

Visual Studio code and AL

Go to Agenda | My Agenda

Session: Visual Studio code and AL

Agenda

- Report extensibility
- Permissions in AL
- Complex return types
- Interface enhancements
- Returning records in lookups
- Onboarding
- Debugger enhancements
- Table extension keys
- Improved reference docs

The slide is part of a video presentation. On the right side of the slide, there is a video feed of a speaker, a man with glasses and a beard, wearing a dark suit jacket over a light blue shirt. He is gesturing with his hands while speaking. The background behind him is dark with some light patterns.

Go to Agenda | My Agenda

Customizing reports

Up until now only possible to substitute reports

- Forced to branch and reimplement base app reports
- Forced to branch and merge ISV additions

Major pain

- Costly for simple changes
- Additional maintenance burden for additional to original report object

Gap has been around since the introduction of extensions

Highest voted idea on <https://aka.ms/bcideas> for several releases

376
Vote

Report Extensions should be introduced
on 8/29/2018 1:29:26 PM

02:42 / 1:05:05

The slide is part of a video presentation. On the right side of the slide, there is a video feed of the same speaker as in the previous slide. At the bottom of the slide, there is a social media-style voting widget showing 376 votes and a user profile picture. The video player interface at the bottom shows a play button, a progress bar at 02:42 / 1:05:05, and other controls.


https://events1.social27.com/modyn365blaunchevent2021w1/agenda/player/95494

Finally support for report extensions!

No longer need to branch reports for additive changes
 Add columns, data items, request page extensions and layouts to existing report
 Multiple ISV apps can add to same base report datasets and PTEs uptake in layout
 Also for processing reports

Still branching/copy for destructive changes
 Modifying existing dataitems and columns
 Changing trigger code on existing report

There are now two report customization stories
 Event based substitution of a report without changing any actions in the UI
 Report extension for additive contributions



04:05 / 1:05:05

https://events1.social27.com/modyn365blaunchevent2021w1/agenda/player/95494

Anatomy of a report extension

New top level object
 Requires an ID, reuses report range
 Extends keyword
 New trepertext snippet

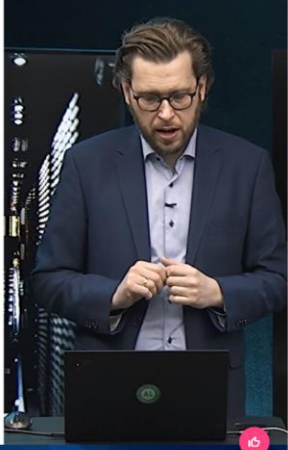
Supported properties
 Only RDCLLayout and WordLayout

Similar top level sections as report
 Dataset for dataitems and columns
 Request page
 Labels (planned for 18.1)
 Triggers
 Procedures
 Variables "global" to report extension

Supported triggers
 On pre and post report run (but not OnInitReport)
 No overwrite, extended report runs first, then by graph

```

reportextension 50130 MyItemssion extends MyTargetReport
{
  RDCLLayout = 'MyItemssion.rdl';
  WordLayout = 'MyItemssion.docx';
  dataset
  {
    // Add changes to datasets and columns here
  }
  requestpage
  {
    // Add changes to the requestpage here
  }
  labels
  {
    myLabel = 'Label!';
  }
  // supported report triggers
  trigger OnPreReport()
  begin
    emit;
  end;
  trigger OnPostReport()
  begin
    emit;
  end;
  // Add procedures
  procedure MyProcedure()
  begin
    emit;
  end;
  // Add local variables
  var
    MyReportExtensionVar: Integer;
}
  
```



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Adding columns to existing dataitems

New add(<existing dataitem>) dataset section

Target existing dataitem in
The extended report
Report extensions in dependency graph

Add columns for
Field in dataitems source table, incl table extensions
Variable in report extension
Protected variable in report graph
Expression
Procedure, local or protected in report graph

Column properties like in normal report dataitems
Except Caption, CaptionML and IncludeCaption > 18.1+

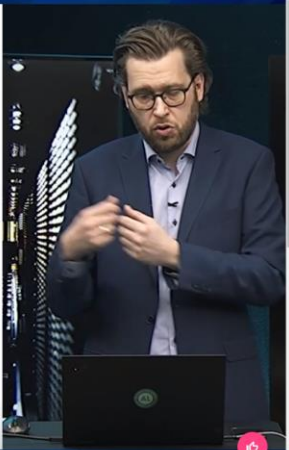
```
reportextension 50110 MyReportExt extends MyReport
{
  dataset
  {
    // Add new column(s)
    // Anchor to any dataitem, including dataitems that
    // come from other report extensions
    add(anchorDataitem)
    {
      // Use any valid expression system for the source
      // For example: a direct field access
      column(columnName; fieldname) {}

      // A global variable in report extension
      column(columnName; 3) {}

      // A global variable in report if protected
      column(columnName; reportvariable) {}

      // An expression to be evaluated
      column(columnName; integerfield + 3) {}

      // A procedure, local or in graph
      column(columnName; myprocedure()) {}
    }
  }
  var
  {
    integer;
  }
}
```



12:03 / 1:05:05

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Modifying existing data items

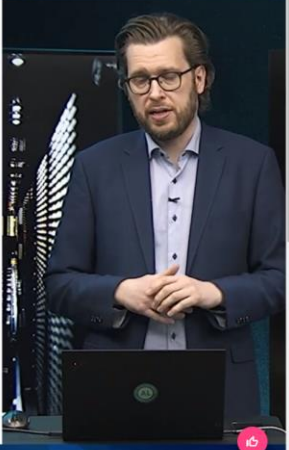
In general not supported through report extensions

Some properties will be supported in 18.1+
RequestFilterFields (additive)
CalcFields (additive)

No plans to support modifying column properties
Add new column, hide from layout and transform value

No plans to support modifying labels

No plans for overwriting existing triggers
Will add more additive triggers in 18.1+



12:03 / 1:05:05

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Add new dataitems to existing dataitems

New sections to add dataitems
 Add nested in target dataitem with addfirst() and addlast()
 Add next to target dataitem with addbefore() and addafter()

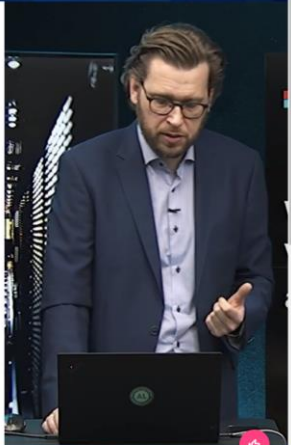
Add to existing dataitems in
 The extended report
 Report extensions in dependency graph

Members similar to normal report dataitems
 Same properties, incl defining table relation
 Same triggers, e.g., OnAfterGetRecord()
 Can add a number of new, nested dataitems

Same column support as in regular dataitems

```

reportextension 50110 MyReportExt extends MyReport
dataset
{
  addfirst(AnchorDataItem)
  {
    dataitem(dataItemName; sourceTable)
    {
      DataItemLink = ID = field(RelatedID);
      // We can nest added dataitems too
      dataitem(NestedDataItem)
      {
        // Like regular nested data items, we can
        // explicitly specify the link
        DataItemLinkReference = topLevelDataItem;
        DataItemLink = ID = field(OtherRelatedID);
        // Columns etc...
      }
    }
  }
}
  
```



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Adding new root dataitems not supported

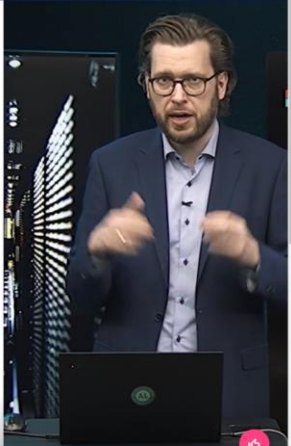
```

AL SalesInvRepExt.al > () ReportExtension 50111 SalesInvRepExt
1
2
3 {
4   dataset
5   {
6     addafter(Header)
7     {
8     }
9   }
10 }
11 }
  
```

TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE

AL SalesInvRepExt.al

Cannot use addbefore or addafter on a top-level data item. The anchor Header is a top level data item. AL[AL0600] [5, 10]



14:49 / 1:05:05

Extending request page

No properties available on requestpage

Layout section similar to page extensions
 addafter(), addbefore(), addfirst(), addlast()
 modify()
 moveafter(), movebefore(), movefirst(), movelast()

Adding fields similar to report request page
 Columns, Variable, Expression, Procedure
 In report extension or protected members in dependency graph

Triggers similar to report request page

```

requestpage
{
  layout
  {
    addfirst(Content)
    {
      group(Options)
      {
        field("Show?"; MyVariable)
        {
          ApplicationArea = All;
          trigger OnValidate()
          begin
          end;
        }
      }
    }
  }
  trigger OnOpenPage()
  begin
  end;
}

```

Request page triggers

Triggers on requestpage node
 Same as page extension triggers

```

OnAfterGetCurrRecord
OnAfterGetRecord
OnClosePage
OnDeleteRecord
OnInsertRecord
OnModifyRecord
OnOpenPage
OnQueryClosePage
OnValidate

```

Triggers on modify request page fields
 Same as page extension modify fields

```

OnAfterAfterLookup
OnAfterValidate
OnAssistEdit
OnBeforeValidate
OnDrillDown
OnLookup
OnValidate

```

Triggers on new requestpage field
 Same as page extension field

```

OnAfterLookup
OnAssistEdit
OnControlAddIn
OnDrillDown
OnLookup
OnValidate

```

Call sequence
 Works like in pages
 "After" call after original field definition triggers
 Others will overwrite

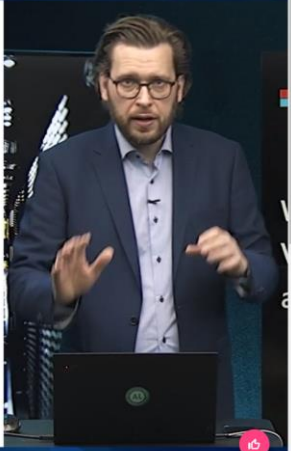
https://events1.social27.com/modyn365betaevent2021w1/agenda/player/95494

Adding layouts

Similar to normal report
Up to one RDLC and/or Word layout
Not required, e.g., column only report extension

Note
No extensibility of layout itself, only substitution
Multiple same-type layouts still not supported
Cannot copy from extended report in VS Code, must export from client

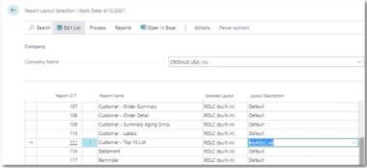
Deployed with app
By default layout will not be "auto-assigned" for report
Automate: Add install code setting up new layouts
Manual: Select in Report Layout Selection page



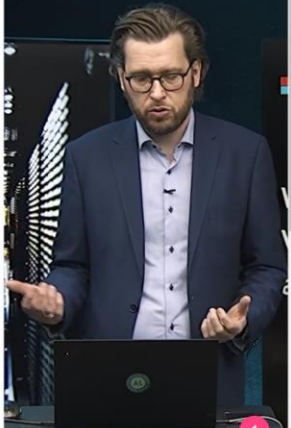
https://events1.social27.com/modyn365betaevent2021w1/agenda/player/95494

Selecting layout in Report Layout Selection page

Select among all installed and custom layouts



Note
Only possible to select layout types supported by base report
Report Layout Selection (and Customer Report Layout) page does not list parent extension for available layouts



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

What happens on uninstall?

Report extension layouts will be
 Hidden on uninstall
 Removed on unpublish

Custom layouts stay
 Runtime error if using removed columns
 Must be fixed manually
 Same as for breaking changes in regular reports

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Business Manager - Dynamics > x

Not secure | vnext.BC/?page=22&tenant=default

Dynamics 365 Business Central

Item Overview

No	Description	Price
1898-S	ATHENS Desk	1,000.80
1900-S	PARIS Guest Chair, black	192.80
1900-S	ATHENS Mobile Pedestal	432.00
1900-S	LONDON Guest Chair, blue	190.10
1925-S	ANTWERP Conference Table	647.60
1925-W	Conference Bundle 1-6	180.80
1928-S	AMSTERDAM Lamp	64.80
1928-W	Conference Bundle 1-8	232.80
1936-S	BERLIN Guest Chair, yellow	192.80
1953-W	Guest Section 1	125.80
1966-S	ROME Guest Chair, green	192.80
1966-S	TOKYO Guest Chair, blue	192.80
1965-W	Conference Bundle 2-8	232.80
1965-S	MILICO Guest Chair, black	190.10
1965-W	Conference Package 1	340.80
1972-S	MOSCOW Guest Chair, yellow	190.10
1980-S	MOSCOW Guest Chair, red	190.10
1980-S	SEOUL Guest Chair, red	192.80
1994-S	ATLANTA Whiteboard, base	1,397.36
2000-S	STONY Guest Chair, green	190.10

https://events1.social27.com/midyn365bclaunchevent2021w1/agenda/player/95494

Not secure | vnext/BC/?tenant=default&page=9652&company=CRONUS%20USA%2C%20Inc.&d...

Dynamics 365 Business Central

Item Overview

No	Description	Country of Origin	Price
1896-S	ATHENS Desk	AT	1,000.80
1900-S	PARIS Guest Chair, black	US	192.80
1906-S	ATHENS Mobile Pedestal	DK	433.60
1908-S	LONDON Swivel Chair, blue	NO	190.10
1920-S	ANTWERP Conference Table	SE	647.80
1925-W	Conference Bundle 1-6	DE	188.80
1928-S	AMSTERDAM Lamp	FR	54.90
1929-W	Conference Bundle 1-8	NL	233.80
1936-S	BERLIN Guest Chair, yellow	CN	192.80
1953-W	Guest Section 1	CY	125.80
1960-S	ROME Guest Chair, green	EL	192.80
1964-S	TOKYO Guest Chair, blue	GB	192.80
1965-W	Conference Bundle 2-8	CA	233.80
1968-S	MEXICO Swivel Chair, black	AU	190.10
1969-W	Conference		341.80
1972-S	MUNICH Swivel Chair, yellow	ES	190.10
1980-S	MOSCOW Swivel Chair, red	IT	190.10
1988-S	STOIL Guest Chair, red	JP	192.80









21:54 / 1:05:05

https://events1.social27.com/midyn365bclaunchevent2021w1/agenda/player/95494

Not secure | vnext/BC/?tenant=default&page=9652&company=CRONUS%20USA%2C%20Inc.&d...

Dynamics 365 Business Central

Item Overview

No	Description	Country of Origin	Barcode	Price
1896-S	ATHENS Desk	AT		1,000.80
1900-S	PARIS Guest Chair, black	US		192.80
1906-S	ATHENS Mobile Pedestal	DK		433.60
1908-S	LONDON Swivel Chair, blue	NO		190.10
1920-S	ANTWERP Conference Table	SE		647.80
1925-W	Conference Bundle 1-6	DE		188.80
1928-S	AMSTERDAM Lamp	FR		54.90
1929-W	Conference Bundle 1-8	NL		233.80

25:49 / 1:05:05

https://events1.social27.com/modyn365betaevent2021w1/agenda/player/95494

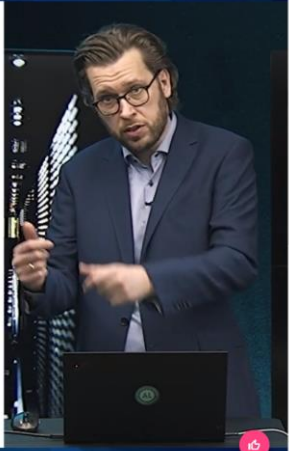
More in LE21-03 - Server and Database

Add barcode to a report

1. Encode string in AL
2. Add encoded string to report dataset
3. Apply barcode font in report layout

Supported format

- Codabar
- Code 39
- Code 128
- Interleaved 2 of 5
- MSI Plessey
- UPC / EAN (EAN-8, EAN-13, UPC-A, UPC-E)
- USPS (IMB/Postnet/Planet)



https://events1.social27.com/modyn365betaevent2021w1/agenda/player/95494

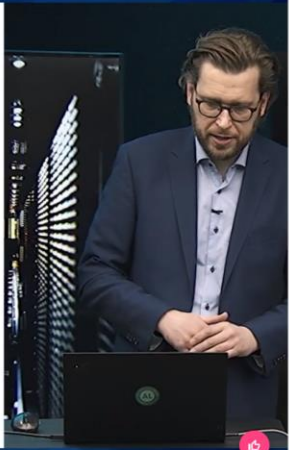
A couple of rough corners at 18.0 release

Major known issues resolved in CU1

- Captions removed on compilation
- addafter/addbefore for dataitems inserting as children

Additional gaps

- Limited modify syntax for dataitem > 18.1+
- Label section support > 18.1+
- Custom layout from report extension layout > 18.1+
- Cannot obsolete report extensions yet
- Txt2AL does not generate report extensions
- Cannot yet insert custom report layout as copy of built-in report extension layout.



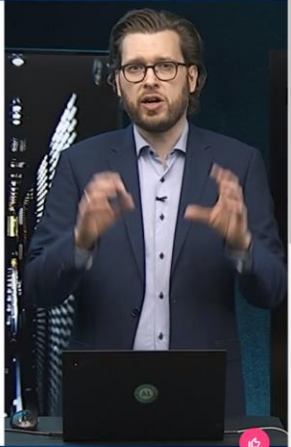
https://events1.social27.com/mmsdyn365bclaunchevent2021w1/agenda/player/95494

Permissions and entitlements: From 2021 Wave 1

Permissions and entitlements are now AL objects
 Deployed as metadata
 Hotfixed as any other app

Benefits of AL source format
 Easier to read (e.g., object names instead of just IDs)
 Supports source control, reviews, merge/diff
 Easy to add comments

Benefits of compiler support
 Intellisense support
 Diagnostics (warnings/errors)
 Search, navigate, and lookup, eg. Ctrl+T and F12
 Directives for conditional compilation



https://events1.social27.com/mmsdyn365bclaunchevent2021w1/agenda/player/95494

Permission object

Defines a named set of object permissions
 Can be public or internal
 Can be assignable to users, or not

Assign permissions for objects
 Directly to object ID
 Using * wildcard (caution: global)
 Nesting other permission sets

Can be extended
 Only additive changes
 Caution: Add to all uses of parent permission set

```

1 permissionset 207 "DMS BASIC"
2
3 Access = Public
4 Assignable = true;
5 Caption = "Dynamics 365 Basic access";
6
7 IncludePermissionSets = "BaseApp Objects - Exec",
8 "System App - Basic",
9 "Session - EXEC";
10
11 Permissions = tabledata "AddIn" = Imd,
12 tabledata "Aggregate Permission Set" = Rmd,
13 tabledata "AllDoc" = Imd,
14 tabledata "AllDoc/OutCaption" = Rmd,
15 tabledata "Chart" = Imd,
16 tabledata "Case Coverage" = Rmd,
17 tabledata "CodeUnit Metadata" = Imd,
18 tabledata "Data Sensitivity" = R,
19 tabledata "Date" = Imd,
20 tabledata "Designed Query" = R,
21 tabledata "Designed Query Caption" = R,
22 tabledata "Designed Query Category" = R,
23 tabledata "Designed Query Column" = R,

```

```

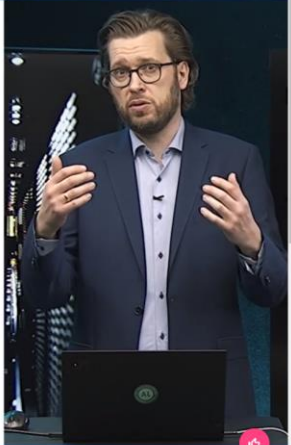
1 permissionset 4111 "SHP Connector"
2
3 Assignable = false;
4 Access = Internal;
5 Permissions = submittit * = R,
6 page * = R,
7 table * = R,
8 tabledata "SHP Account" = rmd;

```

```

1 permissionsetextension 4512 "Email - Admin - SHP" extends "Email - Admin"
2
3 Permissions = tabledata "SHP Account" = rmd;
4
5

```



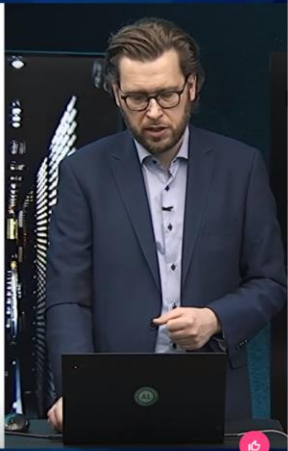
https://events1.social27.com/msdyn365blaunchevent2021w1/agenda/player/95494

Entitlement object

Defines the objects in a license
Links to license type and id
Links to permission sets

Base app only for now

```
entitlement "Dynamics 365 Business Central Essentials BaseApp"
{
  Type = PerUserServicePlan;
  Id = '928656a2-7dd8-4c83-97b6-a356414dbd36';
  ObjectEntitlements = "BaseApp Objects - Exec",
    "D365 BASIC",
    "D365 BUS FULL ACCESS",
    "D365 DIM CORRECTION",
    "D365 DIM MONITOR FIELDS",
    "Reten. Pol. Setup - BaseApp",
    "Security - BaseApp";
}
```



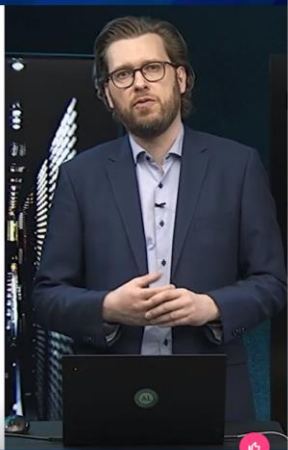
https://events1.social27.com/msdyn365blaunchevent2021w1/agenda/player/95494

Interested in more?

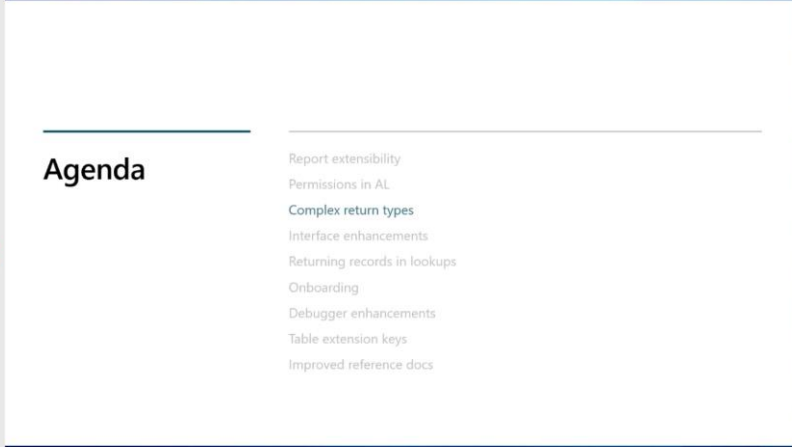
Join session!
LE21-09 Permissions and entitlements

Covers new permissions and entitlements in detail
Opt-in server setting onprem
Conversion tools between xml and AL objects for compare and merge

32:16 / 1:05:05




https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494



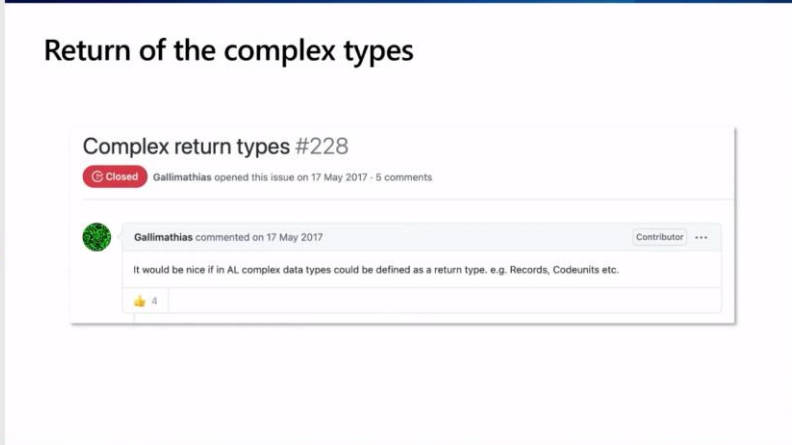
Agenda

- Report extensibility
- Permissions in AL
- Complex return types**
- Interface enhancements
- Returning records in lookups
- Onboarding
- Debugger enhancements
- Table extension keys
- Improved reference docs



A man with glasses and a beard, wearing a dark suit jacket over a light blue shirt, stands behind a laptop. He is looking towards the left of the frame. The background is dark with some faint lights.


https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494



Return of the complex types

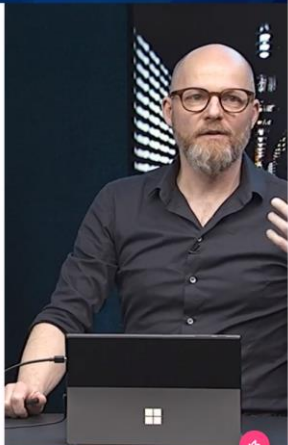
Complex return types #228

Closed Gallimathias opened this issue on 17 May 2017 - 5 comments

 Gallimathias commented on 17 May 2017 Contributor ...

It would be nice if in AL complex data types could be defined as a return type, e.g. Records, Codeunits etc.

👍 4




A man with glasses and a beard, wearing a dark button-down shirt, stands behind a laptop. He is looking towards the camera. The background is dark with some faint lights.

https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494

Simple types

```
procedure GetNumber(): Integer;
begin
  exit(42);
end;
```




A video feed of a speaker, a man with glasses and a beard, wearing a dark shirt, sitting at a desk with a laptop. The video feed is positioned on the right side of the slide.

https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494

Syntax

```
procedure GetCodeunit(): codeunit MyCodeunit;
begin
  // return codeunit with state
end;
```




A video feed of the same speaker as in the first slide, sitting at a desk with a laptop. The video feed is positioned on the right side of the slide.

https://events1.social27.com/modyn365blaunchevent2021w1/agenda/player/95494

Returning a record

```
procedure GetCustomerByName(Name: Text): record Customer;
var
  Customer: record Customer;
begin
  Customer.SetFilter("Search Name", '%1', '@*' + Name + '**');
  exit(Customer);
end;
```




The speaker is a man with a beard and glasses, wearing a dark blue shirt, sitting at a desk with a laptop. He is gesturing with his right hand while speaking.

https://events1.social27.com/modyn365blaunchevent2021w1/agenda/player/95494

Named return values

```
procedure GetCustomerByName(Name: Text) Customer: record Customer;
begin
  Customer.SetFilter("Search Name", '%1', '@*' + Name + '**');
end;
```



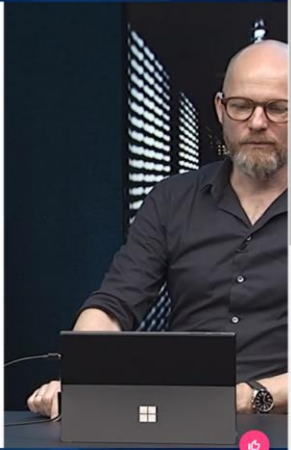
The speaker is the same man as in the first image, sitting at a desk with a laptop. He is looking towards the camera.

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Default return values

```
procedure GetCustomerByName(Name: Text): record Customer;
begin
end;

procedure GetCustomerByName(Name: Text) Customer: record Customer;
begin
end;
```



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Using the returned values

```
procedure UseIt()
var
    Credit: Decimal;
begin
    Credit := GetCustomerByName('SPOT').CalcAvailableCredit();
end;

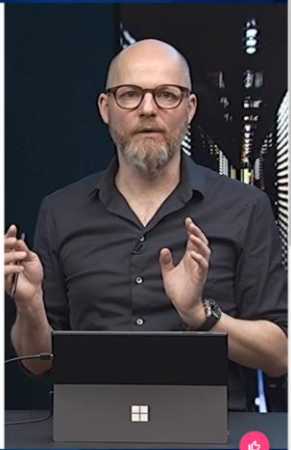
procedure GetCustomerByName(Name: Text): record Customer;
begin
    // ...
end;
```


https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Home Settings Search Creators Networking My Profile My Agenda Satisfaction Survey Notifications Logout

Get more!

```
procedure GetMore(): record OfEverything;
```



?

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Home Settings Search Creators Networking My Profile My Agenda Satisfaction Survey Notifications Logout

Agenda

- Report extensibility
- Permissions in AL
- Complex return types
- Interface enhancements**
- Returning records in lookups
- Onboarding
- Debugger enhancements
- Table extension keys
- Improved reference docs

?

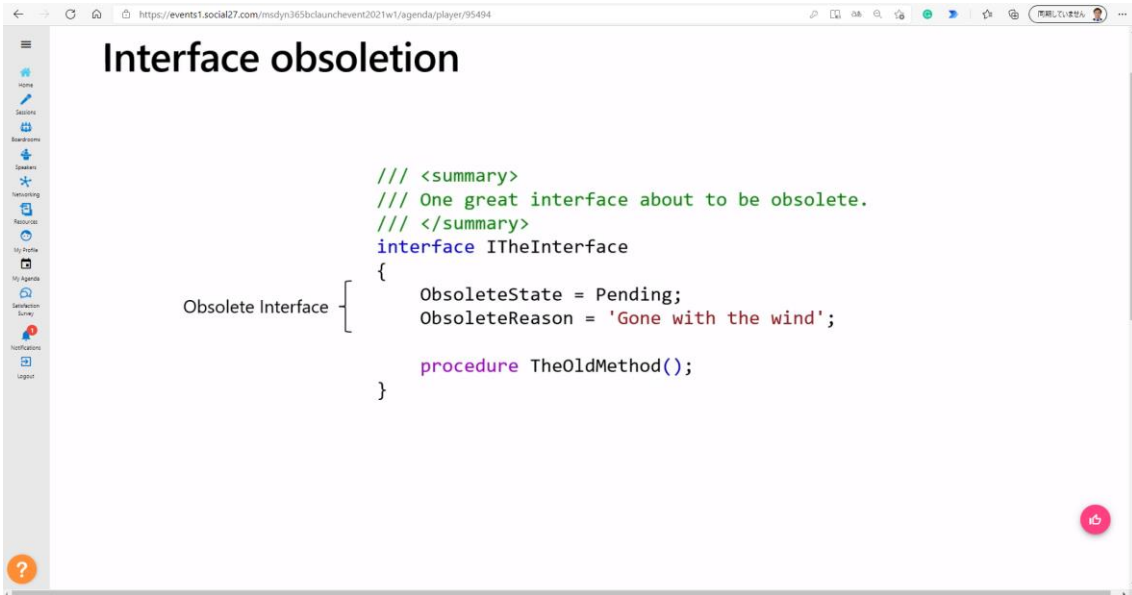
https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Interface obsoletion

```
/// <summary>
/// One great interface about to be obsolete.
/// </summary>
interface ITheInterface
{
    ObsoleteState = Pending;
    ObsoleteReason = 'Gone with the wind';

    procedure TheOldMethod();
}
```

Obsolete Interface {



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

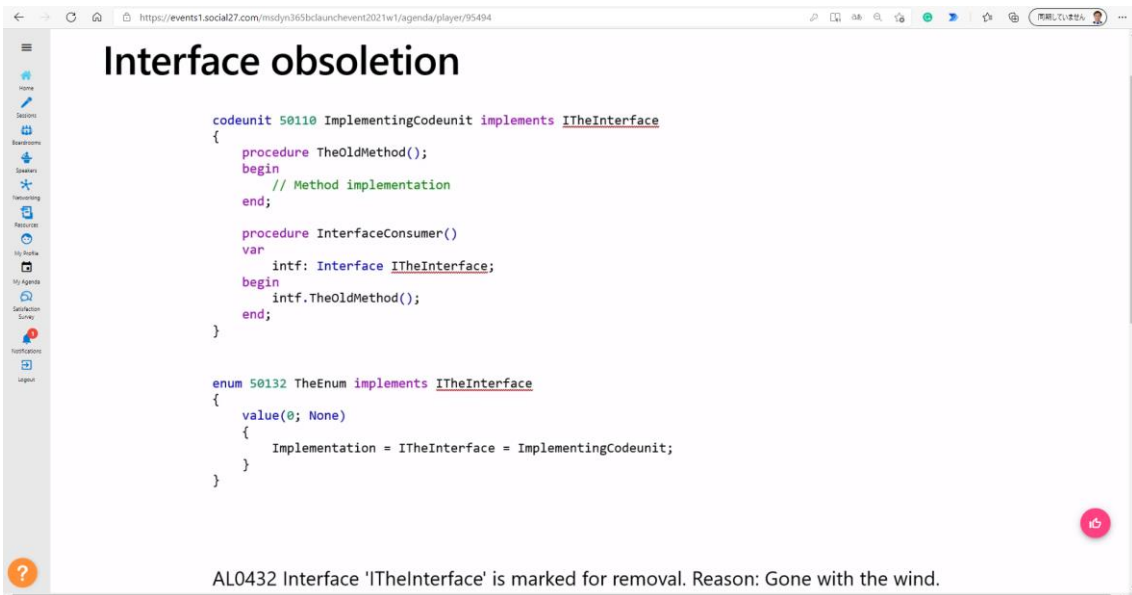
Interface obsoletion

```
codeunit 50110 ImplementingCodeunit implements ITheInterface
{
    procedure TheOldMethod();
    begin
        // Method implementation
    end;

    procedure InterfaceConsumer()
    var
        intf: Interface ITheInterface;
    begin
        intf.TheOldMethod();
    end;
}

enum 50132 TheEnum implements ITheInterface
{
    value(0; None)
    {
        Implementation = ITheInterface = ImplementingCodeunit;
    }
}
```

AL0432 Interface 'ITheInterface' is marked for removal. Reason: Gone with the wind.




https://events1.social427.com/midyn365bclaunchevent2021w1/agenda/player/95494

Interface method obsoletion

```
/// <summary>
/// One great interface about to be obsolete.
/// </summary>
interface ITheInterface
{
    [Obsolete('Too old')]
    procedure TheOldMethod();
}
```

Obsolete Method {



A man with glasses and a beard, wearing a dark shirt, is speaking in a video call. He is sitting at a desk with a laptop in front of him. The background shows a city skyline at night.

https://events1.social427.com/midyn365bclaunchevent2021w1/agenda/player/95494


Interface method obsoletion

```
codeunit 50130 ConsumingCodeunit
{
    procedure InterfaceConsumer()
    var
        intf: Interface ITheInterface;
    begin
        intf.TheOldMethod();
    end;
}

codeunit 50110 ImplementingCodeunit implements ITheInterface
{
    procedure TheOldMethod();
    begin
        // Method implementation
    end;
}
```

AL0432 Method 'TheOldMethod' is marked for removal. Reason: Too old.

40:07 / 1:05:06



A man with glasses and a beard, wearing a dark shirt, is speaking in a video call. He is sitting at a desk with a laptop in front of him. The background shows a city skyline at night.

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Internal Interfaces

```
/// <summary>
/// The Internal interface
/// </summary>
interface ITheInterface
{
    Access = Internal;

    procedure TheMethod();
}
```

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

UnknownValueImplementation property

41:11 / 1:05:05


The second screenshot shows a presentation slide titled "UnknownValueImplementation property". The slide is mostly blank with the title text. On the right side, there is a video inset showing a man with glasses and a beard, wearing a dark shirt, speaking. At the bottom of the slide, there is a video player control bar showing a pause icon, a progress bar at 41:11 / 1:05:05, and other standard video controls. The left sidebar of the application is visible in both screenshots.

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Loyalty benefits sample

```
interface ILoyaltyBenefits
{
    // Declaration
    procedure HasBenefit(benefit: Text): Boolean;
}

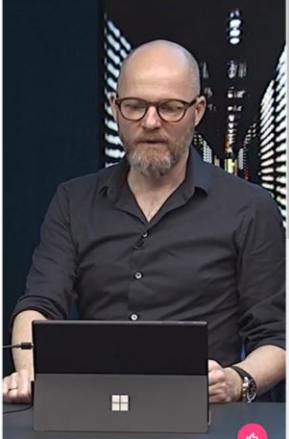
codeunit 50140 NoBenefits implements ILoyaltyBenefits
{
    // Implementation
    procedure HasBenefit(benefit: Text): Boolean;
    begin
        exit(false);
    end;
}
```



Dynamics 365 Business Ce... x +
https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494


DefaultImplementation property

```
enum 50140 LoyaltyLevel implements ILoyaltyBenefits
{
    Extensible = true;
    DefaultImplementation = ILoyaltyBenefits = NoBenefits;
    value(0; Bronze)
    {
        // Implementation = ILoyaltyBenefits = NoBenefits;
    }
}
```



https://events1.social27.com/mofyn365bclaunchevent2021w1/agenda/player/95494


```
var
  LoyaltyBenefits: interface ILoyaltyBenefits;
  Customer: record Customer;
begin
  LoyaltyBenefits := Customer.Loyalty;
  if LoyaltyBenefits.HasBenefit('FreeShipping') then begin
    // Ship for free
  end;
end;
```



https://events1.social27.com/mofyn365bclaunchevent2021w1/agenda/player/95494

```
enumextension 50140 SilverLoyaltyLevel extends LoyaltyLevel
{
  value(1; Silver)
  {
    Implementation = ILoyaltyBenefits = SilverBenefits;
  }
}
```

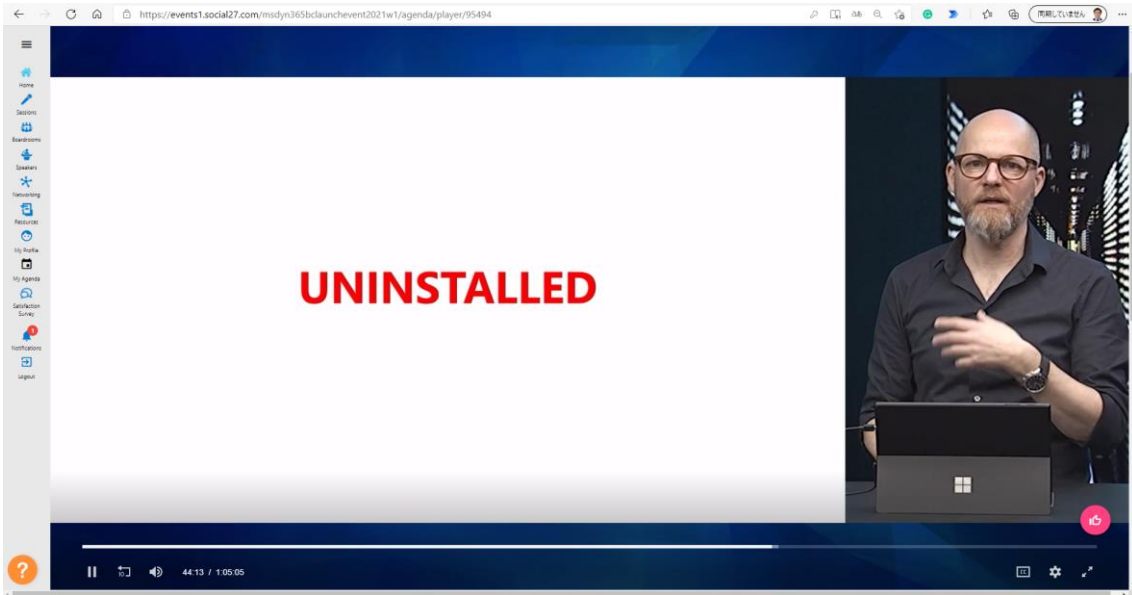
43:51 / 1:05:05



https://events1.social27.com/modyn365bxlaunchevent2021w1/agenda/player/95494

UNINSTALLED

44:13 / 1:05:05

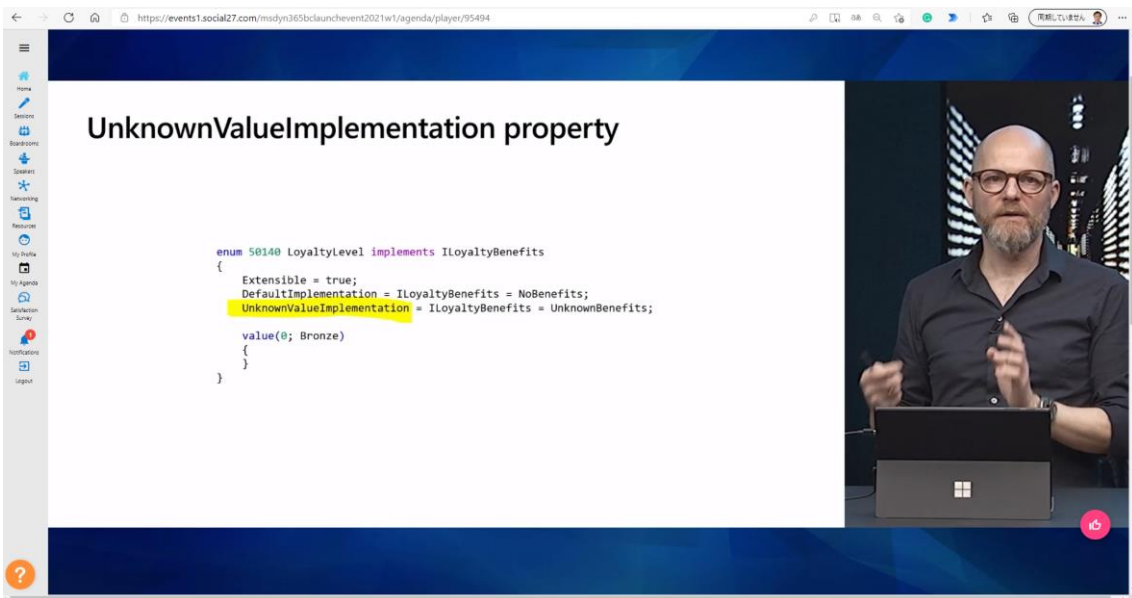


https://events1.social27.com/modyn365bxlaunchevent2021w1/agenda/player/95494

UnknownValueImplementation property

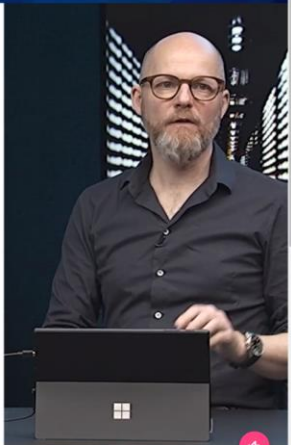
```
enum 50140 LoyaltyLevel implements ILoyaltyBenefits
{
    Extensible = true;
    DefaultImplementation = ILoyaltyBenefits = NoBenefits;
    UnknownValueImplementation = ILoyaltyBenefits = UnknownBenefits;

    value(0; Bronze)
    {
    }
}
```



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

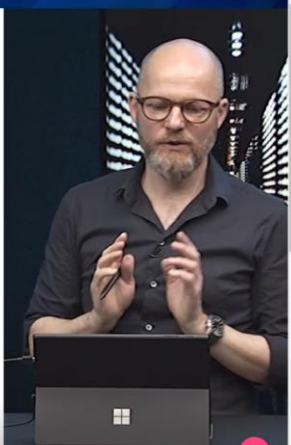
Property	When to use
DefaultImplementation	Use to prevent downstream breakage in extensions when adding interfaces to enums
UnknownValueImplementation	Use to provide interface implementations for values that are removed (unknown) the enum



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Agenda

- Report extensibility
- Permissions in AL
- Complex return types
- Interface enhancements
- Returning records in lookups
- Onboarding
- Debugger enhancements
- Table extension keys
- Improved reference docs



45:51 / 1:05:05

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Selecting an item for a sales line

```
graph LR; SalesLine[Sales Line] --- Item[Item]; SalesLine --> Item;
```

The diagram shows two tables: Sales Line and Item. Sales Line has a primary key (PK) and a foreign key (FK) named ItemNo. Item has a primary key (PK) and a field named Description. Lines connect the PK of Sales Line to the PK of Item, and the FK ItemNo of Sales Line to the PK of Item, indicating a one-to-many relationship.

46:02 / 1:05:05

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

In AL

```
table 37 SalesLine
{
    fields
    {
        field(1; PK; Code[20]) { }
        field(2; Item; Code[20])
        {
            TableRelation = Item;
        }
        field(3; ItemDescription; Text[100])
        {
            TableRelation = Item.Description;
        }
    }
}
```

```
table 27 Item
{
    fields
    {
        field(1; No; Code[20]) { }
        field(3; Description; Text[100])
        { }
    }
}
```

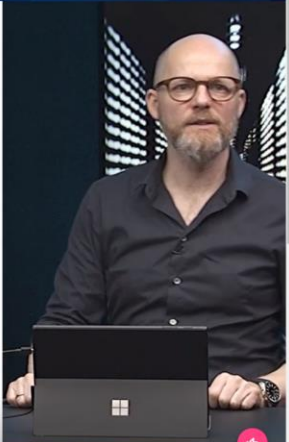
The image shows the AL code for the SalesLine and Item tables. The SalesLine table has three fields: a primary key Code[20], a Code[20] field with a TableRelation to Item, and a Text[100] field with a TableRelation to Item.Description. The Item table has two fields: a Code[20] field and a Text[100] field.

https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494

Triggers

```
table 37 SalesLine
{
    fields
    {
        field(1; PK; Code[20]) { }
        field(2; Item; Code[20])
        {
            TableRelation = Item;

            trigger OnValidate()
            var
                Item: record Item;
            begin
                Item."No." := Rec.Item;
                Item.FindFirst();
                Rec.ItemDescription := Item.Description;
            end;
        }
        field(3; ItemDescription; Text[100])
        {
            TableRelation = Item.Description;
            trigger OnValidate()
            var
                Item: record Item;
            begin
                Item.SetRange(Description, Rec.ItemDescription);
                Item.FindFirst();
                Rec.Item := Item."No.";
            end;
        }
    }
}
```



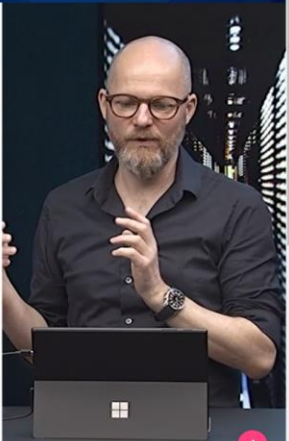
46:49 / 1:05:05

https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494

Triggers

```
table 37 SalesLine
{
    fields
    {
        field(1; PK; Code[20]) { }
        field(2; Item; Code[20])
        {
            TableRelation = Item;

            trigger OnValidate()
            var
                Item: record Item;
            begin
                Item."No." := Rec.Item;
                Item.FindFirst();
                Rec.ItemDescription := Item.Description;
            end;
        }
        field(3; ItemDescription; Text[100])
        {
            TableRelation = Item.Description;
            trigger OnValidate()
            var
                Item: record Item;
            begin
                Item.SetRange(Description, Rec.ItemDescription);
                Item.FindFirst();
                Rec.Item := Item."No.";
            end;
        }
    }
}
```




46:49 / 1:05:05

https://events1.social27.com/midyn365bclaunchevent2021w1/agenda/player/95494

Triggers

```
trigger OnValidate()
var
    Item: record Item;
begin
    Item.SetRange(Description, Rec.ItemDescription);
    Item.FindFirst();
    Rec.Item := Item."No.";
end;
```



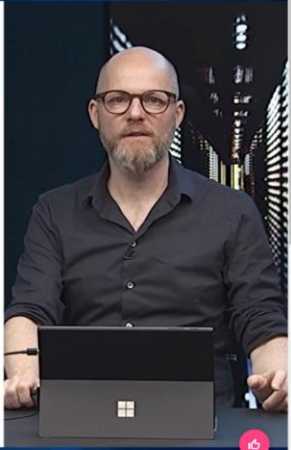
https://events1.social27.com/midyn365bclaunchevent2021w1/agenda/player/95494

Problem!

ItemDescription Bicyc

No. ↑	Description	Base Unit of Measure	Unit Price
1000	Bicycle	PCS	4,000.00
1001	Touring Bicycle	PCS	4,000.00
70061	Bicycle	PCS	0.00

+ New Select from full list



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Problem!

```
field(ItemDescription; Rec.ItemDescription)
{
    TableRelation = Item.Description;

    trigger OnValidate()
    var
        Item: record Item;
    begin
        Item.SetRange(Description, Rec.ItemDescription);
        Item.FindFirst();
        Rec.Item := Item."No.";
    end;
end;
```

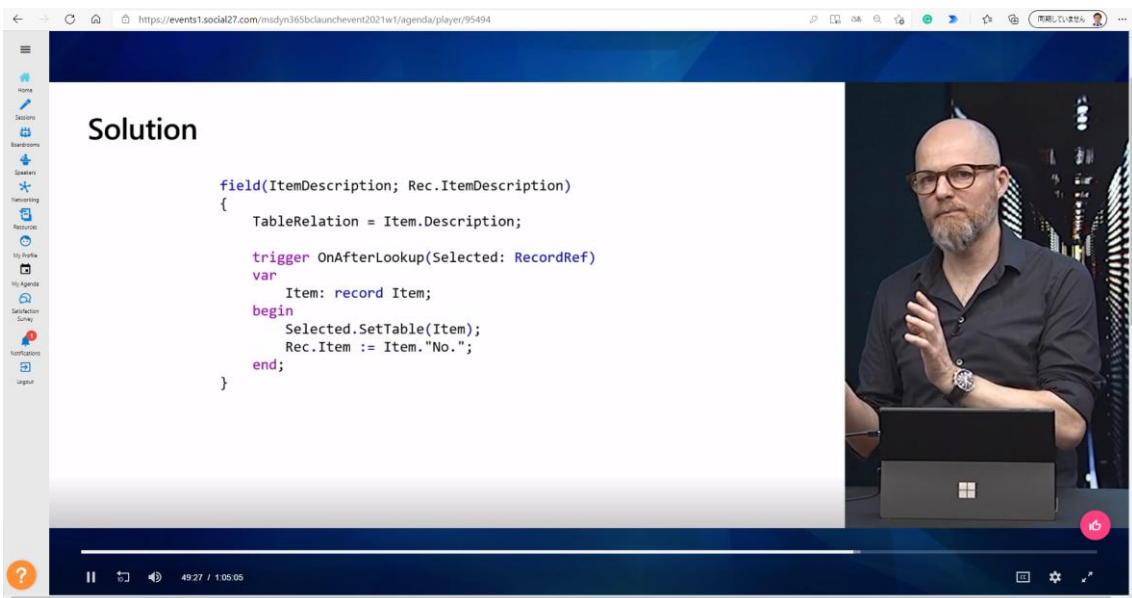


https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Solution

```
field(ItemDescription; Rec.ItemDescription)
{
    TableRelation = Item.Description;

    trigger OnAfterLookup(Selected: RecordRef)
    var
        Item: record Item;
    begin
        Selected.SetTable(Item);
        Rec.Item := Item."No.";
    end;
end;
```



49:27 / 1:05:05

RecordRef

```
field(50339; Type; Option)
{
  OptionMembers = Item,Resource;
}
field(50340; No; Code[20])
{
  TableRelation =
  {
    if (Type = const(Item)) Item
    else if (Type = const(Resource)) Resource;
  }
}
field(50341; Description; Text[100])
{
  TableRelation =
  {
    if (Type = const(Item)) Item.Description
    else if (Type = const(Resource)) Resource.Name;
    ValidateTableRelation = false;
  }
}

field(No; Rec.No)
{
  trigger OnAfterLookup(Selected: RecordRef)
  var
    Item: record Item;
    Resource: record Resource;
  begin
    case Selected.Number of
      Database::Item:
      begin
        Selected.SetTable(Item);
        Rec.Description := Item.Description;
      end;
      Database::Resource:
      begin
        Selected.SetTable(Resource);
        Rec.Description := Resource.Name;
      end;
    end;
  end;
}
```

Agenda

- Report extensibility
- Permissions in AL
- Complex return types
- Interface enhancements
- Returning records in lookups
- Onboarding**
- Debugger enhancements
- Table extension keys
- Improved reference docs

https://events1.social27.com/modyn365blaunchevent2021w1/agenda/player/95494

Defining teaching tips

AboutTitle and AboutText properties
On page object and its fields, actions, and parts
Multi-language versions available

No control of teaching tip sequence
Tours handled by client rendering

```

1 page 000 "Table Invoice List"
2
3 ApplicationName = "Modin", SubSite
4 ClientName = "Modin Development"
5 ClientLogo = "Modin Development"
6 ClientLogoColor = "#000000" (Color Customer No. 1)
7 ClientLogo = "#000000"
8 PageTitle = "List"
9
10 ApplicationPageName = "You_Processes,Report,Deliveries,Posting,Invoice,Account,Approval,Design"
11
12 ApplicationName = "Modin", SubSite (List)
13 ApplicationLogo = "Modin Development"
14 ApplicationLogoColor = "#000000" (Color Customer No. 1)
15 ApplicationLogo = "#000000"
16 ApplicationPageName = "You_Processes,Report,Deliveries,Posting,Invoice,Account,Approval,Design"
17 ApplicationPageTitle = "Table Invoice List"
18 ApplicationPageTitle = "Table Invoice List"
19 ApplicationPageTitle = "Table Invoice List"
20 ApplicationPageTitle = "Table Invoice List"
21 ApplicationPageTitle = "Table Invoice List"
22 ApplicationPageTitle = "Table Invoice List"
23 ApplicationPageTitle = "Table Invoice List"
24 ApplicationPageTitle = "Table Invoice List"
25 ApplicationPageTitle = "Table Invoice List"
26 ApplicationPageTitle = "Table Invoice List"
27 ApplicationPageTitle = "Table Invoice List"
28 ApplicationPageTitle = "Table Invoice List"
29 ApplicationPageTitle = "Table Invoice List"
30 ApplicationPageTitle = "Table Invoice List"
31 ApplicationPageTitle = "Table Invoice List"
32 ApplicationPageTitle = "Table Invoice List"
33 ApplicationPageTitle = "Table Invoice List"
34 ApplicationPageTitle = "Table Invoice List"
35 ApplicationPageTitle = "Table Invoice List"
36 ApplicationPageTitle = "Table Invoice List"
37 ApplicationPageTitle = "Table Invoice List"
38 ApplicationPageTitle = "Table Invoice List"
39 ApplicationPageTitle = "Table Invoice List"
40 ApplicationPageTitle = "Table Invoice List"
41 ApplicationPageTitle = "Table Invoice List"
42 ApplicationPageTitle = "Table Invoice List"
43 ApplicationPageTitle = "Table Invoice List"
44 ApplicationPageTitle = "Table Invoice List"
45 ApplicationPageTitle = "Table Invoice List"
46 ApplicationPageTitle = "Table Invoice List"
47 ApplicationPageTitle = "Table Invoice List"
48 ApplicationPageTitle = "Table Invoice List"
49 ApplicationPageTitle = "Table Invoice List"
50 ApplicationPageTitle = "Table Invoice List"
51 ApplicationPageTitle = "Table Invoice List"
52 ApplicationPageTitle = "Table Invoice List"
53 ApplicationPageTitle = "Table Invoice List"
54 ApplicationPageTitle = "Table Invoice List"
55 ApplicationPageTitle = "Table Invoice List"
56 ApplicationPageTitle = "Table Invoice List"
57 ApplicationPageTitle = "Table Invoice List"
58 ApplicationPageTitle = "Table Invoice List"
59 ApplicationPageTitle = "Table Invoice List"
60 ApplicationPageTitle = "Table Invoice List"
61 ApplicationPageTitle = "Table Invoice List"
62 ApplicationPageTitle = "Table Invoice List"
63 ApplicationPageTitle = "Table Invoice List"
64 ApplicationPageTitle = "Table Invoice List"
65 ApplicationPageTitle = "Table Invoice List"
66 ApplicationPageTitle = "Table Invoice List"
67 ApplicationPageTitle = "Table Invoice List"
68 ApplicationPageTitle = "Table Invoice List"
69 ApplicationPageTitle = "Table Invoice List"
70 ApplicationPageTitle = "Table Invoice List"
71 ApplicationPageTitle = "Table Invoice List"
72 ApplicationPageTitle = "Table Invoice List"
73 ApplicationPageTitle = "Table Invoice List"
74 ApplicationPageTitle = "Table Invoice List"
75 ApplicationPageTitle = "Table Invoice List"
76 ApplicationPageTitle = "Table Invoice List"
77 ApplicationPageTitle = "Table Invoice List"
78 ApplicationPageTitle = "Table Invoice List"
79 ApplicationPageTitle = "Table Invoice List"
80 ApplicationPageTitle = "Table Invoice List"
81 ApplicationPageTitle = "Table Invoice List"
82 ApplicationPageTitle = "Table Invoice List"
83 ApplicationPageTitle = "Table Invoice List"
84 ApplicationPageTitle = "Table Invoice List"
85 ApplicationPageTitle = "Table Invoice List"
86 ApplicationPageTitle = "Table Invoice List"
87 ApplicationPageTitle = "Table Invoice List"
88 ApplicationPageTitle = "Table Invoice List"
89 ApplicationPageTitle = "Table Invoice List"
90 ApplicationPageTitle = "Table Invoice List"
91 ApplicationPageTitle = "Table Invoice List"
92 ApplicationPageTitle = "Table Invoice List"
93 ApplicationPageTitle = "Table Invoice List"
94 ApplicationPageTitle = "Table Invoice List"
95 ApplicationPageTitle = "Table Invoice List"
96 ApplicationPageTitle = "Table Invoice List"
97 ApplicationPageTitle = "Table Invoice List"
98 ApplicationPageTitle = "Table Invoice List"
99 ApplicationPageTitle = "Table Invoice List"
100 ApplicationPageTitle = "Table Invoice List"

```

Post an invoice

You post an invoice when you've completed all its details. This action posts the selected invoice, and you can later send it to the customer if needed.

2 of 3 Previous Next

© 2021 Modin Corporation A/S

53:06 / 1:05:05

https://events1.social27.com/modyn365blaunchevent2021w1/agenda/player/95494

Extending teaching tips

Just like normal page extensions
Add AboutTitle and AboutText to new page extension controls

Modify existing page controls
Overwrite with new title and/or description
Suppress by overwriting with empty value (title)

53:27 / 1:05:05

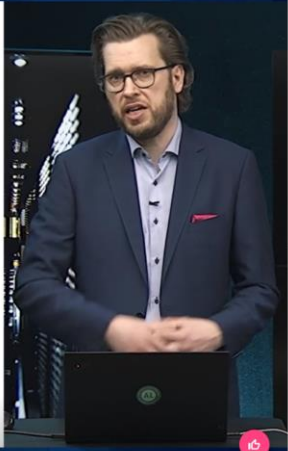
https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494

Home
Sessions
Basics
Speakers
Networking
Workshops
My Profile
My Agenda
Satisfaction Survey
Notifications
Logout

Interested in more?

Watch onboarding session
LE21-15 Simplifying customer onboarding

Also covers Getting Started checklist and much more



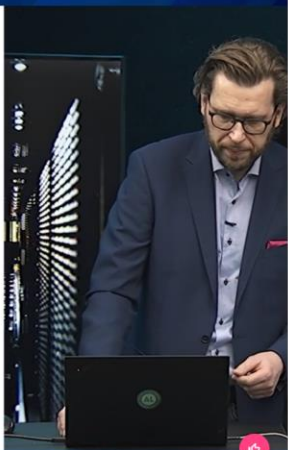
53:58 / 1:05:05

https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494

Home
Sessions
Basics
Speakers
Networking
Workshops
My Profile
My Agenda
Satisfaction Survey
Notifications
Logout

Agenda

- Report extensibility
- Permissions in AL
- Complex return types
- Interface enhancements
- Returning records in lookups
- Onboarding
- Debugger enhancements
- Table extension keys
- Improved reference docs




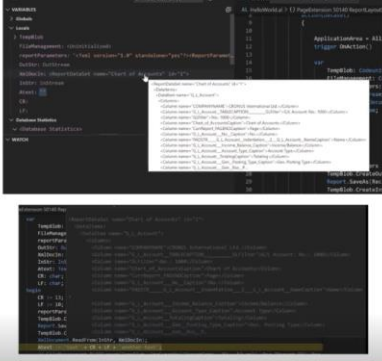
https://events1.social27.com/modyn365blaunchevent2021w1/agenda/player/95494

Inspecting large text variables during debug

Variables pane challenge
Only first line of variables shown
Tooltip is truncated

Source view challenge
Tooltip is truncated

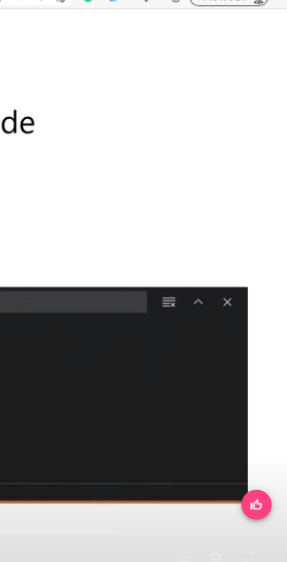
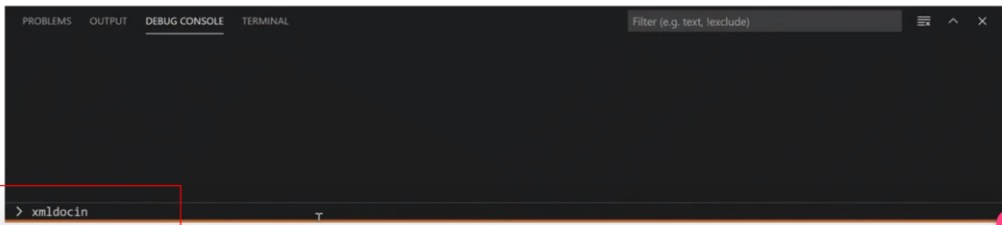
Both are VS Code limitations



https://events1.social27.com/modyn365blaunchevent2021w1/agenda/player/95494

We now support the Debug Console

Standard way to show large variable content in VS Code
Supports multiline variables and will not truncate
Just type in variable name and hit return
Also supports our SQL database statistics



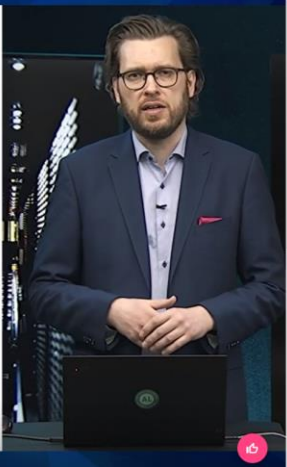
https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

We now support the Debug Console

Standard way to show large variable content in VS Code

- Supports multiline variables and will not truncate
- Just type in variable name and hit return
- Also supports our SQL database statistics

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Filter (e.g. text, exclude)
<Column name="ConsTransMethod">Average Rate (Manual)</Column>
<Column name="IncomeBalance">Balance Sheet</Column>
<Column name="GenPostingType"></Column>
</Columns>
</DataItems>
</DataItems>
</DataItems>
</ReportDataSet>
```



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Snapshot debugger enhancements

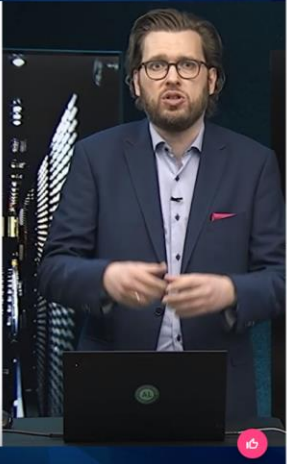
Now enabled for production environments (>v17.3)

Support for built-in system codeunits (from v18.1)

- Capture stack traces originating from system codeunit ids > 2 billion
- E.g., [triggers for system objects](#) like reporting, install, upgrade, system actions etc

Likely supported soon (18.2)

- Ability to hover on variables in source view during snapshot playback
- Ability to see SQL Database statistics at snappoints



Snapshots in code without source files?

You can investigate code and set snappoints in symbol AL (.dal) files
 Use snapshot launch configuration to download symbols directly from prod
 Only for extensions with ShowMyCode=true

How to find and navigate to symbol code to set snappoints?
 Use event viewer to record objects throwing in repro steps
 Use a small HelloWorld sample to navigate to .dal files with Go to Definition (F12) and continue from there

```

AL HelloWorld
1 pageextension 50100 CustListExt extends "Customer List"
2 {
3   trigger OnOpenPage()
4   var
5     ODataUtility: Codeunit ODataUtility;
6   begin
7     Message('App published: Hello world!');
8   end;
9 }
10 }

AL Customer List
1 page 22 "Customer List"
2 {
3   ApplicationArea = Basic, Suite, Service;
4   Caption = 'Customers';
5   CardPageID = "Customer Card";
6   Editable = false;
7   PageType = List;
8   PromotedCategories = "New, Process, Report, Service";
9   QueryCategory = "Customer List";
10  RefreshOnAction = true;
11  SourcePage = Customer;
12  UsageCategory = "L100";
13 }
14 layout
15 {
16   area(content)
17   {
18     repeater(Controls)
19     {
20       ShowCaption = false;
21       field("No.", "No.")
22       {
23         ApplicationArea = All;
  
```

```

AL Customer
1 table 38 Customer
2 {
3   Caption = 'Customer';
4   DataCaptionFields = "No.", Name;
5   DrillDownPageID = "Customer List";
6   LookupPageID = "Customer Lookup";
7 }
8 Permissions = TableData "Cust. Ledger Entry" = r,
9             TableData Job = r,
10            TableData "VAT Registration Log" = rd,
11            TableData "Service Header" = r,
12            TableData "Service Ledger Entry" = r,
13            TableData "Service Item" = rm,
14            TableData "Service Contract Header" = rm,
15            TableData "Price List Header" = rd,
16            TableData "Price List Line" = rd,
17            TableData "Sales Price Access" = rd,
18            TableData "Sales Discount Access" = rd,
19            TableData "Sales Price" = rd,
20            TableData "Sales Line Discount" = rd;
21 }
22 }
23 }
  
```

Agenda

- Report extensibility
- Permissions in AL
- Complex return types
- Interface enhancements
- Returning records in lookups
- Onboarding
- Debugger enhancements
- Extending table keys
- Improved reference docs

https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494

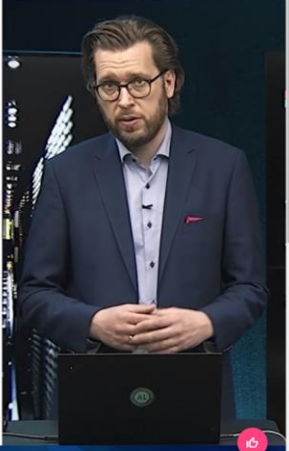
Extending table keys

Add key to base table from table extension

All fields must be in base table
Keys cannot span across base table and table extensions

For more, see
LE21-03 What's new: Server and database

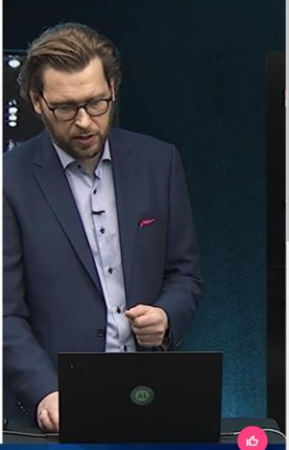
```
tableextension 50100 MyExtension extends Customer
{
  keys
  {
    key(MyCustomKey; City, "Country/Region Code") { }
  }
}
```



https://events1.social27.com/modyn365bclaunchevent2021w1/agenda/player/95494

Agenda

- Report Extensibility
- Permissions in AL
- Complex return types
- Interface enhancements
- Returning records in lookups
- Onboarding
- Debugger enhancements
- Table extension keys
- Improved reference docs



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Autogenerated docs for public APIs

companyInformation resource type

Important: Currently all requests to API v2.0 must append \$schema=version2.0 to the request URL to work properly. This is to replace the new improved \$metadata model available from Business Central 17. An update will be applied shortly to auto-generate the information automatically to API v2.0 for all data API. Business version 2 will be the default unless specified in the request URL with \$schema=version2.0. A notice is being worked on where the connection will be default to the new version.

Requires a company information in Dynamics 365 Business Central.

Methods

Method	Return Type	Description
GetCompanyInformation	companyInformation	Gets a company information object.
UpdateCompanyInformation	companyInformation	Updates a company information object.

Navigation

Navigation	Return Type	Description
Navigation	navigation	Gets the navigation of the company information.

Properties

Property	Type	Description
Id	Guid	The unique ID of the company information. Non-nullable.
DisplayName	string	Specify the company information name. This name will appear on all site documents for the company information.

https://docs.microsoft.com/en-us/dynamics-nav/api-reference/v2.0/resources/dynamics_companyinformation

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Methods added to TOC: Search & breadcrumbs

addtext

Development > AL Programming > AddText(String [, Integer]) Method

Development > AL Programming > AddText(BigText [, Integer]) Method

Search for "addtext" in all BusinessCentral documentation

Instrumenting for Telemetry

Exporting Permission Sets

.NET in AL

Testing the Application

Dynamics 365 / Business Central / Developer and IT-pro / Development / AL Programming / Methods / BigText Data Type / AddText(String [, Integer]) Method

Filter by title

AddText(String [, Integer]) Method

AddText(BigText [, Integer]) Method

GetSubText(Var Text, Integer [, Integer]) Method

GetSubText(Var BigText, Integer [, Integer]) Method

Length() Method

ReadInStream() Method

TextPos(String) Method

Write(OutputStream) Method

Blob Data Type

Boolean Data Type

Byte Data Type

Char Data Type

Code Data Type

Codeunit Data Type

CompanyProperty Data Type

Database Data Type

Date Data Type

DateFormula Data Type

BigText.AddText Method

12/31/2020 • 2 minutes to read • 📄 🗨️ 🌟

Version: Available from runtime version 1.0.

Adds a text string to a BigText variable.

Syntax

```
BigText.AddText(String: String [, Position: Integer])
```

Parameters

BigText Type: BigText An instance of the BigText data type.

String Type: String The string that will be added to the BigText variable. If this parameter is empty, then the BigText variable is not changed.

Position Type: Integer This is an optional parameter that defines the position in the BigText variable where the string is inserted.

https://docs.microsoft.com/en-us/dynamics365/business-central/dev-itpro/developer/methods-auto/bigtext/bigtext-addtext-string-integer-method

https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

DEV and Designer extensions are LOST on sandbox upgrade

Important

Apps that are published to a sandbox from the development environment or created using Designer are published within the scope of the service node that hosts the environment. When the sandbox is upgraded, these apps are removed because the environment is moved to another node that is running the new version. However, the data of the app is not removed, so you only have to re-publish and install the app to make it available.

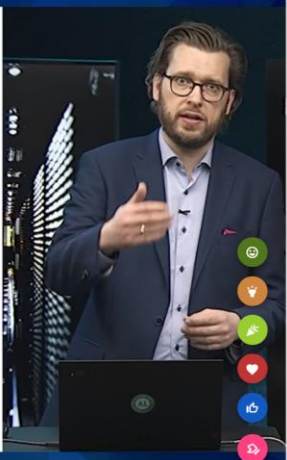
Apps that are uploaded to the environments of both types (production and sandbox) using the Upload Extension action from the Extension Management page are published within a global scope. When the environment is upgraded or moves, these apps are downloaded to the service node and installed, which means that they will not disappear.

Data is retained
Manually reinstall extension on upgraded sandbox

Always export sandbox Designer extensions when changed
As generated source is metadata/code, not data, and thus lost

v18: Deploying DEV from VSC to sandbox adds output window warning
Consider deploying as PTEs if sandbox needed for testing

<https://docs.microsoft.com/en-us/dynamics365/business-central/dev-itpro/administration/tenant-admin-center-environments>



https://events1.social27.com/msdyn365bclaunchevent2021w1/agenda/player/95494

Call to action

Try out report extensions, convert a few existing custom reports

Adopt complex return types

Create fallback handlers for unknown enum values

Use snapshot debugger to troubleshoot in cloud prod environments

Give us feedback
<https://github.com/microsoft/AL>
<https://aka.ms/bcideas>

