS…?

Joe Griffin

https://crmchap.co.uk

joe@griffin.gb.net

joejgriffin

joejgriffin

JJGriffin