



## Microsoft AI

### Modern APIs





[Profil](#)



[@MMe2K](#)

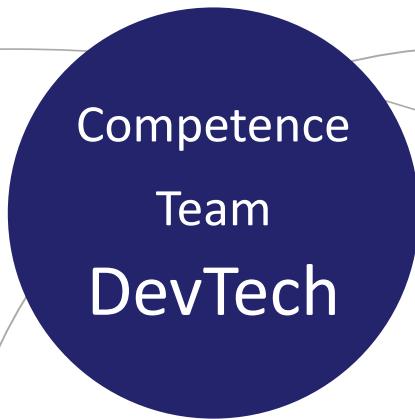


Teams-Channel  
[Microsoft AL](#)

*Never stop learning!*

Technology Evangelist

*Microsoft AL – Business Central – Azure – Cognitive Services – PowerShell –  
Docker – Git – DevOps – .NET – C# – JavaScript – SQL – LINQ – ...*



Michael Megel  
Business Development @ cc Dresden

# VS Code Useful Extensions

- > [AL Formatter](#)
- > [AL Code Outline](#)
- > [AL Object Designer](#)
- > [CRS AL Language Extension](#)
- > [JSON5 syntax](#)
- > [AL Runner](#)
- > [REST Client](#)
  
- > [VS Live Share](#)
- > [Visual Studio Live Share Extension Pack](#)
- > [Visual Studio IntelliCode](#)

AL   Object Designer   6316 objects					Search	Search for objects
All	Tables	Pages	Reports	Codeunits	Queries	XmlPorts
Workspace	Table Ext.	Page Ext.	Page Cust.	Control Add.	Enums	Profiles
Events	Refresh	+ New Object	+ New Dictionary	+ New Entry Table		
TYPE	ID	NAME		APPLICATION	VERSION	PUBLISHER
Table	3	<a href="#">Payment Terms</a>		Application	14.0.27306.0	Microsoft
Table	4	<a href="#">Currency</a>		Application	14.0.27306.0	Microsoft
Table	5	<a href="#">Finance Charge Terms</a>		Application	14.0.27306.0	Microsoft
Table	6	<a href="#">Customer Price Group</a>		Application	14.0.27306.0	Microsoft
Table	7	<a href="#">Standard Text</a>		Application	14.0.27306.0	Microsoft
Table	8	<a href="#">Language</a>		Application	14.0.27306.0	Microsoft
Table	9	<a href="#">Country/Region</a>		Application	14.0.27306.0	Microsoft
Table	10	<a href="#">Shipment Method</a>		Application	14.0.27306.0	Microsoft
Table	11	<a href="#">Country/Region Translation</a>		Application	14.0.27306.0	Microsoft



# Business Central



- > Overview
  - > Explore & Setup
- > Microsoft AI
  - > Web Services
  - > OData Feeds
  - > Modern APIs
- > Tools & Simulation
- > Hand On



# Business Central

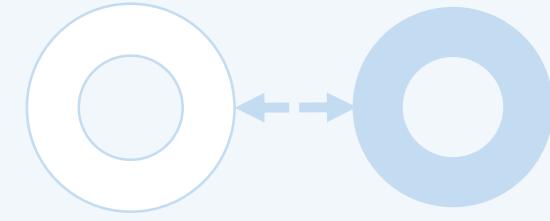
## *Extending and integrating*



### Add-On

*Add-on apps enrich the Business Central functionality*

### Extensions 2.0



### Connect

*Connected Apps connect another service with Business Central*

### API's

# Business Central

## Add-on vs. Connect apps

➤ Connect apps supplement add-ons!

➤ You can even combine the two, relying on an add-in for setup UI or two way communication

	Strengths	"Weaknesses"	Suitable for
Add-on apps	Powerful customization abilities  In-client designer	Require knowledge of internal business logic and data model	Deep integration  Embedded user experience  ISV solutions
Connect apps	Simplicity of REST api's  Similarity with other SaaS services  Fairly quick development cycle  Large choice of coding platforms	Cannot modify UI or business logic	Data integration  Integration with online services, SaaS ISV solutions  Developers with no Dynamics 365 Business Central experience

# Business Central API and Web Services

## › SOAP web services

- › Codeunit
- › Page
- › Query

## › OData web services

- › Page
- › Query

## › API endpoints

- › Page      *PageType = API*

The screenshot shows a search interface titled "TELL ME WHAT YOU WANT TO DO". A search bar contains the text "web". Below the search bar, there is a link "Go to Pages and Tasks" and a "Show all (5)" button. Three items are listed below:

- > Web Sources      Administration
- > Web Services      Lists
- > PAC Web Services      Lists

### Access the Dynamics 365 Business Central Sandbox Environment using Web Services

The Dynamics 365 Business Central Sandbox Environment exposes functionality as SOAP web services. Choose this link to view the web services.

The Dynamics 365 Business Central Sandbox Environment exposes data as restful OData web services. Choose this link to view the web services

The Dynamics 365 Business Central Sandbox Environment exposes the API endpoint. APIs are exposed using <API Base Url>/<Publisher>/<Group>/<Version>/<Entity>. Choose this link to get the base URL for the APIs.

Example: [https://\[REDACTED\].westeurope.cloudapp.azure.com:7048/nav/api/microsoft/automation/beta/companies](https://[REDACTED].westeurope.cloudapp.azure.com:7048/nav/api/microsoft/automation/beta/companies)

[View SOAP Web Services](#)  
[View OData Web Services](#)  
[View API base URL](#)

# Business Central

## Explore classic Web Services

- [ALT] + [Q] ... Tell Me ... Web Services
- Codeunits, Pages & Queries

WEB SERVICES

Web Services						
	<input type="button" value="Search"/>	<input type="button" value="New"/>	<input type="button" value="Edit List"/>	<input type="button" value="Delete"/>	<input type="button" value="Process"/>	<input type="button" value="Open in Excel"/>
OBJECT TYPE	OBJECT ID	OBJECT NAME	SERVICE NAME	ALL TENANT	PUBLISHED	ODATA V4 URL
Codeunit	1801	Company Setup Service	CompanySetupService	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Not applicable
Codeunit	1803	Encrypted Key/Value Management	EncryptedKeyValue	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Not applicable
Codeunit	5324	Exchange Service Setup	ExchangeServiceSetup	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Not applicable
Codeunit	1544	Workflow Webhook Subscription	WorkflowActionResponse	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not applicable
Page	1314	AccountantPortal Activity Cues	AccountantPortalActivityCues	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	1315	Accountant Portal Finance Cues	AccountantPortalFinanceCues	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	1316	Accountant Portal User Tasks	AccountantPortalUserTasks	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	2160	O365 Sales Graph	C2Graph	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	16	Chart of Accounts	Chart of Accounts	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	5500	agedAccountsPayable	ExcelTemplateAgedAccountsPaya...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	5499	agedAccountsReceivable	ExcelTemplateAgedAccountsRece...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	5501	balanceSheet	ExcelTemplateBalanceSheet	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	5493	cashFlowStatement	ExcelTemplateCashFlowStatement	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>
Page	5503	incomeStatement	ExcelTemplateIncomeStatement	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a> <a href="https://bc-spring-us.westeurope.c...">https://bc-spring-us.westeurope.c...</a>

# Business Central

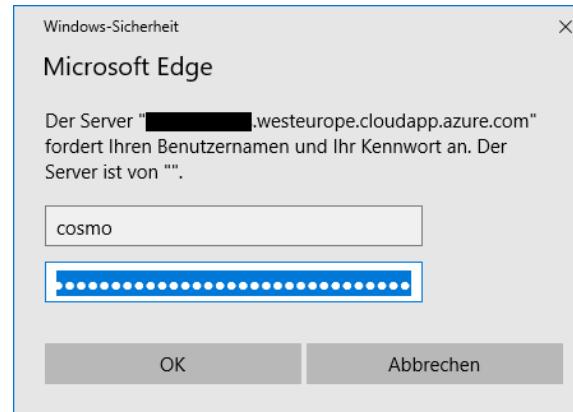
## Explore classic Web Services

### › Copy the URL from

- › Web Services Page
- › Landing Page

### › Basic Authentication

- › Username
- › Web Access Key



```
<?xml version="1.0" encoding="ISO-8859-1"?>
<service xmlns:atom="http://www.w3.org/2005/Atom" xmlns="http://wv
  - <workspace>
    <atom:title>Default</atom:title>
    - <collection href="AccountantPortalActivityCues">
      <atom:title>AccountantPortalActivityCues</atom:title>
    </collection>
    - <collection href="AccountantPortalFinanceCues">
      <atom:title>AccountantPortalFinanceCues</atom:title>
    </collection>
    - <collection href="AccountantPortalUserTasks">
      <atom:title>AccountantPortalUserTasks</atom:title>
    </collection>
    - <collection href="C2Graph">
      <atom:title>C2Graph</atom:title>
    </collection>
    - <collection href="InvoiceDocument">
      <atom:title>InvoiceDocument</atom:title>
    </collection>
    - <collection href="InvoiceReminder">
      <atom:title>InvoiceReminder</atom:title>
    </collection>
    - <collection href="nativeInvoicingAttachments">
      <atom:title>nativeInvoicingAttachments</atom:title>
    </collection>
    - <collection href="nativeInvoicingContacts">
      <atom:title>nativeInvoicingContacts</atom:title>
    </collection>
```

### Access the Dynamics 365 Business Central Sandbox Environment using Web Services

The Dynamics 365 Business Central Sandbox Environment exposes functionality as SOAP web services. Choose this link to view the web services.

The Dynamics 365 Business Central Sandbox Environment exposes data as restful OData web services. Choose this link to view the web services

The Dynamics 365 Business Central Sandbox Environment exposes the API endpoint. APIs are exposed using <API Base Url>/<Publisher>/<Group>/<Version>/<Entity>. Choose this link to get the base URL for the APIs.

Example: <https://████████.westeurope.cloudapp.azure.com:7048/nav/api/microsoft/automation/beta/companies>

[View SOAP Web Services](#)  
[View OData Web Services](#)  
[View API base URL](#)

ryRegion">  
yRegion</atom:title>  
mers">  
ters</atom:title>  
mersCoupons">  
iersCoupons</atom:title>  
Preview">

# Business Central

## Setup Web Service Access

- Open „Users“ and select the right user
- Open „User Card“
- Create a „Web Service Access“ Key
  - Adjust the Key Expire
  - Confirm
  - Copy the Key
- Authentication
  - Basic
  - User & Password

The screenshot shows three windows related to setting up Web Service Access:

- User Card:** A standard user profile screen with fields for User Name (COSMO), State (Enabled), Full Name, and Contact Email. A red box highlights the "Web Service Access" section, which contains fields for "Web Service Access K..." and "Web Service Expiry D...". A red arrow points to the "..." button next to the key field.
- EDIT - SET WEB SERVICE ACCESS KEY:** A modal dialog with two options: "Key Never Expires" (selected) and "Key Expiration Date". It includes "OK" and "Cancel" buttons.
- Web Service Access:** A specific configuration screen where the "Web Service Access Key" field is populated with the value "r02uW/C6aaEO1F1YpU7M10TMijXBNFcwxAX661no4Q=". This field is also highlighted with a red box.

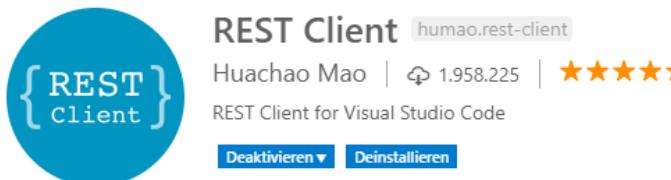
# Business Central

## Tools for Web Service Development & Test

- Fiddler
- Postman
- VS Code REST Client

The screenshot shows the Microsoft Dynamics 365 Business Central REST Client interface. The top navigation bar includes 'Import', 'Runner', 'My Workspace' (with an 'Invite' button), and several tool buttons like 'uninstall ext...', 'install ext...', 'Insert RSP', 'Upload RSP', 'Get RSPac', 'Import RS', 'Apply RSP', and 'APIs'. The main area is titled 'APIs' and shows a 'GET' request with the URL placeholder '{{url}}'. The 'Authorization' tab is selected, showing 'OAuth 2.0' as the type. Below it, there's a note about generating authorization data and buttons for 'Request Headers' and 'Preview Request'. The 'Body' tab is selected, displaying a JSON response structure. The 'Pretty' view shows the JSON with line numbers. The 'Raw' view shows the raw JSON text. The 'Preview' and 'JSON' tabs are also visible. To the right, there are tabs for 'Cookies (1)', 'Headers (17)', and 'Test Results (2/3)'. At the bottom, there's a code editor window with the following code:

```
9  ## Get all Companies
10 Send Request
11 GET {{apiHost}}:7048/nav/api/microsoft/automation/beta/companies
12 Content-Type: application/x-www-form-urlencoded
13 Authorization: Basic {{apiUser}} {{apiToken}}
```





Hands On



# Business Central

## *Hands-On*

### ➤ Prerequisites

- Connect to your Workshop VM → use Instructions on your Workshop Boarding Pass

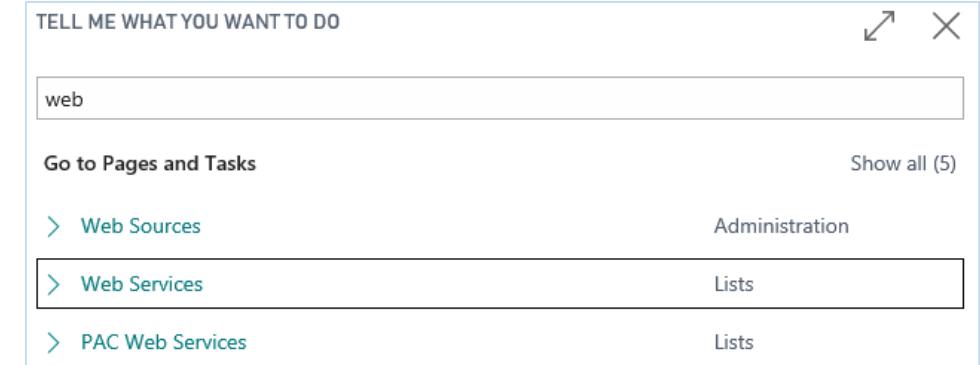
### ➤ Hand On - Tasks

- Learn how to explore Web Services
- Setup the Web Account in your environment

# Business Central

## Explore the Web Services

- Open your Web Client
- Use “Tell me” and navigate to the web services
  - Page Name “Web Services”
- Note the Web Service URL (for later) of:
  - Page “User Tasks”
  - Query “Vendor Ledger Entries”

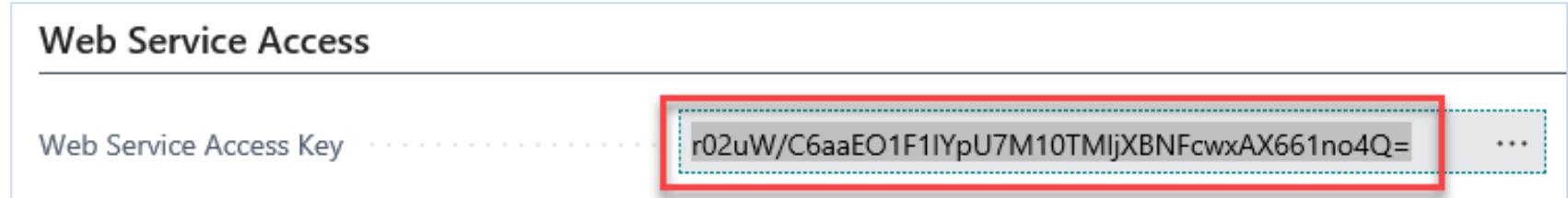
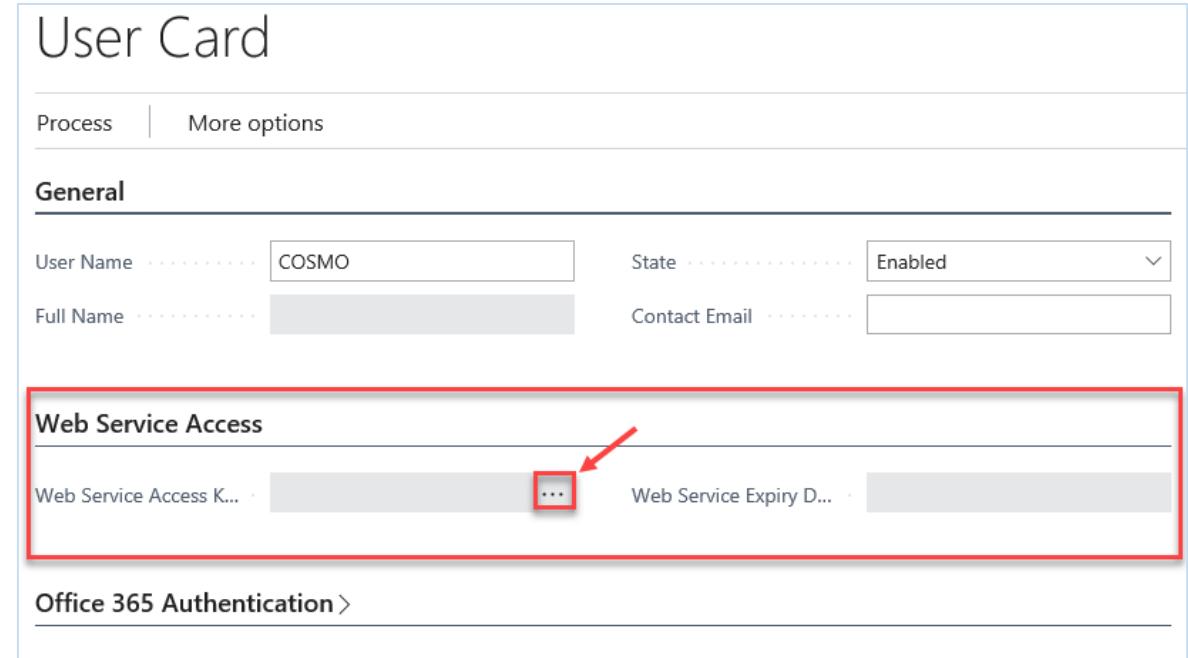


WEB SERVICES				
OBJECT TYPE	OBJECT ID	OBJECT NAME	SERVICE NAME	ALL TENAI
Codeunit	1801	Company Setup Service	CompanySetupService	<input checked="" type="checkbox"/>
Codeunit	1803	Encrypted Key/Value Management	EncryptedKeyValue	<input checked="" type="checkbox"/>
Codeunit	5324	Exchange Service Setup	ExchangeServiceSetup	<input checked="" type="checkbox"/>
Codeunit	1544	Workflow Webhook Subscription	WorkflowActionResponse	<input type="checkbox"/>
Page	1314	AccountantPortal Activity Cues	AccountantPortalActivityCues	<input checked="" type="checkbox"/>
Page	1315	Accountant Portal Finance Cues	AccountantPortalFinanceCues	<input checked="" type="checkbox"/>
Page	1316	Accountant Portal User Tasks	AccountantPortalUserTasks	<input checked="" type="checkbox"/>
Page	2160	O365 Sales Graph	C2Graph	<input checked="" type="checkbox"/>
Page	16	Chart of Accounts	Chart of Accounts	<input type="checkbox"/>
Page	5500	agedAccountsPayable	ExcelTemplateAgedAccountsPay...	<input type="checkbox"/>
Page	5499	agedAccountsReceivable	ExcelTemplateAgedAccountsRece...	<input type="checkbox"/>
Page	5501	balanceSheet	ExcelTemplateBalanceSheet	<input type="checkbox"/>
Page	5493	cashFlowStatement	ExcelTemplateCashFlowStatement	<input type="checkbox"/>
Page	5503	incomeStatement	ExcelTemplateIncomeStatement	<input type="checkbox"/>

# Business Central

## Setup your Web Service Access

- Open your Web Client
- Use “Tell me” and navigate to the web services
  - Page Name “Users”
- Open User Page
- Setup the Web Service Access
  - no expire date
  - Note your generated **Web Service Access Key** and **Username**





# Microsoft AL

## Web Services & OData Feeds



# Business Central | Web Services

## Codeunit

### > Codeunit

- SOAP only
- Wrapper for Methods

### > Procedure

- Web Service Method for SOAP

### > Method Arguments

- Request Parameters in SOAP

### > Method Result

- Response Data in SOAP

```
codeunit 50202 BeerWebService
{
    0 references
    procedure GetBeerCategories(code: Text): Text;
    begin
        end;
}
```

```
<types>
    - <schema xmlns="http://www.w3.org/2001/XMLSchema" targetNamespace="urn:microsoft-dynamics-schema">
        - <element name="GetBeerCategories">
            - <complexType>
                - <sequence>
                    <element name="code" type="string" maxOccurs="1" minOccurs="1"/>
                </sequence>
            </complexType>
        </element>
        - <element name="GetBeerCategories_Result">
            - <complexType>
                - <sequence>
                    <element name="return_value" type="string" maxOccurs="1" minOccurs="1"/>
                </sequence>
            </complexType>
        </element>
    </schema>
</types>
```

# Business Central | Web Services

## Query

### > Query

- SOAP, OData, OData v4
- QueryType = Normal

### > Data Items

- Table
- Columns

### > Nested Data Items

- DataItemLink specify the JOIN  
*Sets a reference between one or more fields of the data item table and data item or data items above it.*
- DataItemTableFilter  
*Sets filters on fields of the underlying table of a query.*
- SQLJoinType may replace the DataItemLinkType  
*Sets the data item link type between data items in a query to determine the records that are included in the resulting dataset.*

```
0 references
query 50200 "Beer Query [CC]"
{
    QueryType = Normal;

    elements
    {
        1 reference
        dataitem(Item; Item)
        {
            0 references
            column(id; Id) { }

            0 references
            column(Name; Description) { }

            0 references
            column(BeerCategoryCode; "Beer Category Code [CC]") { }

            0 references
            column(BeerCategoryName; "Beer Category Name [CC]") { }

            0 references
            column(Description; Description) { }

            1 reference
            dataitem(Customer; "Customer")
            {
                DataItemLink = "Beer Category Code [CC]" = Item."Beer Category Code [CC]";
                0 references
                column(CustomerName; Name) { }

                0 references
                dataitem(Country_Region; "Country/Region")
                {
                    DataItemLink = Code = Customer."Country/Region Code";
                    0 references
                    column(CountryRegionName; Name) { }
                }
            }
        }
    }
}
```

# Business Central | Web Services

## Page

### > Page

- SOAP
- OData
- OData v4

### > Access to Page Records

### > Actions as Web Service Methods

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<definitions xmlns:tns="urn:microsoft-dynamics-schemas/page/usertasksetcomplete" xmlns="http://schemas.xmlsoap.org/wsdl/" targetName="UserTaskSetComplete">
  - <types>
    - <xsd:schema targetNamespace="urn:microsoft-dynamics-schemas/page/usertasksetcomplete" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      - <xsd:simpleType name="Priority">
        - <xsd:restriction base="xsd:string">
          <xsd:enumeration value="Low"/>
          <xsd:enumeration value="Normal"/>
          <xsd:enumeration value="High"/>
        </xsd:restriction>
      </xsd:simpleType>
    - <xsd:complexType name="UserTaskSetComplete">
      - <xsd:sequence>
        <xsd:element name="Key" type="xsd:string" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="Title" type="xsd:string" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="Due_DateTime" type="xsd:dateTime" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="Priority" type="tns:Priority" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="Percent_Complete" type="xsd:int" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="Assigned_To_User_Name" type="xsd:string" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="User_Task_Group_Assigned_To" type="xsd:string" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="Created_DateTime" type="xsd:dateTime" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="Completed_DateTime" type="xsd:dateTime" maxOccurs="1" minOccurs="0"/>
        <xsd:element name="Start_DateTime" type="xsd:dateTime" maxOccurs="1" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:schema>
</definitions>
```

```
1  HTTP/1.1 200 OK
2  Transfer-Encoding: chunked
3  Content-Type: application/json; odata.metadata=minimal; odata.streaming=true
4  Content-Encoding: gzip
5  Server: Microsoft-HTTPAPI/2.0
6  OData-Version: 4.0
7  Access-Control-Allow-Origin: *
8  Access-Control-Allow-Credentials: true
9  Access-Control-Expose-Headers: Date, Content-Length, Server, OData-Version
10 request-id: 6695b059-62c7-43f2-ad03-5ada1d8c3a9c
11 Date: Mon, 28 Jan 2019 09:22:12 GMT
12
13 {
14   "@odata.context": "https://bc-spring-us.westeurope.cloudapp.azure.com:7048/NAV/ODataV
4/$metadata#Company('CRONUS%20USA%2C%20Inc.')/UserTaskSetComplete",
15   "value": []
16 }
```

# Business Central Web Services Registration

## ➤ Object Types

- Page
- Query
- Codeunit

The screenshot shows a code editor with a snippet titled "Snippet: Web Service (AL Language)". The snippet contains XML code for registering a web service. It includes elements like <TenantWebService>, <ObjectType>0</ObjectType>, <ServiceName>serviceName</ServiceName>, <ObjectID>objectId</ObjectID>, and <Published>0</Published>. The code editor interface shows a tree view of available objects on the left.

```
<TenantWebService>
  <ObjectType>0</ObjectType>
  <ServiceName>serviceName</ServiceName>
  <ObjectID>objectId</ObjectID>
  <Published>0</Published>
```

## ➤ Codeunits

- Be clear, what you expose as method and parameter!!!  
*JsonObjects are not supported*

## ➤ Registration XML in AL-App

- XML File
- Use Snippets
- Add Objects for registration

The screenshot shows a code editor with a file named "WebServices.xml". The file contains XML code for registering two tenant web services. Each service has a specific object type, object ID, service name, and publication status. The XML uses <?xml version="1.0" encoding="utf-8"?> at the top and <ExportedData> and </ExportedData> tags at the bottom.

```
<?xml version="1.0" encoding="utf-8"?>
<ExportedData>
  <TenantWebServiceCollection>
    <TenantWebService>
      <ObjectType>Query</ObjectType>
      <ObjectID>50200</ObjectID>
      <ServiceName>beerQuery</ServiceName>
      <Published>true</Published>
    </TenantWebService>
    <TenantWebService>
      <ObjectType>CodeUnit</ObjectType>
      <ObjectID>50200</ObjectID>
      <ServiceName>myWebService</ServiceName>
      <Published>true</Published>
    </TenantWebService>
  </TenantWebServiceCollection>
</ExportedData>
```



Hands On



# Business Central *Hands-On*

## › Prerequisites

- Connect to your Workshop VM → use Instructions on your Workshop Boarding Pass

## › Hand On - Tasks

- Create Web Services with Microsoft AL
- Register Web Services in Microsoft AL

# Business Central

## Create a Web Services from Codeunit

- Create a new Codeunit

- Object ID 50105
- Name "Beer WebService [CC]"

- Add a method "GetBeerCategories"

- Parameter "code"      *Text*
- Result                  *Text*

```
0 references
codeunit 50105 "Beer WebService [CC]"
{
    0 references
    procedure GetBeerCategories(code: Text): Text;
    begin
        end;
}
```

# Business Central

## Create a Web Services from Query

### › Create a new Query

- › Object ID 50105
- › Name "Beer Items [CC]"
- › QueryType: Normal

### › Query from Item Table

- › Id, Name, Is Beer Item, Beer Category Code, Beer Category Name, Description

### › Join Customer Information to the query

- › Customer Name, Customer Region Name

```
        "value": [  
          {  
            "id": "28a3379c-232e-415c-963e-059bb07d1083",  
            "Name": "Atlantic Ale",  
            "BeerCategoryCode": "ALE",  
            "Description": "Atlantic Ale",  
            "BeerCategoryName": "Ale",  
            "CustomerName": "Adatum Corporation",  
            "CountryRegionName": "USA",  
            "AuxiliaryIndex1": "ATLANTIC ALE",  
            "AuxiliaryIndex2": "10000",  
            "AuxiliaryIndex3": "US"  
          },  
          {  
            "id": "1b83a6de-1d54-4c1b-a0f1-8b44a4031a96",  
            "Name": "Freiberger Pilsner",  
            "BeerCategoryCode": " PILS",  
            "Description": "Freiberger Pilsner",  
            "BeerCategoryName": "Pilsner",  
            "CustomerName": "Freiberger Pilsner",  
            "CountryRegionName": "Germany",  
            "AuxiliaryIndex1": "FREIBERGER PILSNER",  
            "AuxiliaryIndex2": "10001",  
            "AuxiliaryIndex3": "DE"  
          }  
        ]
```

# Business Central

## Register your Web Services

- Create a XML file
  - “WebServices.xml”
  - Query as “beerQuery”
  - CodeUnit as “BeerWebService”
- Add your Services
  - Use Snippets!
- Deploy and Check your Web Services
  - Open Page “Web Services” and find your Services

```
❶  WebServices.xml
❷  <?xml version="1.0" encoding="utf-8"?>
❸  <ExportedData>
❹  |   <TenantWebServiceCollection>
❺  |   |   <TenantWebService>
❻  |   |   |   <ObjectType>Query</ObjectType>
❼  |   |   |   <ObjectID>50100</ObjectID>
➋  |   |   |   <ServiceName>beerQuery</ServiceName>
⌃  |   |   |   <Published>true</Published>
⌄  |   |   </TenantWebService>
⌅  |   <TenantWebService>
⌆  |   |   <ObjectType>CodeUnit</ObjectType>
⌇  |   |   <ObjectID>50105</ObjectID>
⌈  |   |   <ServiceName>BeerWebService</ServiceName>
⌉  |   |   <Published>true</Published>
⌊  |   </TenantWebService>
⌋  </TenantWebServiceCollection>
⌌  </ExportedData>
```



# Microsoft AI

## Modern APIs – Part I



# Business Central | Modern API

## Available APIs (48)

Company	Finance	Sales	Purchasing	Reports
company	accounts	customers	irs1099Codes (US only)	agedAccountsPayable
companyInformation	dimensions	customerPayments	purchaseInvoices	agedAccountsReceivable
countriesRegions	dimensionLines	customerPaymentJournals	purchaseInvoiceLines	balanceSheet
currencies	dimensionValues	salesInvoices	vendors	cashFlowStatement
employees	generalLedgerEntries	salesInvoiceLines		incomeStatement
items	journals	salesOrders		retainedEarningsStatement
itemCategories	journalLines	salesOrderLines		customerSales
paymentMethods		salesQuotes		vendorPurchases
paymentTerms		salesQuoteLines		trialBalance
shipmentMethods		salesCreditMemos		
taxAreas		salesCreditMemoLines		
taxGroups				
unitsOfMeasure				
picture, defaultDimension, attachments, pdfDocument				

# Business Central | Modern API

## GET endpoints (API list)

### › URL

- › <hostname>:<API port>/nav/api/<publisher>/<groupname>/<version>/
- › {{apiHost}}:7048/nav/api/microsoft/automation/beta/

### › Get endpoint – API list

- › add            /\$metadata

### › Operations

- › Get, Post, Patch, Delete

### › OData Filters

- › ?\$filter=<fieldname> <condition> <value> [...]
- › ?\$filter=beerCategoryCode eq 'ALE' or beerCategoryCode eq ' PILS'

```
<?xml version="1.0" encoding="utf-8"?>
<edmx:Edmx Version="4.0"
  xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx">
  <edmx:DataServices>
    <Schema Namespace="Microsoft.NAV"
      xmlns="http://docs.oasis-open.org/odata/ns/edm">
      <ComplexType Name="dateTimeTimeZoneType">
        <Property Name="dateTime" Type="Edm.String" Nullable="false" />
      </ComplexType>
      <ComplexType Name="dimensionType">
        <Property Name="code" Type="Edm.String" Nullable="false" />
        <Property Name="displayName" Type="Edm.String" />
        <Property Name="valueCode" Type="Edm.String" Nullable="false" />
        <Property Name="valueDisplayName" Type="Edm.String" />
      </ComplexType>
      <ComplexType Name="documentLineObjectDetailsType">
        <Property Name="number" Type="Edm.String" MaxLength="20" />
        <Property Name="displayName" Type="Edm.String" MaxLength="100" />
      </ComplexType>
      <ComplexType Name="unitOfMeasureType">
        <Property Name="code" Type="Edm.String" MaxLength="10" />
        <Property Name="displayName" Type="Edm.String" MaxLength="50" />
        <Property Name="symbol" Type="Edm.String" MaxLength="10" />
        <Property Name="unitConversion" Type="Microsoft.NAV.itemUnitConversionType" />
      </ComplexType>
      <ComplexType Name="itemUnitOfMeasureConversionType">
```

# Business Central | Modern API *examples*

```
### Get all Companies
Send Request
GET {{apiHost}}:7048/nav/api/microsoft/automation/beta/companies
Authorization: Basic {{apiUser}} {{apiToken}}


### Get all entities from API
Send Request
GET {{apiHost}}:7048/nav/api/microsoft/automation/beta/companies({{companyID}})
Authorization: Basic {{apiUser}} {{apiToken}}


### Get all entities from API
Send Request
GET {{apiHost}}:7048/nav/api/{{apiPublisher}}/{{apiGroup}}/{{apiVersion}}/companies({{companyID}})/{{entities}}
Authorization: Basic {{apiUser}} {{apiToken}}


### Add a new record to API
Send Request
POST {{apiHost}}:7048/nav/api/{{apiPublisher}}/{{apiGroup}}/{{apiVersion}}/companies({{companyID}})/{{entities}}
Content-Type: application/json
Authorization: Basic {{apiUser}} {{apiToken}}


{
    "no": "FREIBERGER",
    "beerCategoryCode": 'PILS',
    "name": "Freiberger Pils",
}
```

# Business Central | Modern API Query

## > Query

- OData v4
- QueryType = API

## > Properties

- APIPublisher
- APIGroup
- APIVersion                    *beta, v1.0, ...*
- EntityName
- EntitySetName

## > Data Items, Nested Data Items, ...

- Names & Columns            *Lower Camel Case*

```
query 50201 "Beer Customer API [CC]"
{
    QueryType = API;
    APIPublisher = 'Workshop';
    APIGroup = 'FavoritBeer';
    APIVersion = 'v1.0';
    EntityName = 'beerCustomer';
    EntitySetName = 'beerCustomers';

    elements
    {
        1 reference
        dataitem(Item; Item)
        {
            0 references
            column(id; Id) { }

            0 references
            column(name; Description) { }

            0 references
            column(beerCategoryCode; "Beer Category Code [CC]") { }

            0 references
            column(beerCategoryName; "Beer Category Name [CC]") { }

            0 references
            column(description; Description) { }
        }
    }
}
```

# Business Central | Modern API

## Page

### > Page

- › generate web service endpoints
- › cannot be displayed in the user interface
- › PageType = API  
..../Workshop/FavoritBeer/v1.0/companies({id})/beerCustomers

### > Properties

- › APIPublisher
- › APIGroup
- › APIVersion                           *beta, v1.0, ...*
- › EntityName
- › EntitySetName
- › SourceTable
- › ODataKeyFields                       *Guid*

### > Repeater & Fields

- › Names                               *Lower Camel Case*

```
query 50201 "Beer Customer API [CC]"
{
    QueryType = API;
    APIPublisher = 'Workshop';
    APIGroup = 'FavoritBeer';
    APIVersion = 'v1.0';
    EntityName = 'beerCustomer';
    EntitySetName = 'beerCustomers';

    elements
    {
        1 reference
        dataitem(Item; Item)
        {
            0 references
            column(id; Id) { }
            0 references
            column(name; Description) { }
            0 references
            column(beerCategoryCode; "Beer Category Code [CC]") { }
            0 references
            column(beerCategoryName; "Beer Category Name [CC]") { }
            0 references
            column(description; Description) { }
        }
    }
}
```

# Business Central | Modern API Simulation & Test ... REST Client

- Simulation of HTTP calls
- Create HTTP file (or activate the http extension) for simulation
- Setup Variables for calls
  - @apiHost
  - @apiUser
  - @apiToken
  - ...



**REST Client** humao.rest-client  
Huachao Mao | ↗ 1.958.225 | ★★★★★  
REST Client for Visual Studio Code  
[Deaktivieren ▾](#) [Deinstallieren](#)

```
Simulation ▶  Query.http ▶ ...
3 references
1 @apiHost = https://[REDACTED].westeurope.cloudapp.azure.com
3 references
2 @apiUser = COSMO
3 references
3 @apiToken = r02uW/C6aaE01F1lYpU7M10TMljXBNFcwxAX661no4Q=
2 references
4 @apiPublisher = WS
2 references
5 @apiGroup = beer
2 references
6 @apiVersion = v1.0
2 references
7 @copmanyID = <companyId>
8
9 ### Get all Companies
Send Request
10 GET {{apiHost}}:7048/nav/api/microsoft/automation/beta/companies
11 Content-Type: application/x-www-form-urlencoded
12 Authorization: Basic {{apiUser}} {{apiToken}}
```



Hands On



# Business Central

## *Hands-On*

### ➤ Prerequisites

- Connect to your Workshop VM → use Instructions on your Workshop Boarding Pass

### ➤ Hand On - Tasks

- Simulation & Test
- Create an API Query with Microsoft AL
- Create an API Page in Microsoft AL

# Business Central

## Hands-On | Simulation & Test ... REST Client

- Install VS Code Extension [REST Client](#)
- Create a Simulation Folder
- Add Simulation Files and Setup Variables for your environment
  - Page.http
  - Query.http
  - APIPage.http
  - APIQuery.http

```
Simulation
  APIPage.http
  APIQuery.http
  Page.http

0 references
@apiHost = <Your-Host-Name>
0 references
@apiUser = <User-Name>
0 references
@apiToken = <Web-Access-Token>
0 references
@apiPublisher = <API-Publisher>
0 references
@apiGroup = <API-Group>
0 references
@apiVersion = <API-Version>
0 references
@companyID = <Company-Id>
```

### Access the Dynamics 365 Business Central Sandbox Environment using Web Services

The Dynamics 365 Business Central Sandbox Environment exposes functionality as SOAP web services. Choose this link to view the web services.

The Dynamics 365 Business Central Sandbox Environment exposes data as restful OData web services. Choose this link to view the web services

The Dynamics 365 Business Central Sandbox Environment exposes the API endpoint. APIs are exposed using <API Base Url>/<Publisher>/<Group>/<Version>/<Entity>. Choose this link to get the base URL for the APIs.

Example: [https://\[REDACTED\].westeurope.cloudapp.azure.com:7048/nav/api/microsoft/automation/beta/companies](https://[REDACTED].westeurope.cloudapp.azure.com:7048/nav/api/microsoft/automation/beta/companies)

[View SOAP Web Services](#)  
[View OData Web Services](#)  
[View API base URL](#)

# Business Central

## Hands-On | Simulation & Test ... Explore the Services & API

### > Query the Metadata

- /\$metadata

```
### Get all "Microsoft.NAV" APIs
Send Request
GET {{apiHost}}:7048/nav/api/microsoft/automation/beta/$metadata
Authorization: Basic {{apiUser}} {{apiToken}}
```

### > Call „GetCompanies“

- Copy the Company Name for OData Feeds & Web Services
- Copy the Company ID for API calls

```
### Get all Companies
Send Request
GET {{apiHost}}:7048/nav/api/microsoft/automation/beta/companies
Content-Type: application/x-www-form-urlencoded
Authorization: Basic {{apiUser}} {{apiToken}}
```

### > Setup the CompanyID variable

### > Get the Company

- .../companies(<CompanyId>)

```
13  {
14    "@odata.context": "https://bc-spring-us.westeurope.cloudapp.azure.com:7048/nav/api/mi
t/automation/beta/$metadata#companies",
15  "value": [
16    {
17      "id": "11964762-da4a-4882-ba17-af8f89e90f5e",
18      "systemVersion": "27306",
19      "name": "CRONUS USA, Inc.",
20      "displayName": "CRONUS USA, Inc.",
21      "businessProfileId": ""
```

```
### Get the company from API
Send Request
GET {{apiHost}}:7048/nav/api/microsoft/automation/beta/companies({{copmanyID}})
Authorization: Basic {{apiUser}} {{apiToken}}
```

# Business Central

## *Hands-On | Simulation & Test ... Explore the Services & API*

- Pages „VendorLedgerEntries“
  - Complete URL information from your Note
  - Get the OData feed
  - Get the OData v4 feed
- Query „beerQuery“
  - Complete URL information from your Note
  - Get the OData feed
  - Get the OData v4 feed
  - Filter the OData v4 query by            *Name = 'Atlantic Ale'*
- Note: Company Name is needed for URL

`@copmpanyName = CRONUS%20USA%2C%20Inc.`

`GET {{apiHost}}:7048/NAV/ODataV4/Company('{{copmpanyName}}')`

# Business Central

## Hands-On | Create an API Query

### › Create a new API Query

- › Query ID 50105
- › Name "Beer Customer API [CC]"

### › Setup the Properties

#### › Query from Customer Table

- › Id, No., Name, Beer Category Code

#### › Join Item and Customer Information to the query

- › Item Description, Customer Region Name

```
query 50105 "Beer Customer API [CC]"
{
    QueryType = API;
    APIPublisher = 'Workshop';
    APIGroup = 'FavoritBeer';
    APIVersion = 'v1.0';
    EntityName = 'beerCustomer';
    EntitySetName = 'beerCustomers';
```

```
    "value": [
        {
            "id": "945a3654-16f8-45ed-8d06-65b743c6fa51",
            "no": "10000",
            "name": "Adatum Corporation",
            "beerCategoryCode": "ALE",
            "description": "Atlantic Ale",
            "countryRegionName": "USA",
            "auxiliaryIndex1": "ATLANTIC ALE",
            "auxiliaryIndex2": "US"
        },
        {
            "id": "6737f40f-a65b-4c44-a176-9764dba02656",
            "no": "20000",
            "name": "Trey Research",
            "beerCategoryCode": " PILS",
            "description": "Freiberger Pilsner",
            "countryRegionName": "USA".
```

# Business Central

## Hands-On | Create an API Page

### ➤ Create a new API Page

- Page ID 50105
- Name "Beer API [CC]"

### ➤ Add the fields to the repeater

```
0 references
field(id; Id)
{
    Caption = 'id';
}
0 references
field(beerCategoryCode; "Beer Category Code [CC]")
{
    Caption = 'beerCategoryCode';
}

0 references
field(beerCategoryName; "Beer Category Name [CC]")
{
    Caption = 'beerCategoryName';
}
0 references
field(name; Description)
{
    Caption = 'name';
}
```

```
page 50105 "Beer API [CC]"
{
    PageType = API;
    Caption = 'apiPageName';
    APIPublisher = 'Workshop';
    APIGroup = 'FavoritBeer';
    APIVersion = 'v1.0';
    EntityName = 'beer';
    EntitySetName = 'beers';
    SourceTable = item;
    DelayedInsert = true;
    ODataKeyFields = Id;

    layout
{
```

# Business Central

## Hands-On | Test API Query and API Page

### > Test API Pages „Beer API [CC]“

- Setup the variables in your simulation file (publisher, group, entity, entities, ...)
- Get all entities from „Beer API [CC]“ page {{apiVersion}}/companies({{copmanyID}})/{{entities}}
- Get one entity from „Beer API [CC]“ {{apiVersion}}/companies({{copmanyID}})/{{entities}}(<id>)
- Filter the entities by: Name = 'Atlantic Ale'  
.../{{entities}}?\$filter=...

```
{  
  "@odata.context": "https://bc-spring-us.westeurope.cloudapp.azure.com:7048/nav/api/Workshop/Favc  
$metadata#companies(11964762-da4a-4882-ba17-af8f89e90f5e)/beers/$entity",  
  "@odata.etag": "W/\"JzQ0O2tXUFlMS11SLzhpWi8zdE81ZFhVQ1lJWk8zTDE3OGdvW9xZmo3MGFRZW89MTswMDsn\"",  
  "id": "28a3379c-232e-415c-963e-059bb07d1083",  
  "beerCategoryCode": "ALE",  
  "beerCategoryName": "Ale",  
  "name": "Atlantic Ale"  
}
```

### > Test API Query „ Beer Customer API [CC]“

- Setup the variables in your simulation file (publisher, group, entity, entities, ...)
- Get all entities from „Beer API [CC]“ page {{apiVersion}}/companies({{copmanyID}})/{{entities}}
- Filter the entities by: Beer Category Code = 'ALE' or ' PILS'  
.../{{entities}}?\$filter=...

```
{  
  "@odata.context": "https://bc-spring-us.westeurope.cloudapp.azure.com:7048/n  
1.0/$metadata#companies(11964762-da4a-4882-ba17-af8f89e90f5e)/beerCustomers",  
  "value": [  
    {  
      "id": "945a3654-16f8-45ed-8d06-65b743c6fa51",  
      "no": "10000",  
      "name": "Adatum Corporation",  
      "beerCategoryCode": "ALE",  
      "description": "Atlantic Ale",  
      "countryRegionName": "USA",  
      "auxiliaryIndex1": "ATLANTIC ALE",  
      "auxiliaryIndex2": "US"  
    },  
    {  
    }
```



# Microsoft AI

## Modern APIs Part II



# Business Central

## *Navigation Properties*

```
<EntityType Name="salesOrder">
  <Key>
    <PropertyRef Name="id" />
  </Key>
  <Property Name="id" Type="Edm.Guid" Nullable="false" />
  <Property Name="number" Type="Edm.String" MaxLength="20" />
  <Property Name="externalDocumentNumber" Type="Edm.String" MaxLength="35" />
  <Property Name="orderDate" Type="Edm.Date" />
  <Property Name="customerId" Type="Edm.Guid" />
  <Property Name="contactId" Type="Edm.String" MaxLength="250" />
  <Property Name="customerNumber" Type="Edm.String" MaxLength="20" />
  .....
  <Property Name="lastModifiedDateTime" Type="Edm.DateTimeOffset" />
  <NavigationProperty Name="salesOrderLines" Type="Collection(Microsoft.NAV.salesOrderLine)" ContainsTarget="true" />
  <NavigationProperty Name="customer" Type="Microsoft.NAV.customer" ContainsTarget="true" />
  <NavigationProperty Name="currency" Type="Microsoft.NAV.currency" ContainsTarget="true" />
  <NavigationProperty Name="paymentTerm" Type="Microsoft.NAV.paymentTerm" ContainsTarget="true" />
</EntityType>
```

/salesOrders({{salesOrderId}})/customer

# Business Central */api Structure*

- @odata.etag
- id (GUID)
- lastModifiedDateTime
- Lower camel casing
- Complex Types

```
{  
    "@odata.etag": "W/\\"JzI400d3QUFBQUo3QkRFQU1BQXdBREFBQUFBPT03OzEzMjM2NTUwOyc=\\\"",  
    "id": "5b0b9c1c-312d-4809-96b2-056690a11057",  
    "number": "1000",  
    "displayName": "General Landscaping",  
    "type": "Service",  
    "blocked": false,  
    "baseUnitOfMeasure":  
        {  
            "code": "HOUR",  
            "displayName": "Hour",  
            "symbol": null,  
            "unitConversion": null  
        },  
    "gtin": "",  
    "itemCategory": null,  
    "inventory": -52,  
    "unitPrice": 50,  
    "priceIncludesTax": false,  
    "unitCost": 27,  
    "taxGroupCode": "LABOR",  
    "lastModifiedDateTime": "2017-08-29T18:51:23.17Z"  
}
```

# Business Central

## Deep Insert

- Enables creating a tree of related entities in one request

```
{  
    "invoiceDate": "2016-07-26",  
    "dueDate": "2016-08-26",  
    "customerId": "653053b1-d580-4102-845f-00df29a77e1e",  
    "salesInvoiceLines" : [  
        {"itemId":"5b0b9c1c-312d-4809-96b2-056690a11057", "quantity": 4},  
        {"itemId":"ad7220e5-8149-4cf0-a1e6-07d2bbbf414c", "quantity": 1}  
    ]  
}
```

# Business Central

## Bound Actions

### ➤ Actions called through the API

- Bound to an entity
- Actions coming from /api endpoint
- Can be created for ODataV4 web services

### ➤ Method inside of API Page

- Marked with Attribute “ServiceEnabled”
- Data Exchange with WebServiceActionContext

```
[ServiceEnabled()]
procedure execute(var actionContext: WebServiceActionContext)
var

actionContext.SetresultCode(WebServiceActionResultCode::Created);
```

### ➤ HTTP Call with POST

```
}}/{{entitiesName}}({{64a9fe2a-7d46-4e70-8ba5-0494e04ab1c8}})/Microsoft.NAV.execute
```

```
<EntityType Name="salesQuote">
  <Key>
    <PropertyRef Name="id" />
  </Key>
  <Property Name="id" Type="Edm.Guid" Nullable="false" />
  <Property Name="number" Type="Edm.String" MaxLength="20" />
  <Property Name="externalDocumentNumber" Type="Edm.String" MaxLength="35" />
  <Property Name="documentDate" Type="Edm.Date" />
  <Property Name="dueDate" Type="Edm.Date" />
  ...
  <Property Name="acceptedDate" Type="Edm.Date" />
  <Property Name="lastModifiedDateTime" Type="Edm.DateTimeOffset" />
  <NavigationProperty Name="salesQuoteLines" Type="Collection(Microsoft.NAV.sales
  ...
  <NavigationProperty Name="paymentTerm" Type="Microsoft.NAV.paymentTerm" Contain
  <NavigationProperty Name="shipmentMethod" Type="Microsoft.NAV.shipmentMethod" C
</EntityType>
<Action Name="makeInvoice" IsBound="true">
  <Parameter Name="bindingParameter" Type="Microsoft.NAV.salesQuote" />
</Action>
<Action Name="send" IsBound="true">
  <Parameter Name="bindingParameter" Type="Microsoft.NAV.salesQuote" />
</Action>
```

# Business Central

## *Versioning for APIs*

- Enabled versioning of API library
  - <https://api.businesscentral.dynamics.com/v1.0/api/beta/>
  - <https://api.businesscentral.dynamics.com/v1.0/api/v1.0/>

# Business Central

## *Upcoming Webhook support*

- Operations on subscription entity
  - GET – list all subscriptions for a company
  - POST – create a new webhook subscription
  - DELETE – delete a subscription
  - PATCH – Renew a subscription

# Business Central

## *Get notified - subscribe*

- POST ..../api/v1.0/subscriptions

```
{  
    "clientState": "secret",  
    "notificationUrl": "https://mywebsite.azurewebsites.net/api/webhook",  
    "resourceUrl": "/api/v1.0/companies(name='CRONUS International Ltd.')/salesInvoices"  
}
```

- Handshake for POST & PATCH needed.
- Both parties supplies a secret, which needs to be echoed back
  - Client provides clientSecret
  - BC provides validationToken

# Business Central

## *Notification payload*

```
[  
  {  
    "subscriptionId": "9c0b28a0-ae4a-4b66-881b-c8f3db3b802d",  
    "clientState": "API WEBHOOK NOTIFICATION TEST",  
    "expirationDateTime": "2018-09-21T16:08:28.597Z",  
    "resource": "/api/beta/companies({ID})/items(93aaa13a-251b-4c3b-ba80-ccc38c34c5e3)",  
    "changeType": "updated",  
    "lastModifiedDateTime": "2018-09-20T16:08:27.597Z"  
  },  
  {  
    "subscriptionId": "9c0b28a0-ae4a-4b66-881b-c8f3db3b802d",  
    "clientState": "API WEBHOOK NOTIFICATION TEST",  
    "expirationDateTime": "2018-09-21T16:08:28.597Z",  
    "resource": "/api/beta/companies({ID})/items(93b37d68-d179-4ef9-bd26-60eb024d9e68)",  
    "changeType": "created",  
    "lastModifiedDateTime": "2018-09-20T16:08:28.597Z"  
  },  
  {  
    "subscriptionId": "fe564bb8-a735-4a29-802c-ff6aa630557d",  
    "clientState": "API WEBHOOK NOTIFICATION TEST",  
    "expirationDateTime": "2018-09-25T19:16:35.17Z",  
  }]
```



Hands On



# Business Central *Hands-On*

## ➤ Prerequisites

- Connect to your Workshop VM → use Instructions on your Workshop Boarding Pass

## ➤ Hand On - Tasks

- Add, implement and test a Bound Action to the API Page
- Insert some Records with API (ID Guid is necessary)

# MICROSOFT AL



Channel

Competence Team DevTech

Microsoft AL

