

Design Insights

Data Upgrade in Microsoft Dynamics NAV 2015

Dmitry Chadayev
Sr. Program Manager
Microsoft Dynamics NAV

August 2014

Data Upgrade

In Microsoft Dynamics NAV 2015,

we have introduced a new way of running the data upgrade logic which migrates data from the old table structure of the previous version of the product to the new table structure of the new version.

The data upgrade is now prepared and run with the help of specially prepared object – the upgrade codeunit. The data upgrade was enhanced significantly to optimize the performance and provide developers with a superior tool for testing, automating, tracking progress and troubleshooting of the data upgrade code.

Why upgrade data?

Field(s) deleted

Primary key changed

Table deleted

Table 50000 Gadget

1	nnnnnn	n	nn
2	nnnn	nn	n
3	nnnnnn	n	n
4	nnnnn	nnn	n

Field(s) type changed

Field(s) length decreased

Etc.

What to do with the data?

Migrate to another table

Change the format

Remove/obsolete

Table 50000 Gadget

1	nnnnnn	n	nn
2	nnnn	nn	n
3	nnnnnn	n	n
4	nnnnn	nnn	n

Save for history

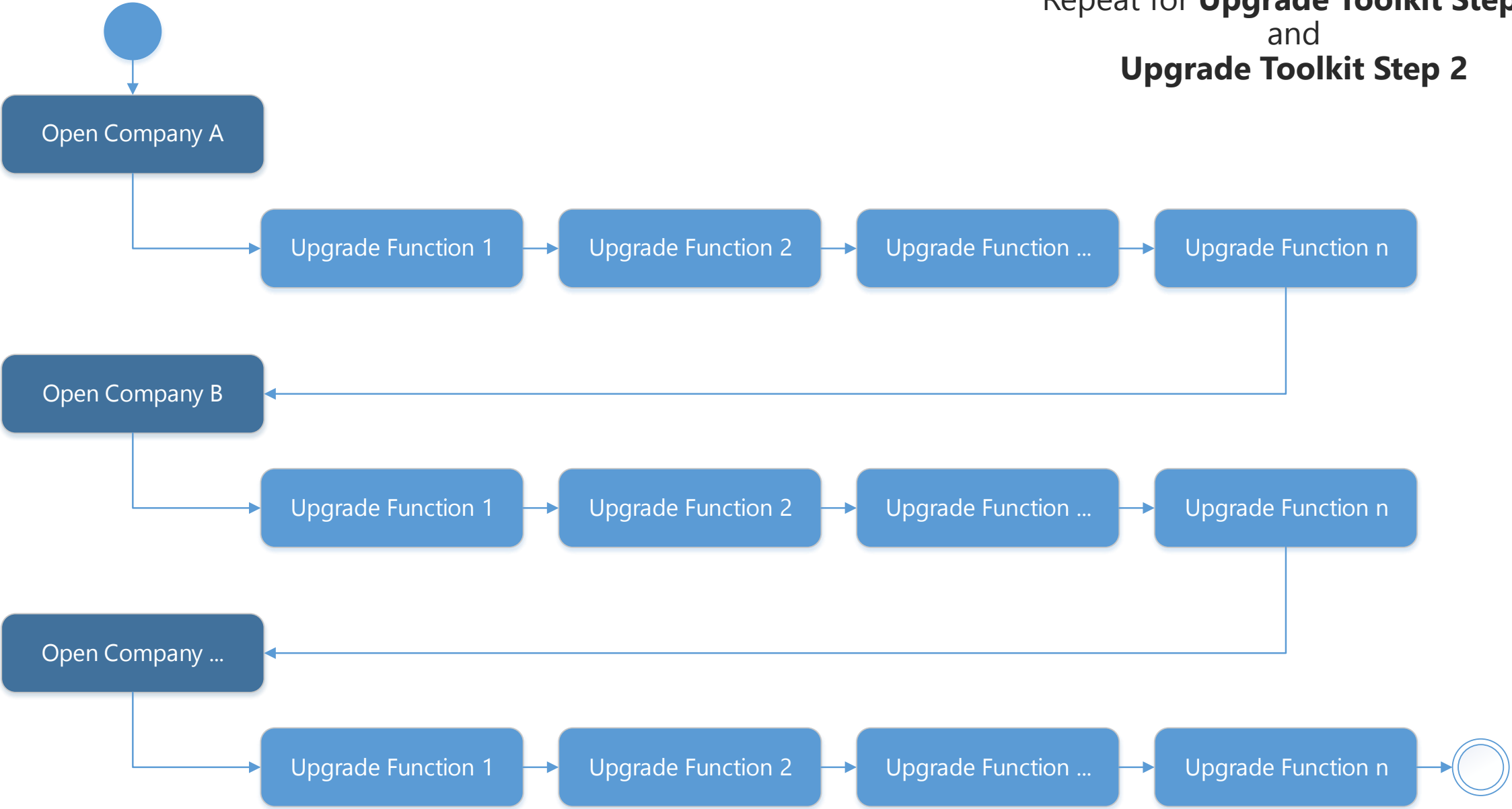
Rearrange/recalculate

Etc.

How did it work before?

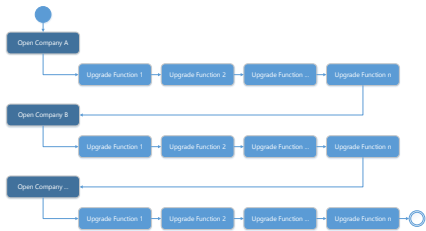
Previous Version Upgrade Toolkit Step 1		New Version	New Version Upgrade Toolkit Step 2	
Check upgrade preconditions Any specific setup or actions that should be done before we upgrade? Repeat for all companies.	Run data conversion – Step 1 Remove the data from the obsolete/changed tables or fields, or move it to upgrade tables for further processing. Repeat for all companies.	Convert the database Upgrade the system tables and other objects to the format of the new version. Done by using development environment	Check upgrade preconditions Any specific setup or actions that should be done before we upgrade? Repeat for all companies.	Run data conversion – Step 2 Run custom data upgrade code, process and move the data from the upgrade tables to the new tables. Repeat for all companies.
Track upgrade progress and state, troubleshoot upgrade errors			Track upgrade progress and state, troubleshoot upgrade errors	

Repeat for **Upgrade Toolkit Step 1**
and
Upgrade Toolkit Step 2



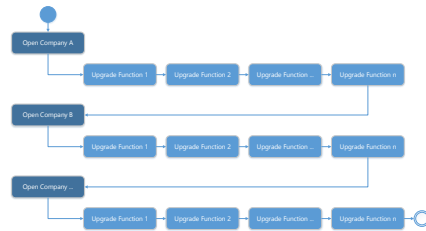
Just to put it into perspective...

Upgrade Toolkit Step 1



+

Upgrade Toolkit Step 2

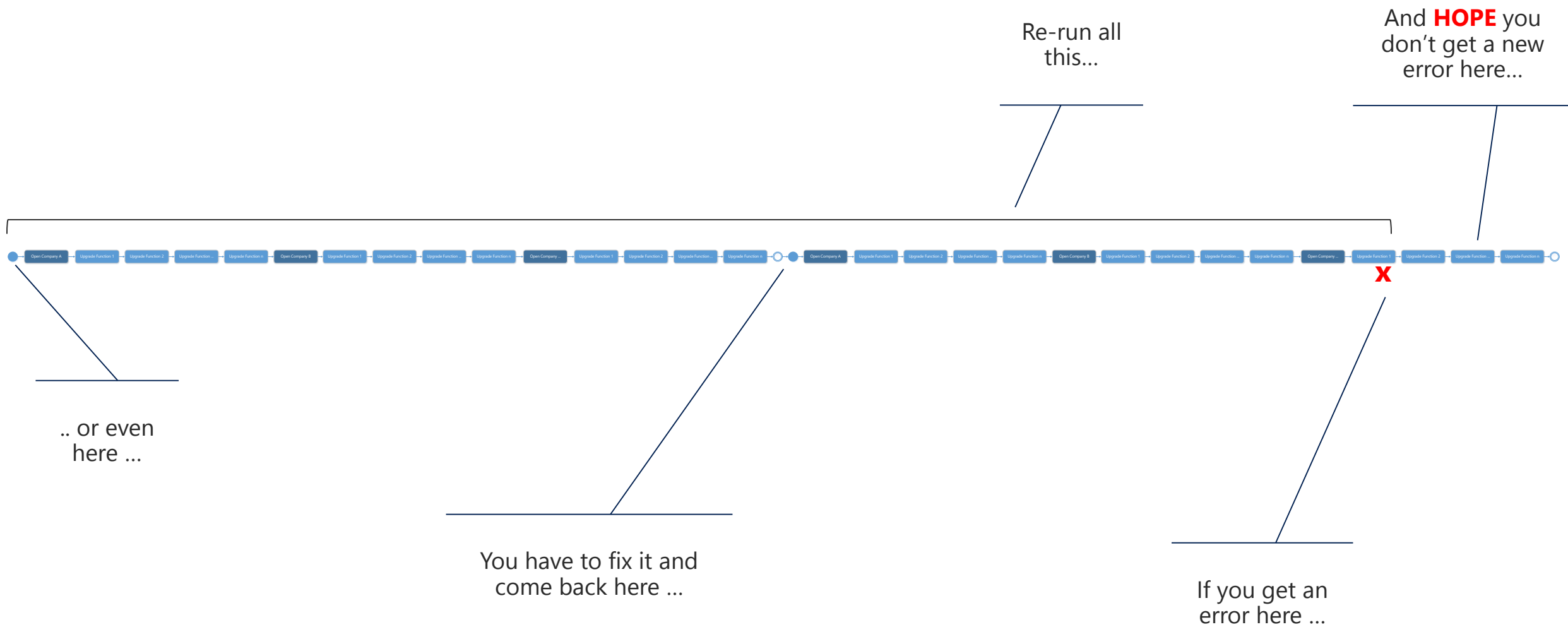


=

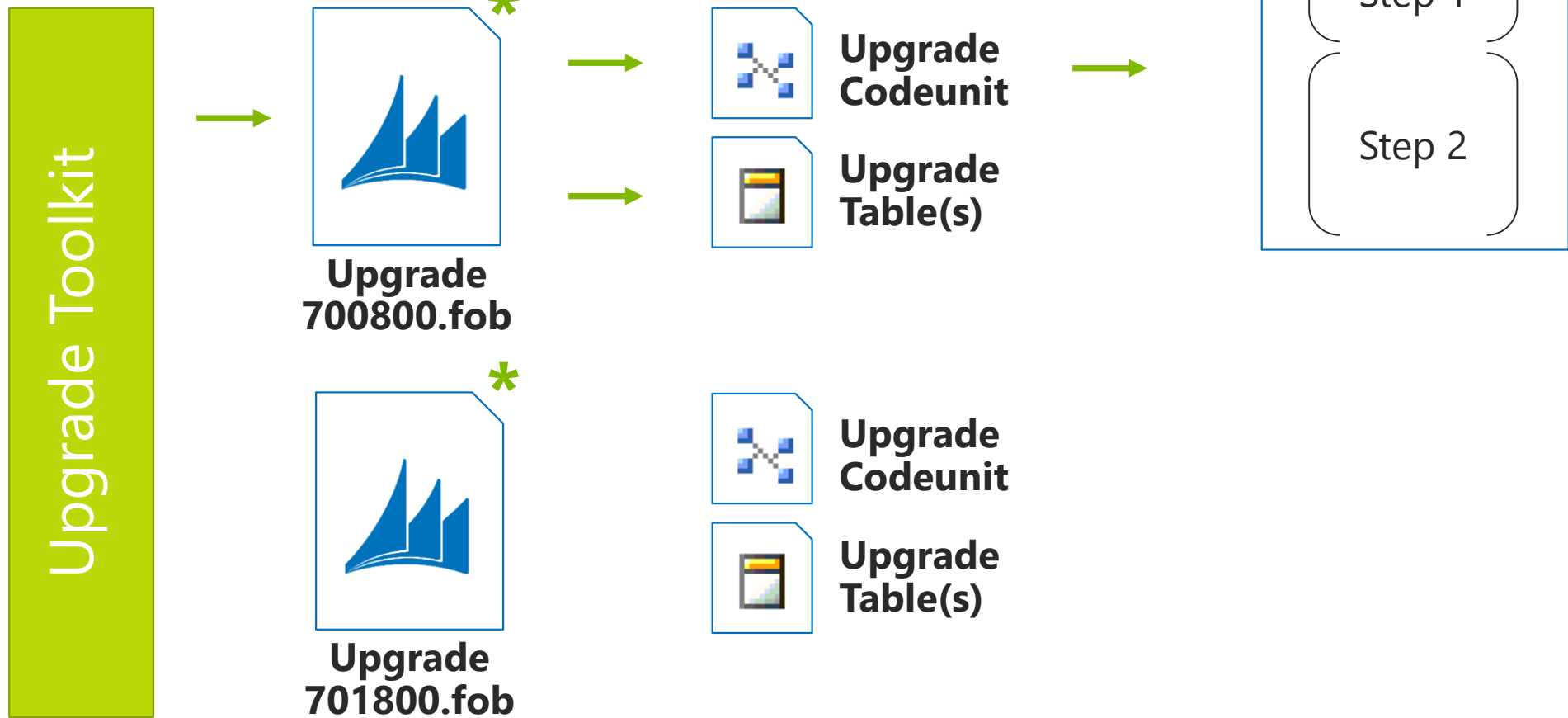


Upgrade Function 1

= Assume it can run
for a few minutes to a
few **hours**...



How is it working now?



* Compiled by Microsoft Dynamics NAV 2015 version (8.0)

Upgrade Codeunit

GetTableSyncSetup - Properties

Property	Value
ID	3
Local	<No>
FunctionType	TableSyncSetup

UPG7180.W1 - Properties

Property	Value
ID	104025
Name	UPG7180.W1
Permissions	<Undefined>
CFRONTMayUsePermissions	<No>
TableNo	<Undefined>
SingleInstance	<No>
Subtype	Upgrade

Codeunit 104025 UPG7180.W1 - C/AL Editor

```
OnRun()
[TableSyncSetup] GetTableSyncSetup(VAR TableSynchSetup : Record "Table Synch. Setup")

DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Payment Terms",170001,TableSynchSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Vendor",170002,TableSynchSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Vendor Invoice Disc.",170003,TableSynchSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Job",170004,TableSynchSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(455,0,TableSynchSetup.Mode::Force);

[CheckPrecondition] CheckUPGPreconditions()
{...}

[Upgrade] UpgradePaymentTerms()
IF UPGPaymentTerms.FINDSET THEN
REPEAT
{...}
UNTIL UPGPaymentTerms.NEXT = 0;

[Upgrade] UpgradeVendors()
IF UPGVendors.FINDSET THEN
REPEAT
{...}
UNTIL UPGVendors.NEXT = 0;

[Upgrade] UpgradeVendorInvDisc()
IF UPGVendorInvoiceDisc.FINDSET THEN
REPEAT
{...}
UNTIL UPGVendorInvoiceDisc.NEXT = 0;

[Upgrade] UpgradeJobs()
IF UPGJobs.FINDSET THEN
REPEAT
{...}
UNTIL UPGJobs.NEXT = 0;
```

CheckUPGPreconditions - Prop...

Property	Value
ID	7
Local	<No>
FunctionType	CheckPrecondition

UpgradePaymentTerms - Prop...

Property	Value
ID	1
Local	<No>
FunctionType	Upgrade

* Upgrade codeunits cannot be run directly. They can only be invoked by Microsoft NAV Server

Upgrade Toolkit Step 1 is “outsourced” to the NAV server.

The server executes it during the
table schema synchronization.

Both steps now reside within a
single upgrade codeunit and are
**executed by the latest version of
the product.**

Copy or move data from the
obsolete/changed tables and
fields into upgrade tables
or force-delete them if data is
not needed.
(previously known as
UPGTK Step 1)

Check upgrade
preconditions

Run custom data upgrade
code, process and move
the data from the upgrade
tables to the new tables.
(previously known as
UPGTK Step 2)

Upgrade Codeunit

```
Codeunit 104025 UPG7180.W1 - C/AL Editor

OnRun()

[TableSyncSetup] GetTableSyncSetup(VAR TableSynchSetup : Record "Table Synch. Setup")

DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Payment Terms",170001,TableSynchSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Vendor",170002,TableSynchSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Vendor Invoice Disc.",170003,TableSynchSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Job",170004,TableSynchSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(455,0,TableSynchSetup.Mode::Force);

[CheckPrecondition] CheckUPGPreconditions()

{...}

[Upgrade] UpgradePaymentTerms()
IF UPGPaymentTerms.FINDSET THEN
REPEAT
{...}
UNTIL UPGPaymentTerms.NEXT = 0;

[Upgrade] UpgradeVendors()
IF UPGVendors.FINDSET THEN
REPEAT
{...}
UNTIL UPGVendors.NEXT = 0;

[Upgrade] UpgradeVendorInvDisc()
IF UPGVendorInvoiceDisc.FINDSET THEN
REPEAT
{...}
UNTIL UPGVendorInvoiceDisc.NEXT = 0;

[Upgrade] UpgradeJobs()
IF UPGJobs.FINDSET THEN
REPEAT
{...}
UNTIL UPGJobs.NEXT = 0;
```

Upgrade Toolkit Step 1 – Schema Synchronization

Objectives:

- Save the data from the tables and columns you are about to change in the upgrade tables (if the data is still needed)
- Apply table changes

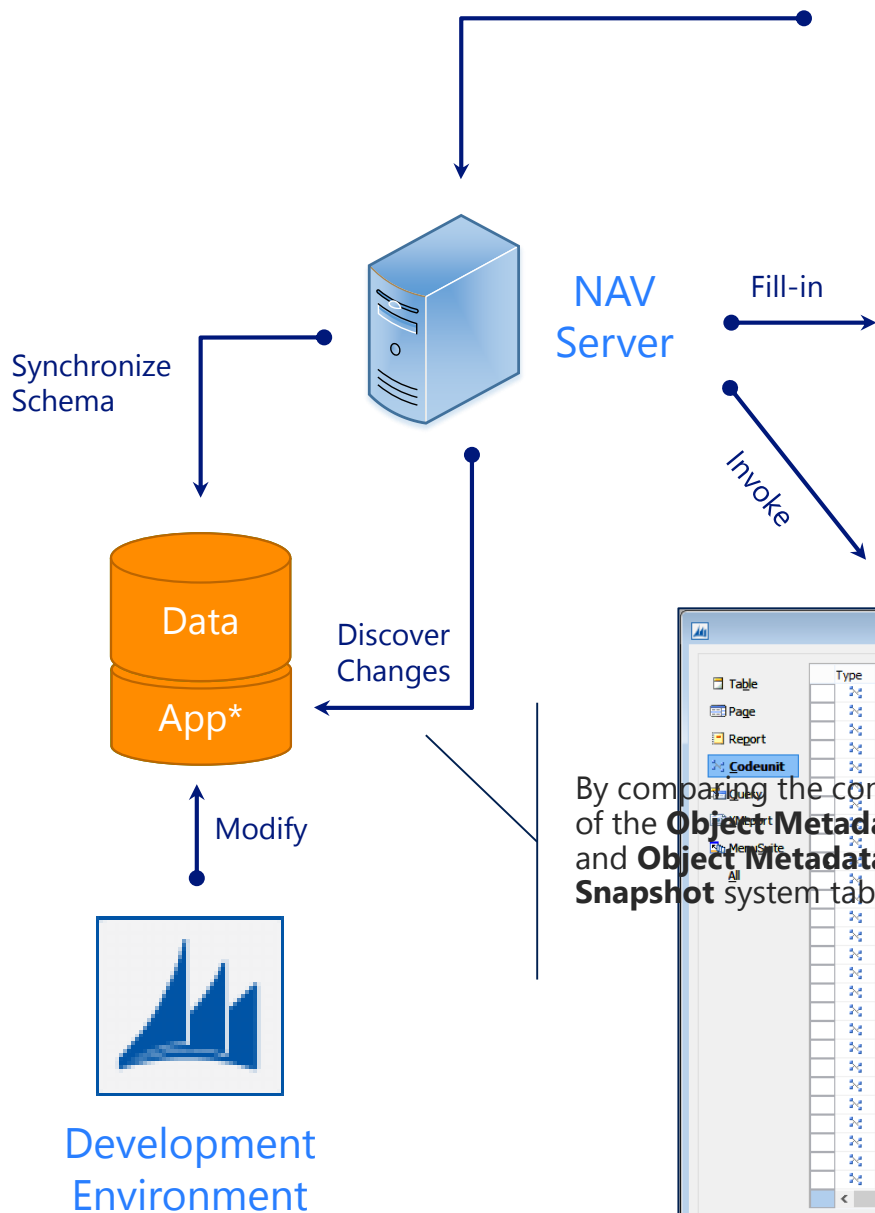


Table ID	Upgrade Table ID	Mode
3	170001	Copy
23	170002	Copy
25	170003	Copy
167	170004	Move
445	0	Force

Virtual Table
2000000135
Table Synch. Setup,
containing IDs of all
tables with schema
changes

Codeunit 55000 Upgrade Codeunit - C/AL Editor

```

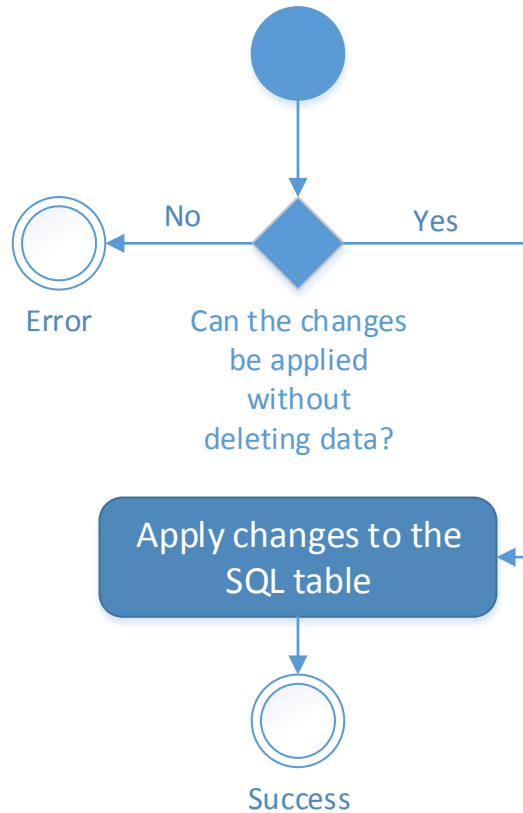
Documentation()
OnRun()
[TableSyncSetup] GetTableSyncSetup(VAR TableSyncSetup : Record "Table Synch. Setup")

DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Payment Terms",17001,TableSyncSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Vendor",17002,TableSyncSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Vendor Invoice Disc.",17003,TableSyncSetup.Mode::Copy);
DataUpgradeMgt.SetTableSyncSetup(DATABASE::"Job",17004,TableSyncSetup.Mode::Move);
DataUpgradeMgt.SetTableSyncSetup(445,0,TableSyncSetup.Mode::Force);

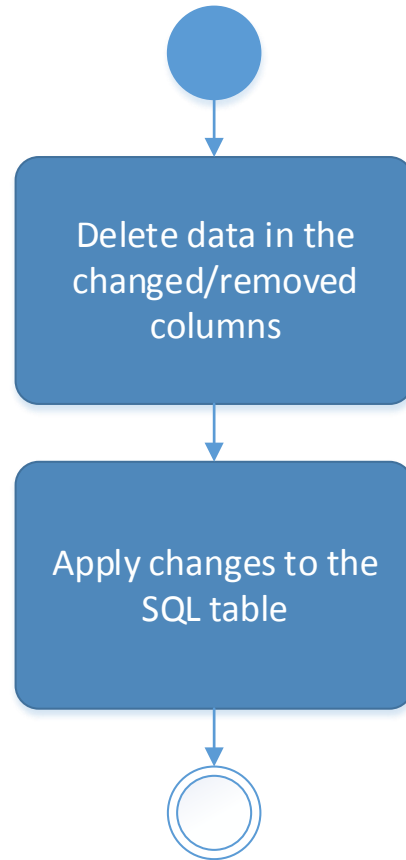
[CheckPrecondition] CheckPreconditions()
{...}

```

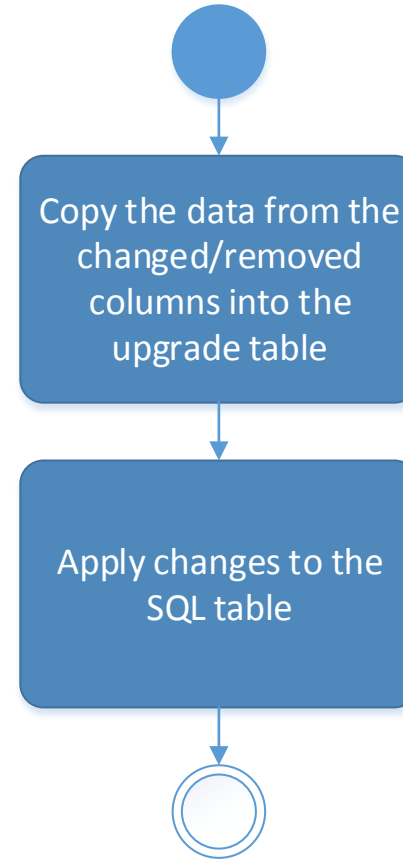
Check



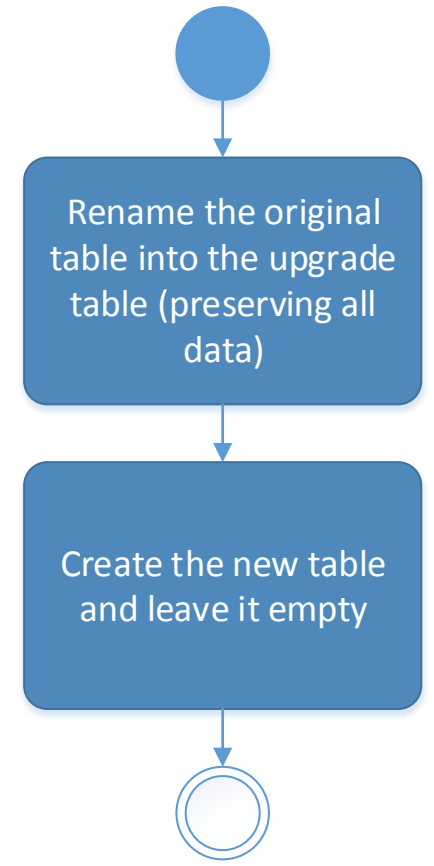
Force



Copy



Move



Upgrade Toolkit Step 2 – Data Upgrade

Objectives:

- Process the data saved in the upgrade tables
- Run any additional data manipulations to align the data to the requirements and formats of the new solution

Before:



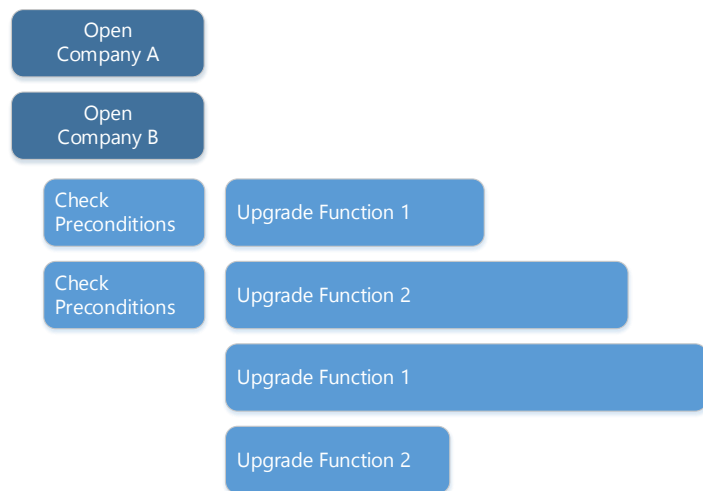
Now:

All “check preconditions” and “upgrade” functions in an upgrade codeunit can be executed in parallel across all companies in a database.

Before:



Now:

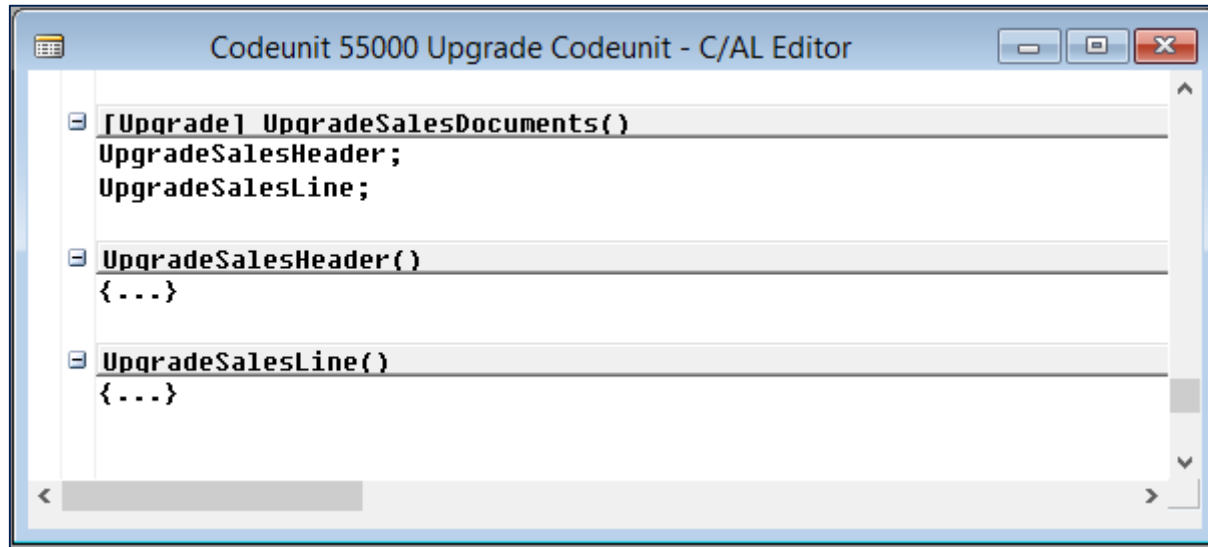


Each upgrade function runs in its **own system session**.

We have **preserved** the ability to run the upgrade functions **sequentially** (for example, in cases where they lock each other).

Data upgrade is an **asynchronous process**, therefore you can start it for **multiple tenants** in parallel too.

Combining order-
dependent functions



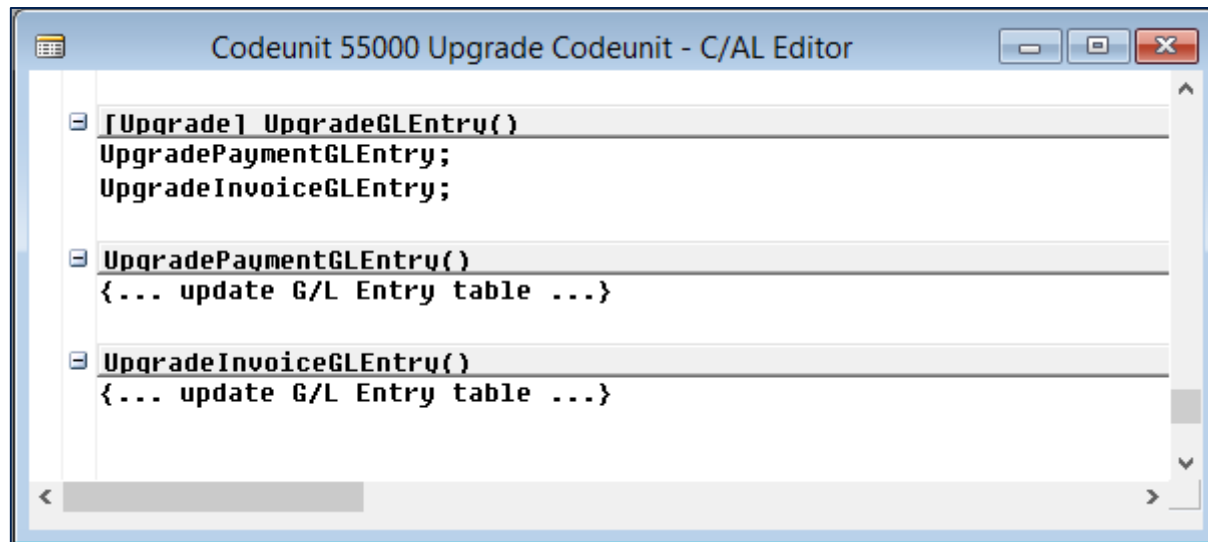
```
Codeunit 55000 Upgrade Codeunit - C/AL Editor

[Upgrade] UpgradeSalesDocuments()
UpgradeSalesHeader;
UpgradeSalesLine;

UpgradeSalesHeader()
{...}

UpgradeSalesLine()
{...}
```

Combining functions
which lock each other



```
Codeunit 55000 Upgrade Codeunit - C/AL Editor

[Upgrade] UpgradeGLEntry()
UpgradePaymentGLEntry;
UpgradeInvoiceGLEntry;

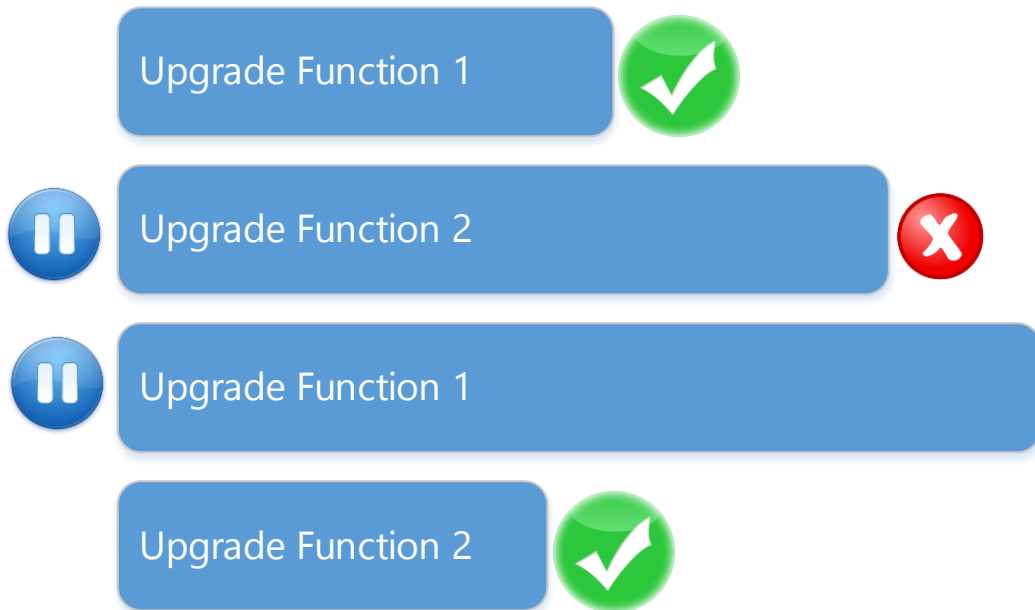
UpgradePaymentGLEntry()
{... update G/L Entry table ...}

UpgradeInvoiceGLEntry()
{... update G/L Entry table ...}
```

Upgrade functions should be built so that they are:

- **independent** of each other
- **not locking** each other
- not expecting any particular **order of execution**

If these conditions cannot be met, you can mark the related (for example, the ones locking each other) and order-dependent functions as [Normal] and then **call them from an upgrade function**.

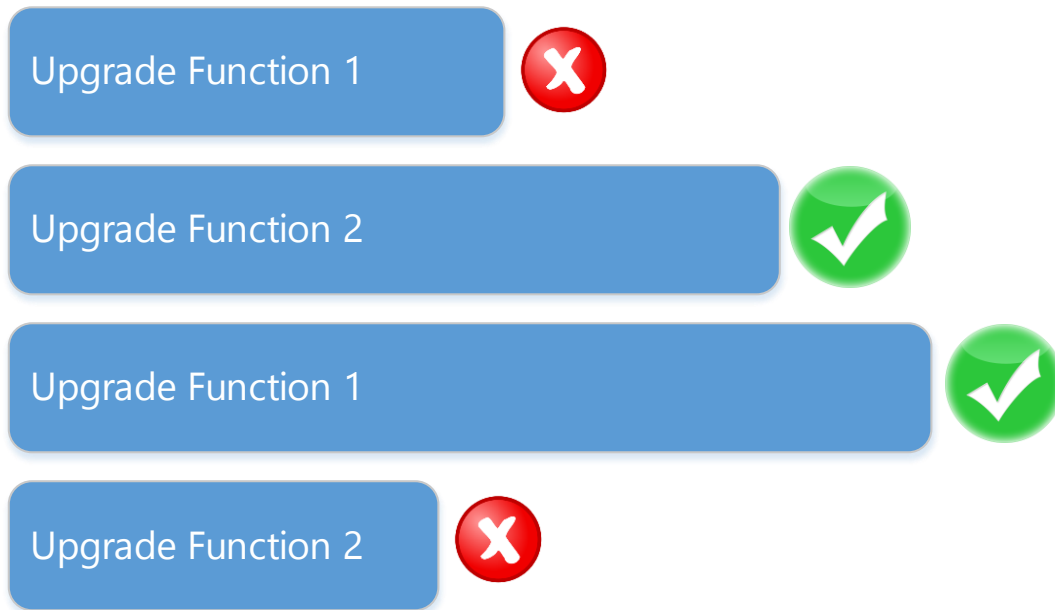


When errors occur in one of the functions:

- the functions which are already **completed, will remain completed**
- the function which is **failed** and the functions which were **in progress** are rolled back and **set to a “pending” state**

Once the error is fixed, you can **resume the upgrade process** by restarting the “pending” functions.

You can choose to restart a **particular function** from a **particular codeunit** for a **particular company**.



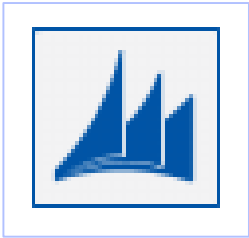
You can decide if you want to stop the upgrade on the first error, or let the other upgrade functions continue and **display all errors** at the end.

You can observe **the progress** and get a **complete overview** of the execution results **during and after upgrade**.

How do I run data
upgrade?



Microsoft Dynamics NAV 2015
Administration Shell



Microsoft Dynamics NAV 2015
Development Environment

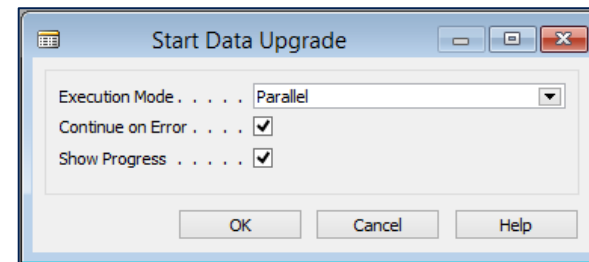
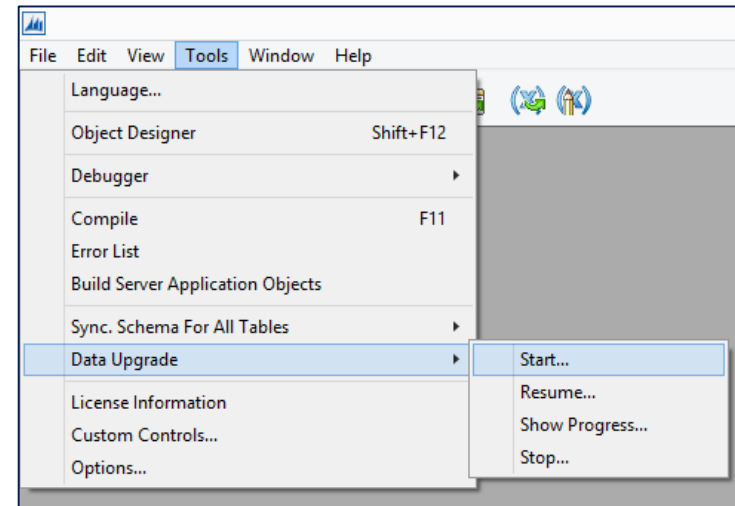
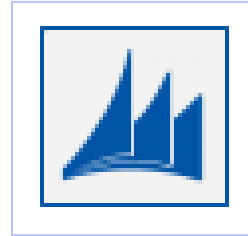


Start-NAVDataUpgrade

Get-NAVDataUpgrade

Resume-NAVDataUpgrade

Stop-NAVDataUpgrade





Get-NAVDataUpgrade -Details | ogv

```
Administrator: Microsoft Dynamics NAV Create Community Technical Preview Ad... - [X]
Cmdlet Remove-NAVApplication Microsoft...
Cmdlet Remove-NAVCompany Microsoft...
Cmdlet Remove-NAVServerInstance Microsoft...
Cmdlet Remove-NAVServerPermission Microsoft...
Cmdlet Remove-NAVServerPermissionSet Microsoft...
Cmdlet Remove-NAVServerSession Microsoft...
Cmdlet Remove-NAVServerUser Microsoft...
Cmdlet Remove-NAVServerUserPermissionSet Microsoft...
Cmdlet Remove-NAVWebService Microsoft...
Cmdlet Rename-NAVCompany Microsoft...
Cmdlet Resume-NAVDataUpgrade Microsoft...
Cmdlet Save-NAVTenantConfiguration Microsoft...
Cmdlet Set-NAVServerConfiguration Microsoft...
Cmdlet Set-NAVServerInstance Microsoft...
Cmdlet Set-NAVServerPermission Microsoft...
Cmdlet Set-NAVServerPermissionSet Microsoft...
Cmdlet Set-NAUS Microsoft...
Cmdlet Set-NAUW Microsoft...
Cmdlet Start-NAU Microsoft...
Cmdlet Stop-NAU Microsoft...
Cmdlet Sync-NAU Microsoft...

PS C:\WINDOWS\system32>
PS C:\WINDOWS\system32>
```

You can request the state of **the upgrade progress periodically**, observing which functions are being executed, which are completed or failed to complete...

Get-NAVDataUpgrade DynamicsNAV80 -Details ogv							
Filter							
+ Add criteria							
SessionId	CodeunitId	FunctionName	CompanyName	StartTime	Duration	State	Error
169	55,000	CheckPreconditions	CRONUS International Ltd.	8/4/2014 5:01:52 PM	00:00:00	Completed	
170	55,000	UpgradeSalesOrder	CRONUS International Ltd.	8/4/2014 5:01:52 PM	00:00:05.8186638	Completed	
171	55,000	UpgradeSalesInvo...	CRONUS International Ltd.	8/4/2014 5:01:52 PM		InProgress	
172	55,000	UpgradePurchase...	CRONUS International Ltd.	8/4/2014 5:01:52 PM	00:00:09.5072848	Completed	
174	55,000	UpgradePurchase...	CRONUS International Ltd.	8/4/2014 5:01:52 PM		InProgress	
173	55,000	UpgradeGLEntry	CRONUS International Ltd.	8/4/2014 5:01:52 PM	00:00:00	Completed	



Get-NAVDataUpgrade –Progress

```
Administrator: Microsoft Dynamics NAV Create Community Technical Preview Ad... - [X]
Cmdlet Remove-NAVServerInstance Microsoft...
Cmdlet Remove-NAVServerPermission Microsoft...

The data upgrade process is in progress...
Percent complete: 33.33 %
[ooooooooooooooooooooooooooooo]

Cmdlet Rename-NAVCompany Microsoft...
Cmdlet Resume-NAVDataUpgrade Microsoft...
Cmdlet Save-NAVTenantConfiguration Microsoft...
Cmdlet Set-NAVServerConfiguration Microsoft...
Cmdlet Set-NAVServerInstance Microsoft...
Cmdlet Set-NAVServerPermission Microsoft...
Cmdlet Set-NAVServerPermissionSet Microsoft...
Cmdlet Set-NAVServerUser Microsoft...
Cmdlet Set-NAVWebServerInstanceConfiguration Microsoft...
Cmdlet Start-NAVDataUpgrade Microsoft...
Cmdlet Stop-NAVDataUpgrade Microsoft...
Cmdlet Sync-NAVTenant Microsoft...

PS C:\WINDOWS\system32> Get-NAVDataUpgrade DynamicsNAV80 -Progress
PS C:\WINDOWS\system32> Start-NAVDataUpgrade DynamicsNAV80 -force
PS C:\WINDOWS\system32> Get-NAVDataUpgrade DynamicsNAV80 -Progress
```

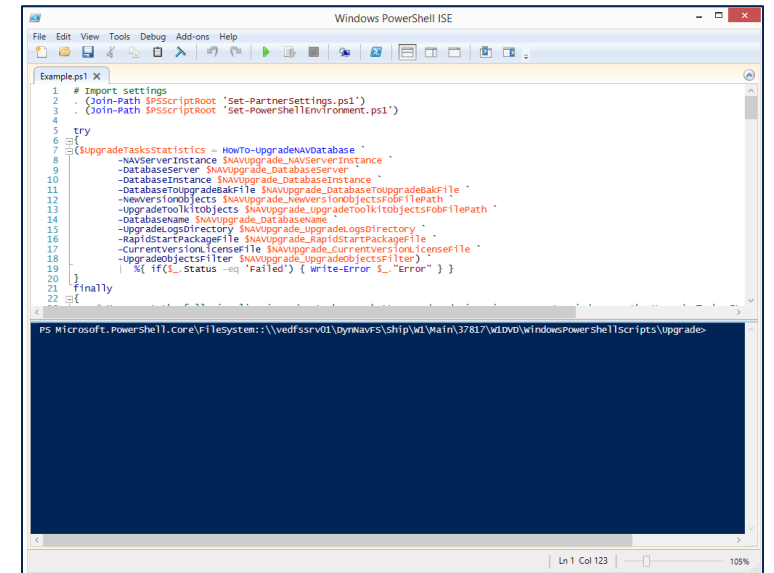
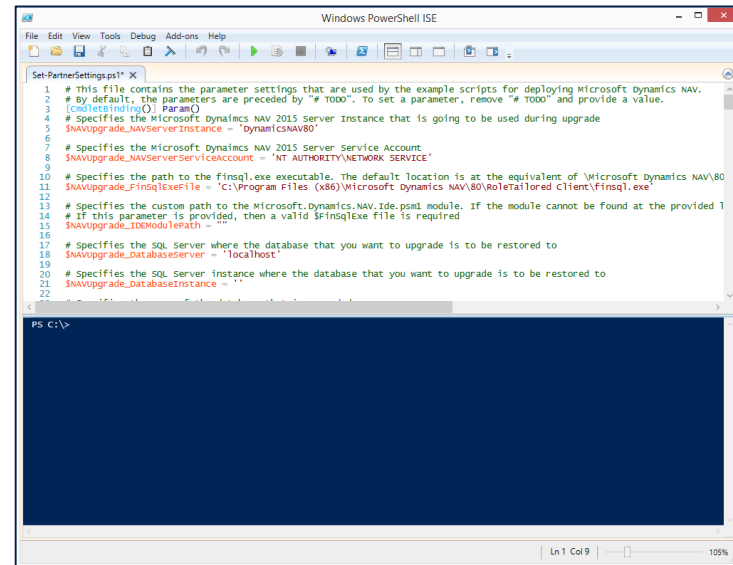
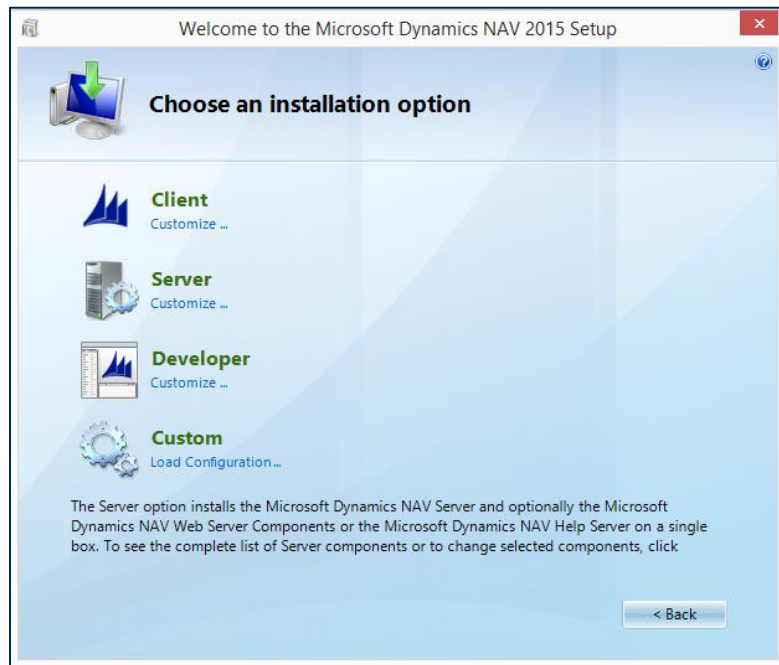
... Or you can observe **the upgrade progress in real-time.**

How do I automate data
upgrade?

Install Microsoft
Dynamics NAV 2015

Fill in the
parameters in
Set-
PartnerSettings
.ps1

Run
Example.ps1 *



* If you run in Windows PowerShell ISE, remember to import the NAVAdminTools module:
Import-Module "C:\Program Files\Microsoft Dynamics NAV\80\Service\NavAdminTool.ps1"

Data Upgrade

Microsoft Dynamics NAV 2015

Parallel execution

Resume after error

Asynchronous

Catch all errors

Detailed progress

+ "old-style" Serial
execution and error
handling

Benefits

The new approach:

- **Simplifies the upgrade environment** because all steps are performed by the latest version of the product.
- **Increases the data upgrade performance** because it is possible to execute the data upgrade functions in parallel within the upgrade codeunit and across companies.
- **Reduces the amount of code to be written** to handle the data upgrade because some actions are now executed automatically by Microsoft Dynamics NAV Server.
- **Minimizes the number of manual actions** which should be performed during the data upgrade, making the data upgrade process less error-prone.
- **Uses familiar upgrade toolkit design concepts** (such as upgrade tables, upgrade functions).
- **End-to-end sample PowerShell script** for performing data upgrade (on the DVD).

For more information...

Visit the Microsoft Dynamics NAV Getting Ready page

On this page you will have access to resources for developers, consultants, sales and business decision makers. For more detailed information please visit our readiness library, which includes a variety of materials such as how-to videos, demo scripts, white papers and slide decks.

<http://go.microsoft.com/fwlink/?LinkID=510890>

