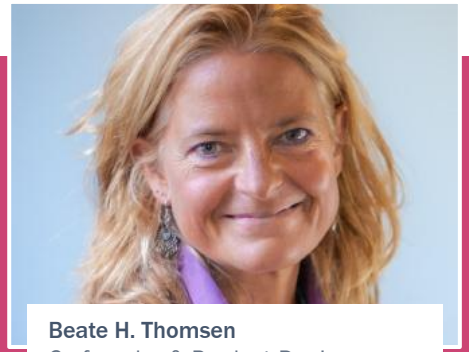
RAPIDI
DATA INTEGRATION MADE EASY

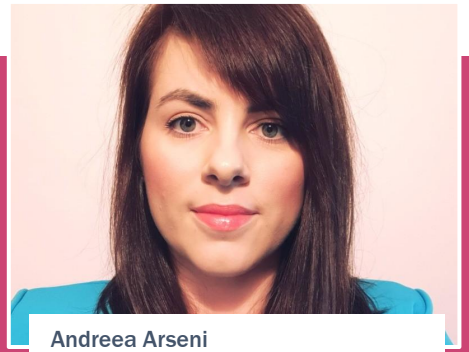
Live Webinar

Upgrade Dynamics NAV to Business Central

What happens with your integration



Beate H. Thomsen
Co-founder & Product Design

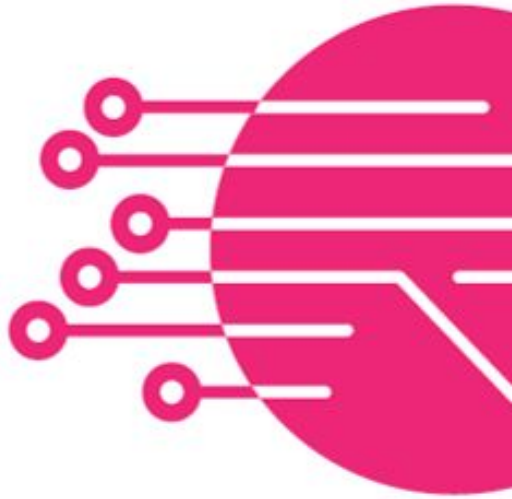


Andreea Arseni
Senior Data Integration Consultant

salesforce PARTNER SINCE 2007

Microsoft Dynamics Microsoft Partner SINCE 1987

AGENDA



- Welcome & Introduction
- Switching from Dynamics NAV to D365 Business Central
- On-premise vs Cloud-based Business Central
- 6-Steps Upgrade Approach
- Summary & Take Aways
- Q&A

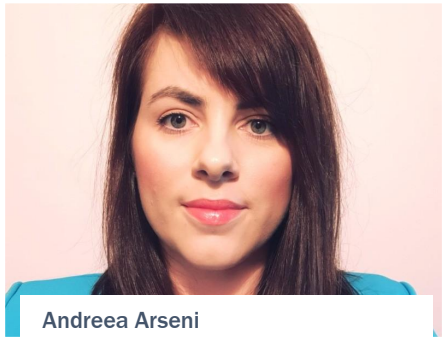
Welcome & Introduction

Who we are and why we are here today

Welcome & Introduction

Beate H. Thomsen
Co-founder & Product Design

- Co-founder of Rapidi
- 20+ years shaping the product's vision and development
- Guiding principle: "Keep it simple, functional yet beautiful"
- Master's Degree in IT, design and business - specializing in website usability
- Speaks 6 languages: Danish, German, English, Spanish, French & Catalan
- Passionate about nature, hiking, travelling, discovering new cultures & trends



Andreea Arseni
Senior Data Integration Consultant

- Senior Data Integration Consultant at Rapidi
- In the iPaaS world since 2017, delivering system integrations
- Focus: Dynamics 365 (Business Central & Finance), Salesforce, and HubSpot
- Speaks: Romanian, English & Spanish
- Island-based and sunset enthusiast

Rapidi is an iPaaS - a no-code data integration platform with 30+ years of experience in data integration and data replication - that lets you connect different business system applications such as the major ERPs and CRMs like the Microsoft Dynamics Products suite with Salesforce, HubSpot, Dynamics CRM / Sales and beyond.

Switching from Dynamics NAV to Business Central

What it means for your integration

Common Assumption: "We will have to rebuild it all"

What teams assume

Ask a project team whether the integration between NAV and their CRM will survive the move and the answer is usually no.

Not because anyone analysed it. Because that is what happened the last time they changed a core system.

In a lot of NAV estates that has genuinely been true. It is also the only outcome most teams have heard about, so it becomes the default expectation for every estate.

What actually decides it

Only one end of the integration is moving.
The other side is untouched. The question is how the integration reached into NAV, and whether that reach still lands on anything in Business Central.

Some integrations reached into NAV's internals - reading the tables directly, or running as code inside NAV. Business Central does not expose those layers, so that end has nothing left to hold onto and has to be **rebuilt**.

A second group reached NAV through a published interface and never knew its internals, only an address and a contract. Those get **reconfigured:** repoint the connection, set up the new authentication, remap what was renamed.

A third group sits in between - custom code written directly against NAV's web services. The connection survives, but the code was written for a specific version and has to be **adapted and/or recoded**.

Your integration's compatibility was determined when it was originally built, not when you perform an upgrade.
Therefore, rather than asking whether the integration will survive, the more insightful question is:
"How does this integration access data?" There are three scenarios.

RAPIDI
DATA INTEGRATION MADE EASY

Three Different Cases: Which One is Yours

1

Rebuild

How to spot it: connectors wired to a specific NAV version, with no reusable mapping layer underneath. The field mapping and business rules live inside the connector itself, rather than being held anywhere you could lift them out from. If someone asks where the mapping is documented, the answer is the code.

The work is the same whether you find it now or in month three. What changes is whether it was planned or discovered.

2

Reconfigure

How to spot it: mapping and business logic held separately from the system version, in an integration platform like Rapidi's rather than inside a connector. The configuration names a service endpoint, not a database or a NAV object. If someone asks where the mapping is documented, you can open it and read it.

NAV from roughly 2016 onwards and Business Central share the same connection type.

3

Adapt custom code

How to spot it: custom code bound to a particular API version or table schema. It reaches NAV through web services, so there is a proper interface underneath, but someone wrote logic on top of it rather than configuring it. If someone asks where the mapping is documented, the answer is the code, but the connection is not part of the code.

It can carry forward, but adapting it has an engineering timeline, not a configuration one. It deserves its own estimate.

This is about how the integration was built, not how old NAV is. An estate on an older version with the mapping held separately is in better shape than a recent one with years of customisation welded into the connector. Reuse is a genuine property, not a magic one.

RAPIDI
DATA INTEGRATION MADE EASY

On-Premise vs Cloud-based Business Central

- *the environment specific differences*

Dynamics NAV -> Business Central On Premise/Cloud

- *what carries over and what you adjust*

What stays the same, on-premise or cloud

- **The data model.** Business Central inherited NAV's tables - Customer, Contact, Item, Sales Header/Line, Cust. Ledger Entry. Your field mappings largely port over as-is.
- **The integration design.** Scope, direction, matching keys, dedupe rules, transformation logic, conflict handling, sync frequency - none of that is a function of where Business Central runs.
- **The CRM side.** HubSpot / Salesforce / D365 CE doesn't know or care where your ERP lives. Zero rework there.
- **Your Rapidi transfers.** Same templates, same transfer definitions. You can reuse the same integration and additional logic to accommodate this new change.

What you actually adjust

For example, an existing NAV ↔ Salesforce integration carries over to Business Central with the same setup. The work is **adjustment**, not **redesign**:

1. Re-point the connection to the new Business Central environment
2. Reconcile mappings against Business Central's tables and fields: most are identical, some renamed, obsoleted, or replaced.
3. Re-expose any custom fields that were added to NAV
4. Re-test and re-schedule.

You're repurposing, not rebuilding.

On-Premise vs Cloud-based Business Central

- *the environment specific differences*

		Dynamics NAV (today)	Business Central On-Premise	Business Central Cloud (SaaS)
1	**Access method**	SQL direct or SOAP/OData web services	Same options still available today	API/OData only - **no SQL access**
2	**Authentication**	Windows / NavUserPassword	Windows / UserPassword	**OAuth 2.0 only** - basic auth and web service keys are gone
3	**Where the engine sits**	On your network	On your network, or cloud + VPN/local connector	Direct cloud-to-cloud, no gateway or firewall work
4	**Custom fields**	Read straight from the table	Same, or via extension	Must be exposed via **AL table extension** + API page
5	**Volume & throughput**	Unlimited, only hardware-bound	Unlimited	**Throttled** - rate limits, HTTP 429, needs batching + retry
6	**Test environments**	Manual restore of a copy	Manual restore of a copy	Sandbox on demand, refreshed from prod

💡 **In short:** on-premise changes almost nothing about how you connect. Cloud changes how you connect, not what you connect.

6-Steps Upgrade Approach

- *a guide to a successful upgrade project*

Six steps to a migration you actually control

DYNAMICS NAV TO BUSINESS CENTRAL, OR ANY SYSTEM MOVE

1

Map and scope

Decide what moves and what stays behind as history. Agree the cutoff date up front.

2

Cleanse at source

Deduplicate and fix the data in the old system, not later in the new one.

3

Dry-run a subset

Test-load a representative slice and see what actually breaks, while it is still cheap.

4

Run old and new in parallel

Both systems live and current with two-way sync. No freeze, no lost work.

5

Validate and reconcile

Against acceptance criteria you agreed before you started. This is the step teams skip.

6

Cut over on your terms

Switch off the old system once the new one is proven, not on a date fixed months earlier.

Same order, every time. The difference is treating step five as **mandatory**, not as something you get to if there is time.

RAPIDI
DATA INTEGRATION MADE EASY

Summary & Take Aways

1 Which case you are in

Only one end of the integration is moving. What matters is what it reached into.

Rebuild: the mapping lives inside the connector code.

Reconfigure: the mapping is held in an integration platform where we place Rapidi, so one endpoint changes.

Adapt: custom code over web services, carries forward but needs its own estimate.

Decided when it was built, not when you upgrade.

2 On-premise or cloud changes how you connect

The data model, the integration design and the CRM side all carry over.

What changes is the connection: access method, authentication, custom fields, throughput.

On-premise keeps SQL, but not the same schema.

Cloud is API and OData only, throttled, OAuth 2.0.

Not a redesign. An adjustment.

3 Run in parallel, then cut over

Six steps, the same order every time.

Map and scope with a cutoff date, cleanse at source, dry-run a subset.

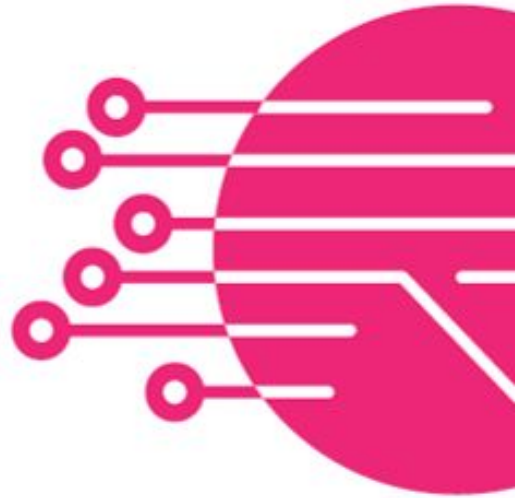
Then run old and new together on a two-way sync. No freeze, no lost work.

Validate against criteria you agreed before you started. That is the step teams skip.

Switch the old system off once the new one is proven.

If you already run an integration, an upgrade can be a reconfigure rather than a rebuild. The question worth answering before the budget is fixed is which of the three cases you are in.

RAPIDI
DATA INTEGRATION MADE EASY



Questions & Next Steps

info@rapidionline.com
www.rapidionline.com
MyRapidi.com/wiki

What to do next?

Download the 6-Step Upgrade Guide

The full parallel-run playbook, with a checklist per step: map and scope, cleanse at source, dry-run a subset, two-way sync in parallel, validate and reconcile, cut over on your terms.

https://www.rapidionline.com/resources/six-step-migration-guide

Contact us for an assessment

Book an appointment with Rapidi to discuss your data integration and migration project.

https://www.rapidionline.com/book-a-free-consulation-rapidi-data-integration-solutions

RAPIDI
DATA INTEGRATION MADE EASY

Please give us feedback

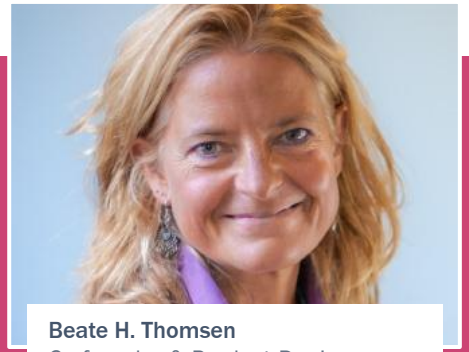
https://www.getfeedback.com/r/B18KMiaV

Follow & Connect

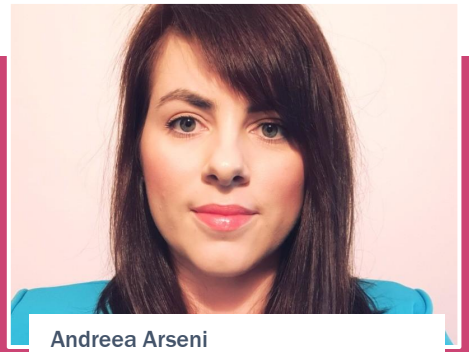
info@rapidionline.com
www.rapidionline.com
MyRapidi.com/wiki

RAPIDI
DATA INTEGRATION MADE EASY

Thank you!



Beate H. Thomsen
Co-founder & Product Design



Andreea Arseni
Senior Data Integration Consultant

salesforce PARTNER SINCE 2007

Microsoft Dynamics Microsoft Partner SINCE 1987

Follow & Connect