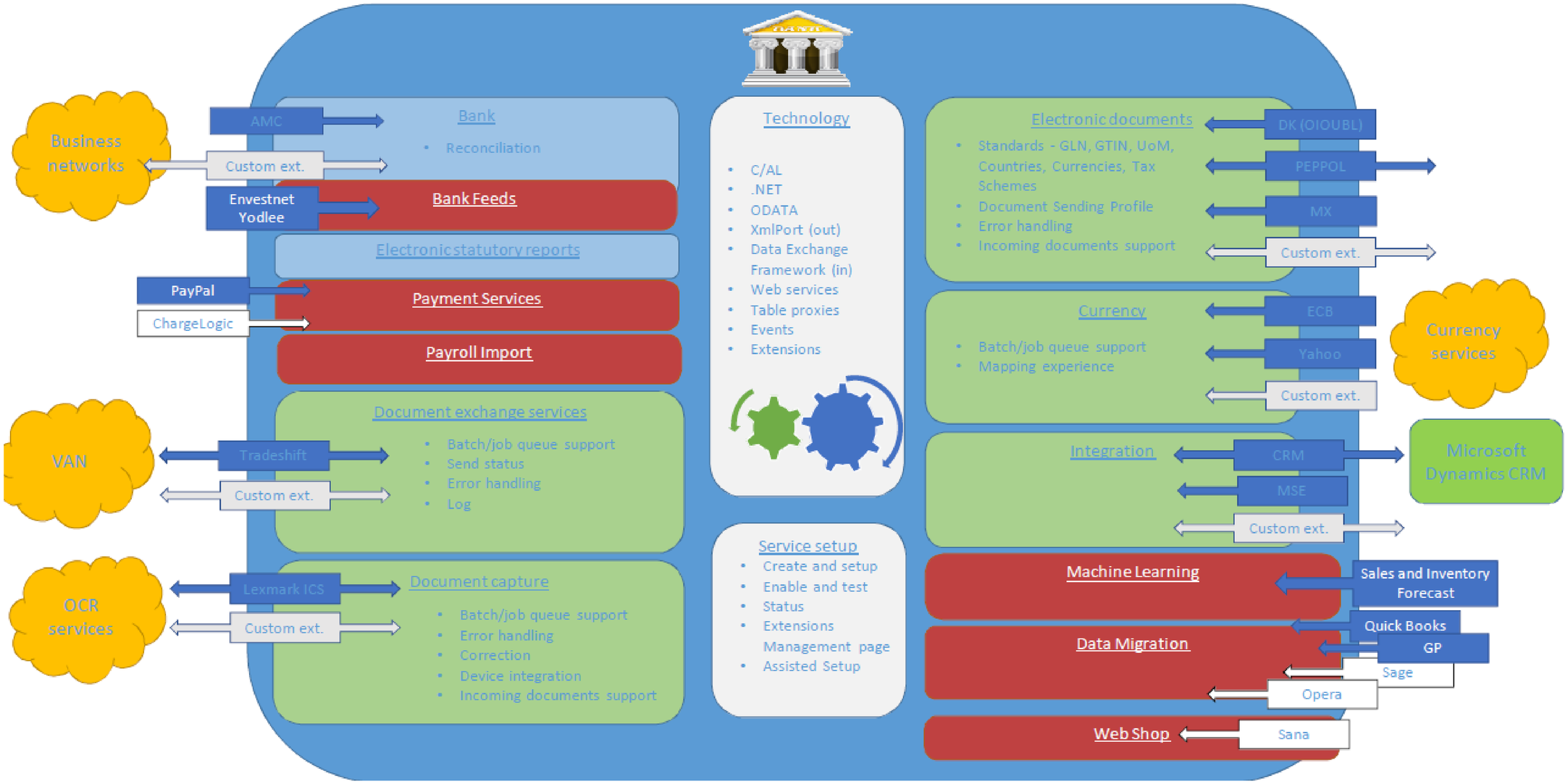
CREATING GREAT API'S

ANDERS LARSEN, NIKOLA KUKRIKA
MICROSOFT DEVELOPMENT CENTER COPENHAGEN

Techdays 2016

NAV TECH DAYS 2017
mibuso.com

NAV Integration to other services

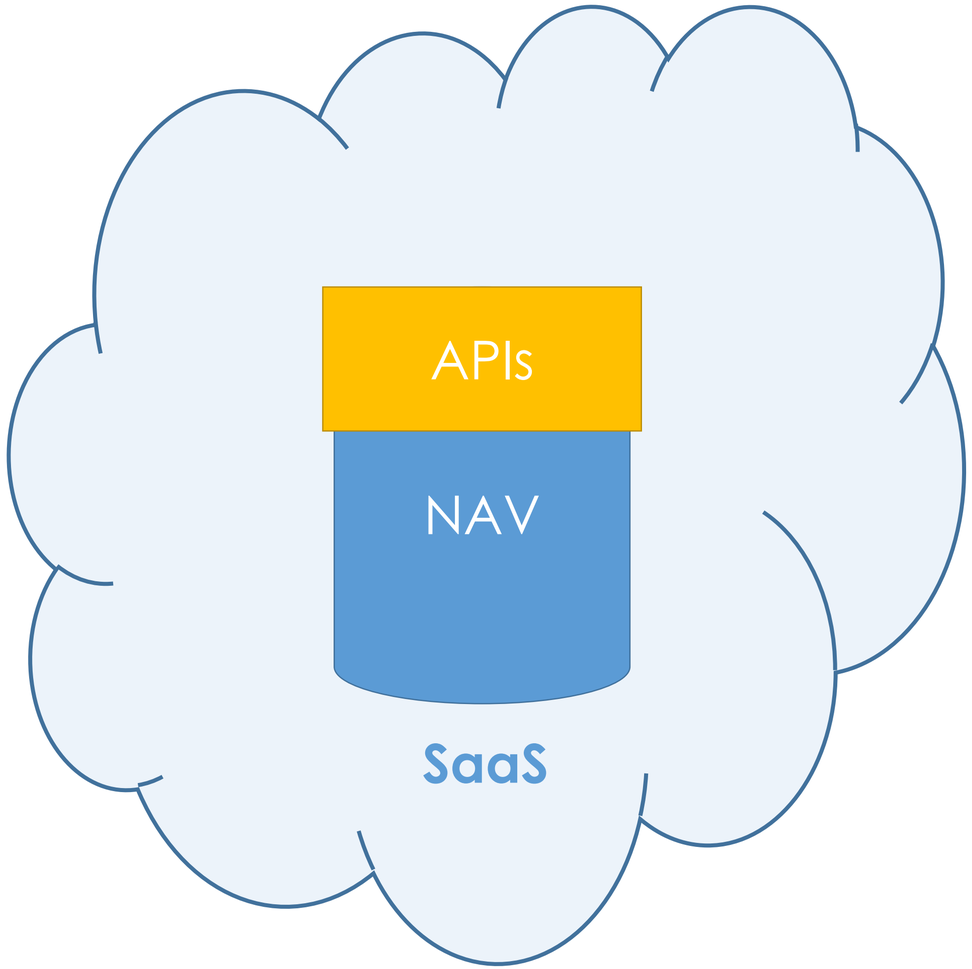


APIs – Allowing others to integrate to NAV

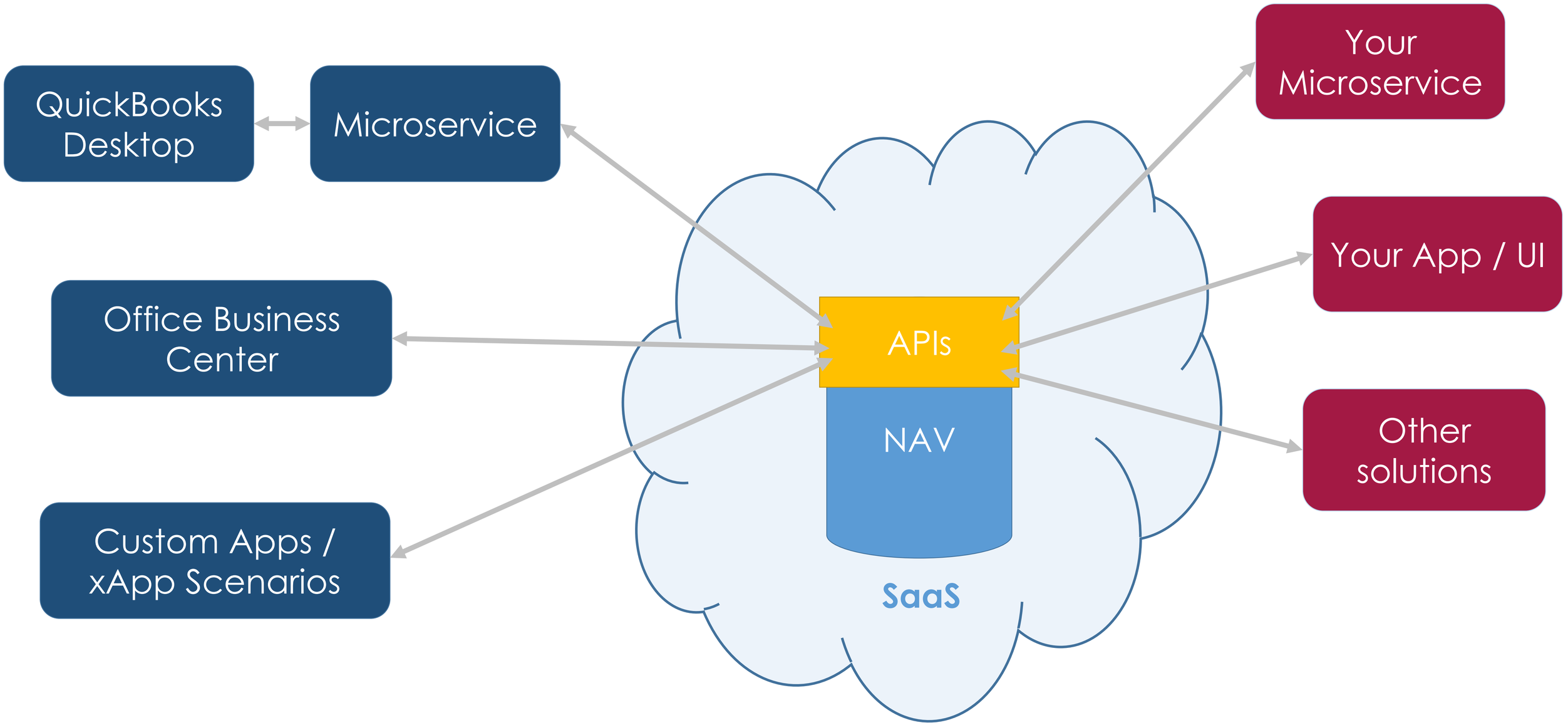
NAV TECH DAYS 2017
mibusa.com

API is a shield:

- **Stabile platform to build on**
  No breaking changes
- **Hides the complexity of the system via simple interface**



APIs – Allowing others to integrate to NAV

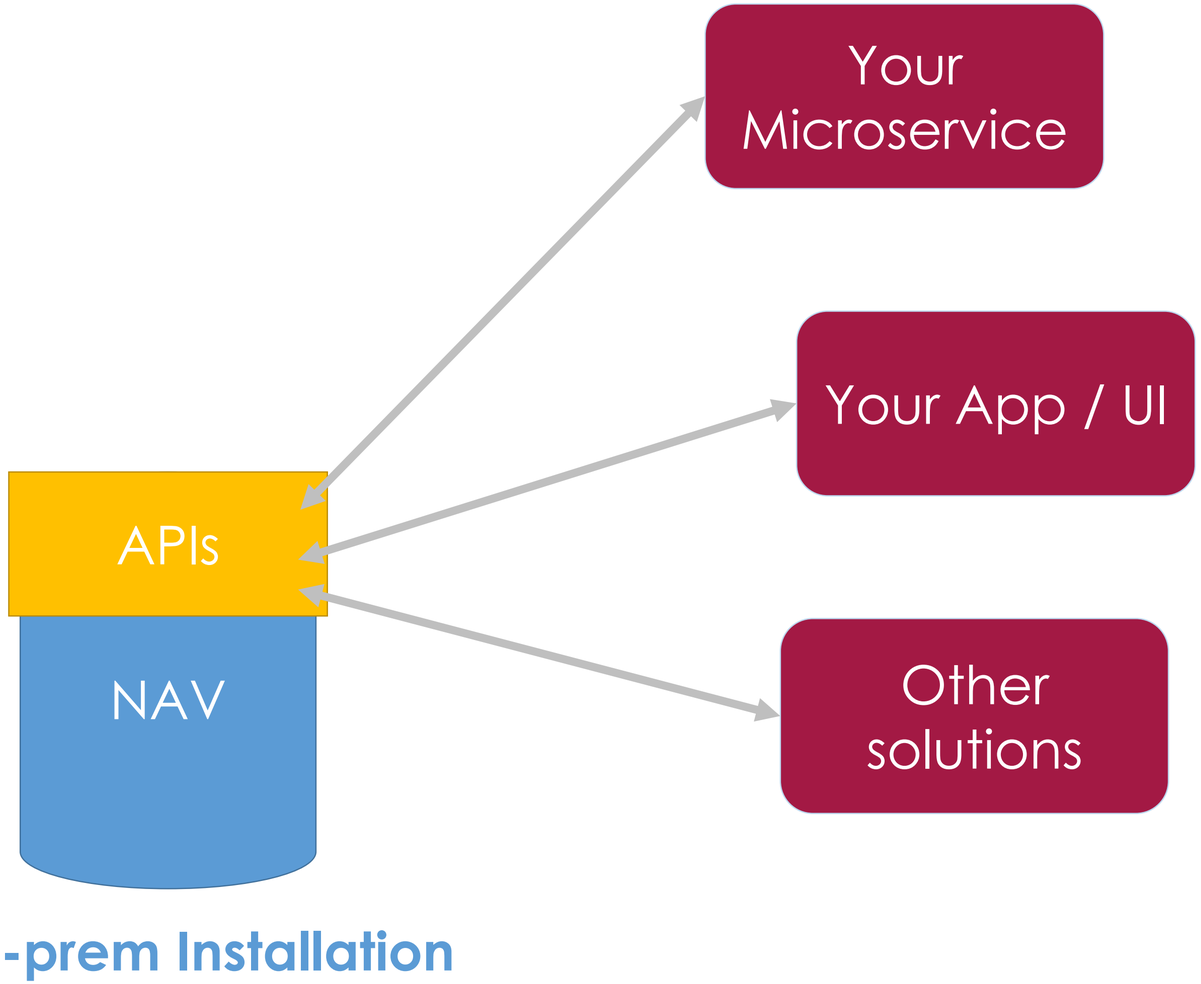


APIs – Allowing others to integrate to NAV

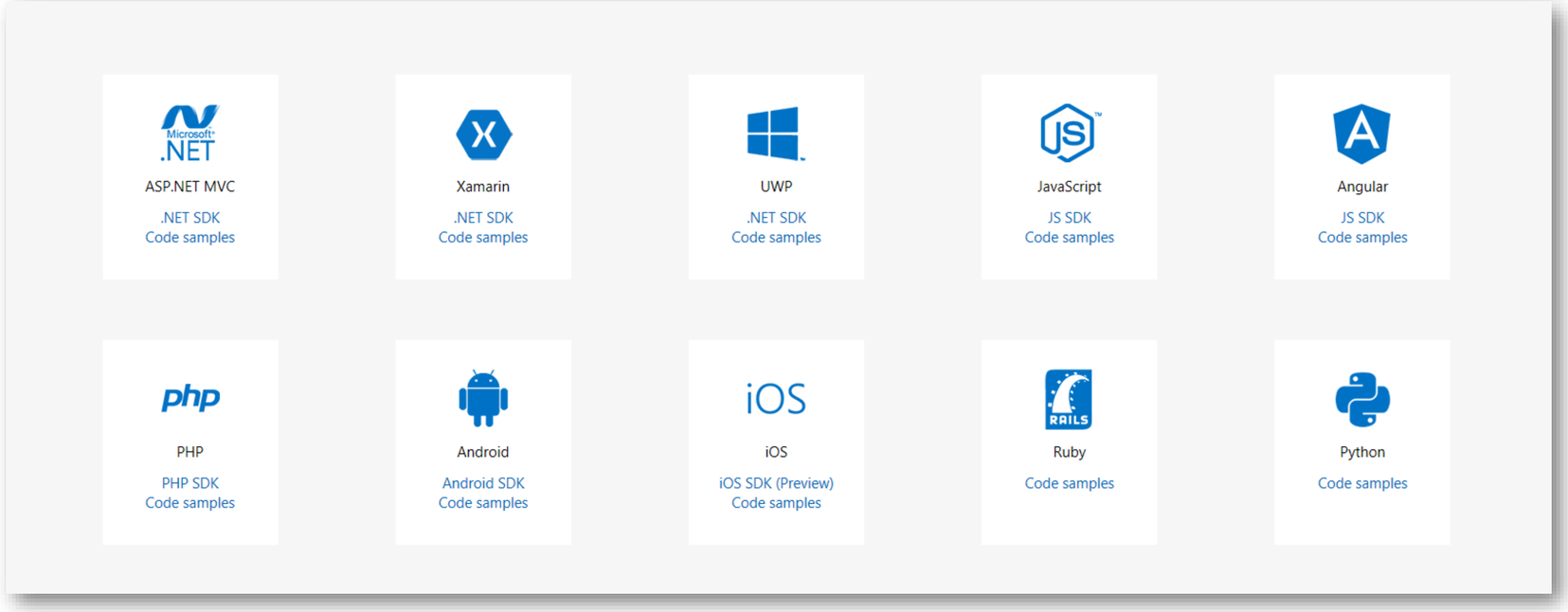
NAV TECH DAYS 2017
mibuso.com

Your integrations will work for both on-prem and SaaS

No Breaking Changes
No code merge
No upgrade



Development Platforms



AL

And more…

Session Objectives

Introduction to OData and Deep Dive in NAV Odata stack

Currently Available APIs

Endpoints - Graph, SaaS and on-prem

Building your APIs, Design Patterns and Future Plans

NAV TECH DAYS 2017
mibuso.com

Introduction to REST and OData

What is OData?

NAV TECH DAYS 2017
mibusa.com

REST (The term ***RE****presentational* ***S****tate* ***T****ransfer* was introduced and defined in 2000 by Roy Fielding in his doctoral dissertation)

Every service interprets and implements the standard

Stateless – Clients should assume no state exists

OData (Open Data Protocol)

OData is a standard defined on top of REST

Started in 2007 by Microsoft

Current is Version 4

Learning OData - Useful links

NAV TECH DAYS 2017
mibusa.com

Getting started:
http://www.odata.org/getting-started/

Standard:
http://docs.oasis-open.org/odata/odata/v4.0/odata-v4.0-part1-protocol.html

Coding guidelines:
https://github.com/Microsoft/api-guidelines/blob/vNext/Guidelines.md

What is OData in practice

For NAV we support 4 keywords:

GET - Read
POST - Insert
PATCH - Modify
DELETE - Delete

Data is transferred via JSON:
URL: https://api.invoicing.office.net/v1.0/api/beta/companies
Resonse:


```
{
  "id": "049f5c71-ca04-4cd2-bd05-7c8080195f89",
  "systemVersion": "19048",
  "name": "CRONUS UK Ltd.",
  "displayName": "CRONUS UK Ltd.",
  "businessProfileId": ""
}
```

API URLs - SaaS

Single endpoint:

Invoicing **-** https://api.invoicing.office.net/v1.0/api/beta/companies

Financials - https://api.financials.dynamics.net/v1.0/api/beta/companies

Single Endpoint Version

API Version

Direct URL:

https://tenantName.invoicing.office.net:7048/MS/api/beta/companies

https://tenantName.financials.dynamics.com:7048/MS/api/beta/companies

On Prem:

http://machineName:7048/DynamicsNAV110/api/beta/companies

You can fetch this from web services page

Versioning

Current version is beta (breaking changes are possible, though frowned upon)

Planning to go to V1 in next few months

Increase a version for breaking changes

Previous version must work as it is

Reading Data

GET https://api.invoicing.office.net/v1.0/api/beta/companies(fd401493-8ce3-492b-a727-22cef9664182)

Keys are canonical – must never change

We must keep track of deleted records for the given id

Most keys are represented as guids - standard, easy to generate unique values

Reading Data

Page size is specified in the server setting - "ODataServicesMaxPageSize"
1000 by default

Continuation link:

@odata.nextLink=
https://api.invoicing.office.net/v1.0/api/beta/companies%28629259f0-8dbf-445f-a260-ba5b1e02d410%29/salesInvoices?$skiptoken=be75ac21-427d-424f-b3d1-5070b8a3919a

Reading Data - Examples

NAV TECH DAYS 2017
mibusa.com

Reading a list of sales invoices:

https://api.invoicing.office.net/v1.0/api/beta/**companies(fd401493-8ce3-492b-a727-22cef9664182)/salesInvoices**

Specific sales invoice – set a key

https://api.invoicing.office.net/v1.0/api/beta/companies(fd401493-8ce3-492b-a727-22cef9664182)/**salesInvoices(f5c0ce0d-015d-4825-8174-1c55a3507dcc)**

Nested records

https://api.invoicing.office.net/v1.0/api/beta/companies(fd401493-8ce3-492b-a727-22cef9664182**)/salesInvoices(f5c0ce0d-015d-4825-8174-1c55a3507dcc)/salesInvoiceLines**

Complex keys (2 or more values):

https://api.invoicing.office.net/v1.0/api/beta/companies(fd401493-8ce3-492b-a727-22cef9664182)/salesInvoices(f5c0ce0d-015d-4825-8174-1c55a3507dcc)/**salesInvoiceLines(documentId=783326f8-38a3-4971-bc99-2cee0d11ffba,sequence=30000)**

Alternate key:

https://api.invoicing.office.net/v1.0/api/beta/companies(**name='My%20Company'**)

Creating data

POST https://api.invoicing.office.net/v1.0/api/beta/companies(fd401493-8ce3-492b-a727-22cef9664182)/customers

Add Content-Type: application/json to header
Add Autohorization
Add Body:

```
{
    "customerId":"46933f2b-92e9-4a3e-8bef-2cb8026675ef"
}
```

Modifying/Deleting data

Specify ETag to ensure concurrency

If-Match: W/"JzI4O0d3QUFBQUo3QlRjQU1BQXdBRFlBTkFBQUFBQUE2OzUwOTkyOTA7Jw=="

To ignore concurrency you can use:

If-Match: *

Concurrency is important in scenarios data is shown to the user, otherwise they may overwrite different data than they saw.

NAV OData Deep Dive

NAV OData Deep Dive

NAV TECH DAYS 2017
mibusa.com

1. Authentication
2. Metadata and Data Structures
3. $expand, deep inserts and $batch
4. $filter
5. Actions
6. Reading Binary Content (PDF, Image)
7. Error Handling
8. Selecting Language

Authentication

Two ways of authentication:

- Web service access key
- Bearer token

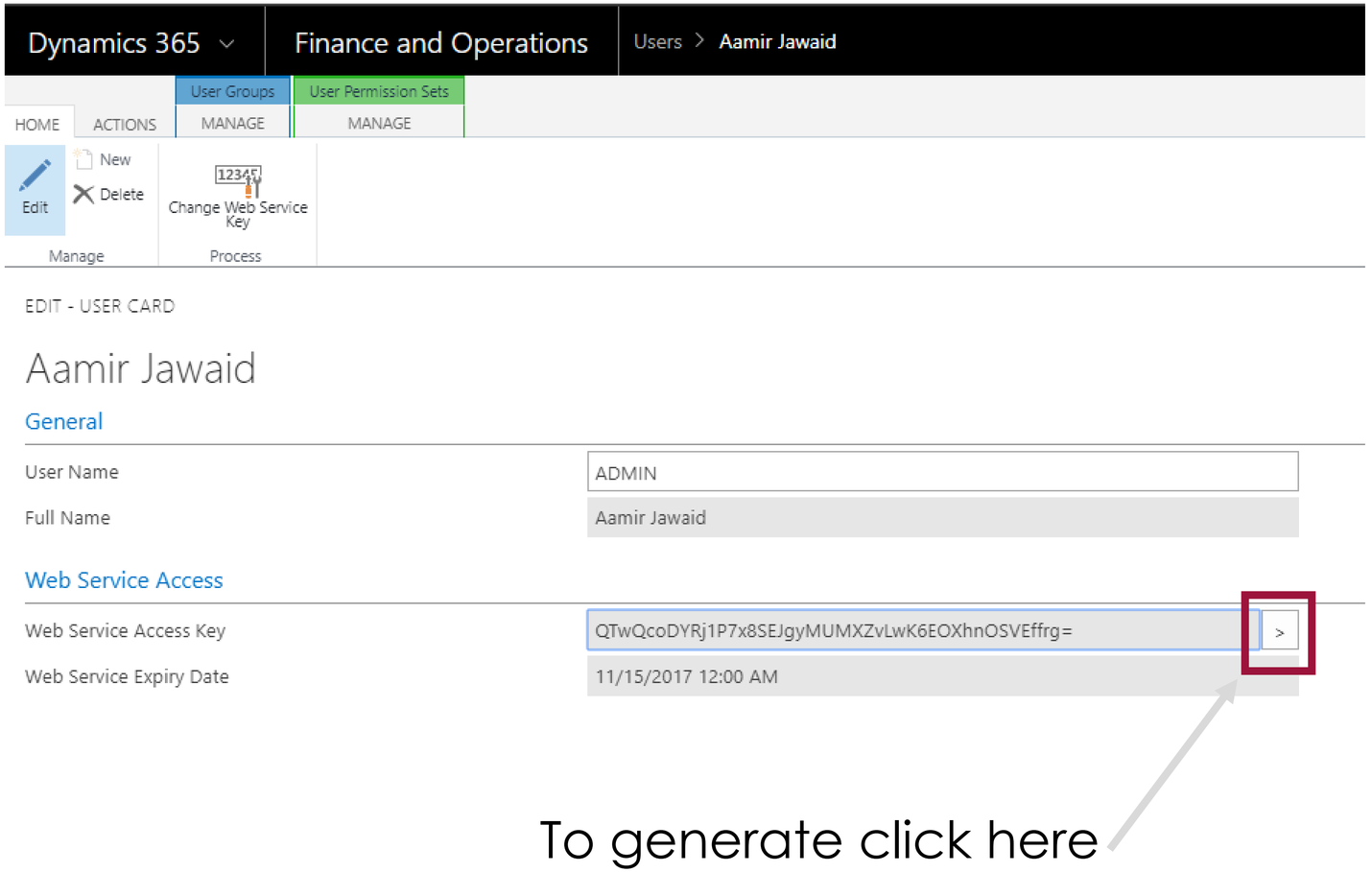
API is always running as an User

Web Service Access Key

1. On users card set web access key
2. Base64 encode the value username\webaccess key:
   ADMIN:fafdaiofdasfacaop=

3. In header set

Authorization: Basic **<base64EncodedKey>**



Bearer token

1. Obtain the token
2. Update the header with:
   Authorization: Bearer {token}

Token contains data about the user – e.g.
"unique_name":"admin@bigfishfitness.onmicrosoft.com"

Bearer token

Things needed:

1. Authority
2. Application ID
3. Resource
4. User credentials

```
3 references
public async Task<AuthenticationResult> GetUserToken()
{
    var authenticationContext =
        new AuthenticationContext($"{txtAuthority.Text}/{TenantName}");

    var userToken = await authenticationContext.AcquireTokenAsync(
        txtResource.Text,
        txtClientApplicationId.Text,
        new Uri(txtClientAppRedirectUri.Text),
        new PlatformParameters(PromptBehavior.Auto),
        new UserIdentifier(txtUserPrincipalName.Text, UserIdentifierType.RequiredDisplayableId));

    return userToken;
}
```

Get access tokens article:
https://developer.microsoft.com/en-us/graph/docs/concepts/auth_overview

Tokens vs Web Access Keys?

NAV TECH DAYS 2017
mibusa.com

1. Tokens have simpler authentication
   - If user is logged in you can reuse the login

2. Tokens can use Single Endpoint:
   - Simpler setup in the app – we can determine the tenant
   - If tenant does not exist it, we will provision it within 40s

3. Tokens expire – easier management

4. Web Access Keys are good for prototyping

Metadata

“There are two ways of learning about OData service:

1. Documentation
2. $metadata”

Metadata

NAV TECH DAYS 2017
mibuso.com

URL - https://api.invoicing.office.net/v1.0/api/beta/$metadata

Example response:

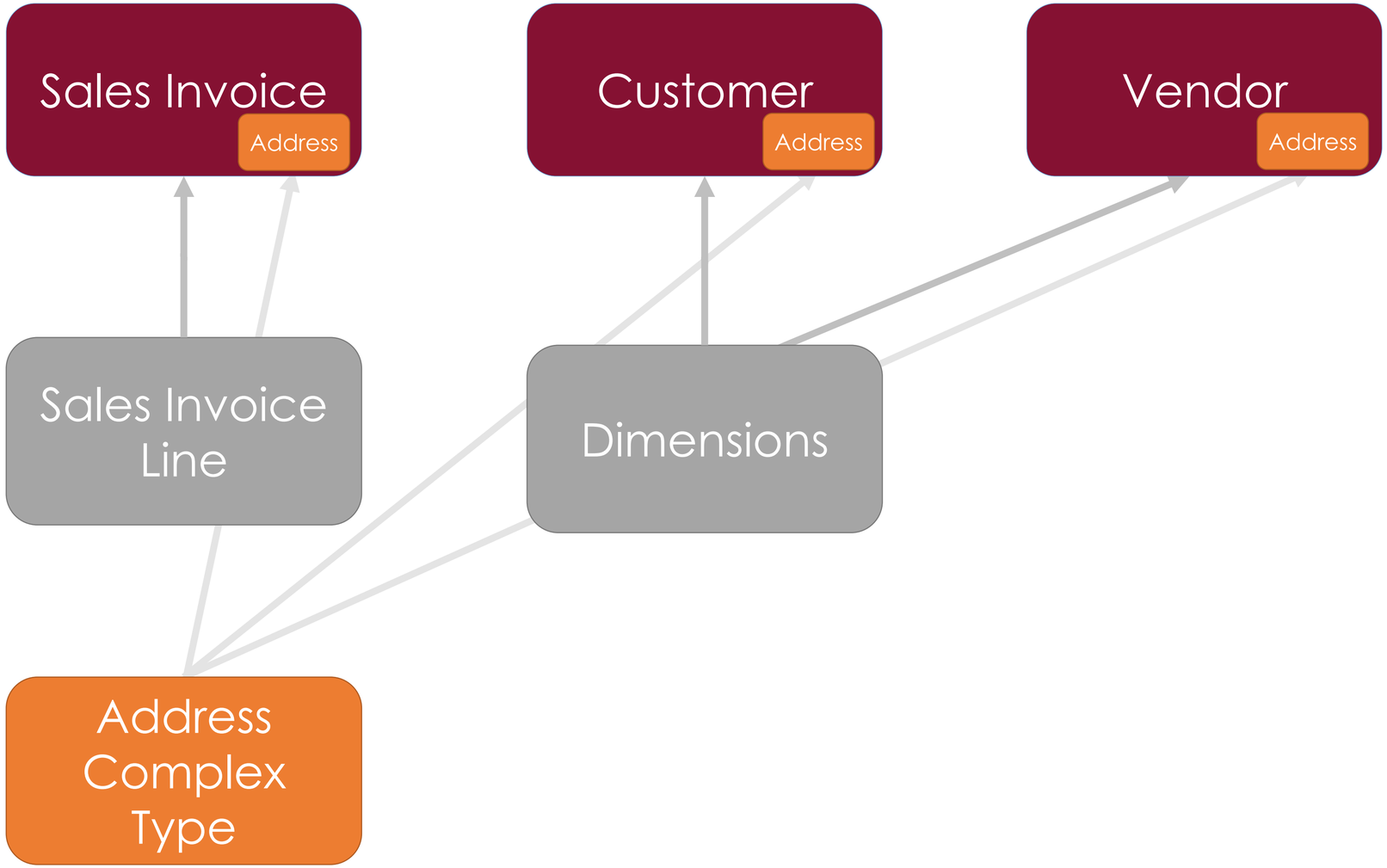
```
<edmx:Edmx Version="4.0" xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx">
  <edmx:DataServices>
    <Schema Namespace="Microsoft.NAV" xmlns="http://docs.oasis-open.org/odata/ns/edm">
      <ComplexType Name="PostalAddress">
        <Property Name="street" Type="Edm.String" MaxLength="102" />
        <Property Name="city" Type="Edm.String" MaxLength="30" />
        <Property Name="state" Type="Edm.String" MaxLength="30" />
        <Property Name="countryLetterCode" Type="Edm.String" MaxLength="10" />
        <Property Name="postalCode" Type="Edm.String" MaxLength="20" />
      </ComplexType>
      <ComplexType Name="PaymentMethod">
      <ComplexType Name="PaymentTermsType">
      <EntityType Name="generalLedgerEntry">
        <Key>
          <PropertyRef Name="id" />
        </Key>
        <Property Name="id" Type="Edm.Int32" Nullable="false" />
        <Property Name="postingDate" Type="Edm.Date" />
        <Property Name="documentNumber" Type="Edm.String" MaxLength="20" />
        <Property Name="documentType" Type="Edm.String" />
        <Property Name="accountId" Type="Edm.Guid" />
        <Property Name="accountNumber" Type="Edm.String" MaxLength="20" />
        <Property Name="description" Type="Edm.String" MaxLength="50" />
        <Property Name="debitAmount" Type="Edm.Decimal" Scale="Variable" />
        <Property Name="creditAmount" Type="Edm.Decimal" Scale="Variable" />
        <Property Name="lastModifiedDateTime" Type="Edm.DateTimeOffset" />
        <NavigationProperty Name="accountIdLink" Type="Microsoft.NAV.account" ContainsTarget="true" />
        <NavigationProperty Name="accountNumberLink" Type="Microsoft.NAV.account" ContainsTarget="true" />
      </EntityType>
        <Property Name="displayName" Type="Edm.String" MaxLength="50" />
        <Property Name="category" Type="Edm.String" />
        <Property Name="subCategory" Type="Edm.String" MaxLength="80" />
        <Property Name="blocked" Type="Edm.Boolean" />
        <Property Name="lastModifiedDateTime" Type="Edm.DateTimeOffset" />
      </EntityType>
      <EntityType Name="agedAccountsPayable">
```

Entities / Sub-entities / Complex Types

Entities

Sub-entities
Cannot live
without patent

Complex Type
Reusable struct
Lives on main record



Design Question – Sub-Entity or Complex Type

Performance Impact – Complex type is always loaded

Do we need to access the sub-entity directly without main entity?

Complex type is updated in a single transaction with main entity

Navigational Properties

Contains a link to a related record

Can be inserted - Deep Insert

Can be expanded with $expand

```
<EntityType Name="salesQuotes">
  <Key>
    <PropertyRef Name="id" />
  </Key>
  <Property Name="id" Type="Edm.Guid" Nullable="false" />
  <Property Name="number" Type="Edm.String" MaxLength="20" />
  <Property Name="documentDate" Type="Edm.Date" />
  <Property Name="dueDate" 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="customerName" Type="Edm.String" MaxLength="50" />
  <Property Name="billingPostalAddress" Type="Microsoft.NAV.PostalAddress" />
  <Property Name="currencyCode" Type="Edm.String" />
  <Property Name="paymentTerms" Type="Edm.String" MaxLength="10" />
  <Property Name="shipmentMethod" Type="Edm.String" MaxLength="10" />
  <Property Name="salesperson" Type="Edm.String" MaxLength="20" />
  <Property Name="discountAmount" Type="Edm.Decimal" Scale="Variable" />
  <Property Name="totalAmountExcludingTax" Type="Edm.Decimal" Scale="Variable" />
  <Property Name="totalTaxAmount" Type="Edm.Decimal" Scale="Variable" />
  <Property Name="totalAmountIncludingTax" Type="Edm.Decimal" Scale="Variable" />
  <Property Name="status" Type="Edm.String" />
  <Property Name="validUntilDate" Type="Edm.Date" />
  <Property Name="acceptedDate" Type="Edm.Date" />
  <Property Name="lastModifiedDateTime" Type="Edm.DateTimeOffset" />
  <NavigationProperty Name="salesQuoteLines" Type="Collection(Microsoft.NAV.salesQuoteLines)" ContainsTarget="true" />
  <NavigationProperty Name="customerIdLink" Type="Microsoft.NAV.customers" ContainsTarget="true" />
  <NavigationProperty Name="customerNumberLink" Type="Microsoft.NAV.customers" ContainsTarget="true" />
  <NavigationProperty Name="customerNameLink" Type="Microsoft.NAV.customers" ContainsTarget="true" />
  <NavigationProperty Name="paymentTermsLink" Type="Microsoft.NAV.paymentTerms" ContainsTarget="true" />
  <NavigationProperty Name="shipmentMethodLink" Type="Microsoft.NAV.shipmentMethods" ContainsTarget="true" />
</EntityType>
```

$expand

Includes values from navigational properties in the response

URL:

…/api/beta/companies(629259f0-8dbf-445f-a260-ba5b1e02d410)/salesInvoices?$expand=salesInvoiceLines

Deep inserts

Insert an entity with sub-entities and navigational properties with a single call

It runs as a single transaction – all or nothing
Only Insert – there is no Deep Modify defined by standard

Example – inserting an Invoice with lines

**$batch -** /api/beta/$batch

Only a single write transaction per change-set

Many write transactions

Don’t batch up multiple write transactions

```
--batch_abc123
Content-Type: multipart/mixed; boundary=changeset_abc

--changeset_abc
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Id: abc1234

POST companies(b4dc4011-9519-4d01-8649-d3bcfdc00cfa)/salesInvoices HTTP/1.1
Content-Type: application/json

{ "customerNumber": "40000","id": "ce074c6d-d219-44fc-8341-ea2194ab3fb1" }

--changeset_abc--

--batch_abc123
Content-Type: multipart/mixed; boundary=changeset_abcd

--changeset_abcd
Content-Type: application/http
Content-Transfer-Encoding: binary
Content-Id: abc12345

POST companies(b4dc4011-9519-4d01-8649-d3bcfdc00cfa)/salesInvoices(ce074c6d-d219-44fc-8341-
ea2194ab3fb1)/salesInvoiceLines HTTP/1.1
Content-Type: application/json

{
  "itemNumber": "1952-W",
  "quantity": 5
}

--changeset_abcd--
```

$filter

NAV TECH DAYS 2017
mibusa.com

Filter behaves the same as NAV filter in the UI
Must work on a field from a table – calculated fields are ignored
No unions supported

Operator	Description	Example
Logical Operators		
Eq	Equal	/Suppliers?$filter=Address/City eq 'Redmond'
Ne	Not equal	/Suppliers?$filter=Address/City ne 'London'
Gt	Greater than	/Products?$filter=Price gt 20
Ge	Greater than or equal	/Products?$filter=Price ge 10
Lt	Less than	/Products?$filter=Price lt 20
Le	Less than or equal	/Products?$filter=Price le 100
And	Logical and	/Products?$filter=Price le 200 and Price gt 3.5
Or	Logical or	/Products?$filter=Price le 3.5 or Price gt 200
Not	Logical negation	/Products?$filter=not endswith(Description,'milk')
Arithmetic Operators		
Add	Addition	/Products?$filter=Price add 5 gt 10
Sub	Subtraction	/Products?$filter=Price sub 5 gt 10
Mul	Multiplication	/Products?$filter=Price mul 2 gt 2000
Div	Division	/Products?$filter=Price div 2 gt 4
Mod	Modulo	/Products?$filter=Price mod 2 eq 0
Grouping Operators		
( )	Precedence grouping	/Products?$filter=(Price sub 5) gt 10

Reading binary content – images, pdf…

JSON contains direct link to the image:

picture@odata.mediaReadLink= https://api.financials.dynamics.net/v1.0/ api/beta/companies(629259f0-8dbf-445f-a260-ba5b1e02d410)/companyInformation(f26ce386-1497-4b85-95a0-6165526f0b43)/picture

picture@odata.mediaReadLink= https://api.financials.dynamics.net/v1.0/ api/beta/companies(629259f0-8dbf-445f-a260-ba5b1e02d410)/employees(5daf4ae8-e5da-4da1-926c-0b6704a0a22b)/picture

Bound Actions

NAV TECH DAYS 2017
mibuso.com

Must run on a single entity
Metadata specifies to which entity an action belongs

```
Action [ Name=Post IsBound=true ]
    <Parameter Name="bindingParameter" Type="Microsoft.NAV.salesInvoice" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
```

Invoke a simple POST request without body
POST …/api/beta/companies(<id>)/salesInvoices(<id>)/**Microsoft.NAV.**Post

Do not specify Content-Type

No parameters are possible
Can only return an entity

Bound Actions

In the header the response may contain link to the new entity:

POST …/salesQuotes(eb6cd3d5-2c3c-4122-962a-95a75f8d3c70)/Microsoft.NAV.MakeInvoice

Will return location of an invoice:

Location: http://navdevvm-0124.europe.corp.microsoft.com:7047/Navision_MainGit/api/beta/companies(629259f0-8dbf-445f-a260-ba5b1e02d410)/salesInvoices(86c7073c-9beb-45dc-af57-9943426de581)

Unbound Actions

NAV TECH DAYS 2017
mibuso.com

OData V4 concept – not supported by API
Expose the codeunit as a Web Service
Can take parameters and return data
They are on the root level in metadata

```
Action [ Name=CompanySetupService_ConfigureCompany ]
    <Parameter Name="name" Type="Edm.String" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
    <Parameter Name="address" Type="Edm.String" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
    <Parameter Name="address2" Type="Edm.String" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
    <Parameter Name="city" Type="Edm.String" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
    <Parameter Name="county" Type="Edm.String" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
    <Parameter Name="postCode" Type="Edm.String" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
    <Parameter Name="countryCode" Type="Edm.String" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
    <Parameter Name="phoneNo" Type="Edm.String" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
    <ReturnType Type="Edm.Boolean" xmlns="http://docs.oasis-open.org/odata/ns/edm" />
Action [ Name=EncryptedKeyValue_Insert ]
Action [ Name=EncryptedKeyValue_Update ]
Action [ Name=EncryptedKeyValue_Remove ]
Action [ Name=EncryptedKeyValue_Cleanup ]
Action [ Name=ExchangeServiceSetup_Store ]
Action [ Name=WorkflowActionResponse_Approve ]
Action [ Name=WorkflowActionResponse_Reject ]
```

Design Question – To Action or just do PATCH?

Should this be an Action?

- Post an Invoice **- Yes**
- Accept a quote - flips accepted to true and sets accepted date. **- No, it should be PATCH**

When to use Actions VS PATCH?

- PATCH should be used if you can PATCH it back. PATCH should not have major side effects on other entities
- Action should be used for irreversible changes and when many records will be updated.

Error Handling

Error Code Categories

Category	Description	Resolution
BadRequest_*	Will typically be an error in the forming of the request or an error accessing the service.	Resolve the bug in the forming of the request.
Authentication_*	An error authenticating to the service.	Attempt to use different credentials.
Authorization_*	The authenticated identity does not have the correct permissions.	Attempt operation using different credentials.
Internal_*	Typically this is an internal error in the application on the server or data integrity issue. For example, the Dynamics NAV instance cannot communicate with the SQL Server.	Attempt the operation again. Resolve data issues.
Application_*	Typically an application logic error.	Request is made again with updated data.

User Friendly Error Codes

Exception Type	Error Message	Error Code
NavCSideDataException	There is no Cust. Ledger Entry within the filter.	Internal_DataNotFoundFilter
NavCSideRecordNotFoundException	The Acc. Sched. KPI Web srv. Setup does not exist. Identification fields and values: Primary Key=''	Internal_RecordNotFound
NavCSideValidateTableRelationException	The field Account No. of table Gen. Journal Line contains a value (ABL001) that cannot be found in the related table (Vendor).	Internal_InvalidTableRelation
NavCSideDuplicateKeyException	The Attachment Entity Buffer already exists. Identification fields and values: Document Id='{DAC3AB2F-5FEA-4AD2-A663-EF832F270A7B}'	Internal_EntityWithSameKeyExists
NavNCLDialogException	You cannot delete Item 1000 because there is at least one outstanding Sales Quote that includes this item.	Application_DialogException
	You are not allowed to apply and post an entry to an entry with an earlier posting date.	Application_DialogException
	The "amount" should be a negative number.	Application_DialogException
NavNCLEvaluateException	The value "Depreciation" cannot be evaluated into type Option.	Application_EvaluateException
NavNCLStringLengthExceededException	The length of the string is 57, but it must be less than or equal to 50 characters. Value: JACKSBORO PUMP & SPECIALTY BRIDGEPORT PUMP & SUPPLY, INC.	Application_StringExceededLength
NavFilterException	The filter "='SELLACRE24_W%2FOORINGS'" is not valid for the No. field on the Item table. The length of the string is 22, but it must be less than or equal to 20 characters.	Application_FilterErrorException
NavTestFieldException	Balance must be equal to '0' in G/L Account: No.=10100. Current value is '1,638.4'. Customer Posting Group must have a value in Customer: No.=C00690. It cannot be zero or empty.	Application_FieldValidationException

Full list of Error Codes

NAV TECH DAYS 2017
mibuso.com

https://msdn.microsoft.com/en-us/dynamics-nav/fin-graph/dynamics_error_codes

Selecting Language

To select a language put:
Accept-Language: **<languageCode>** (e.g. fr-Ca)

This will be equal to GLOBALLANGUAGE(3082)

Text constants and error messages will use that locale if it exists

It is only during that API call

Differences - API VS OData V4 endpoint

NAV TECH DAYS 2017
mibusa.com

APIs – auto published, not possible to change via extensions
You can publish ODataV4 Web Services from extensions

URL is Different, Company vs Companies

Company endpoint is different, no alternate keys on OData

Unbound actions are only on ODataV4

Action Namespace for ODataV4 is NAV.<actionName> for API is Microsoft.NAV.<actionName>

NAV TECH DAYS 2017
mibuso.com

Demo

NAV TECH DAYS 2017
mibuso.com

APIs out of the Box

APIs out of the Box

Kept it simple

Inspired by APIs such as QuickBooks Online, Xero, Sage etc.

Alignment with other Microsoft APIs (naming and structure)

Hides any internal NAV fields

Will be expanded to include more, balancing simplicity

Important considerations

Breaking changes in beta are frowned upon

When API gets a version it stays like this – We must be able to support it “forever”

No metadata difference across tenants / countries – all fields must be exposed, exception will be thrown if unsupported field is used

Not possible to modify/extend – use OData V4 endpoint for extensions

APIs out of the Box

NAV TECH DAYS 2017
mibusa.com

Customers API

- Example of fields that we expose
  - No. - number
  - Name - displayName
  - Address – following office format

- Example of fields that we do not expose
  - Customer Posting Group
  - Invoice Disc. Code

APIs out of the Box

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				

API Documentation

https://msdn.microsoft.com/en-us/dynamics-nav/fin-graph/resources/dynamics_overview

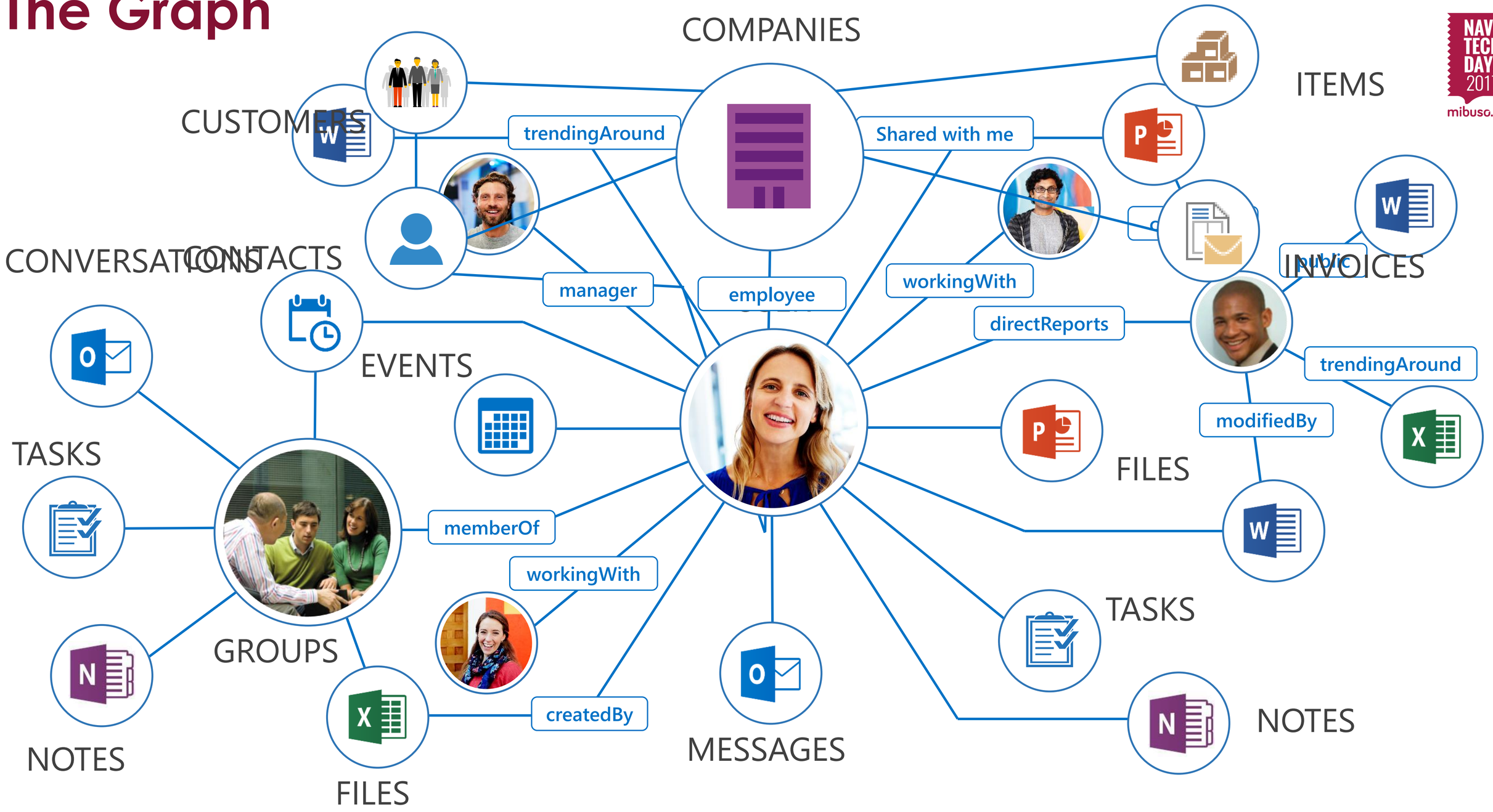
Code - object range from 5470-5508

The Graph

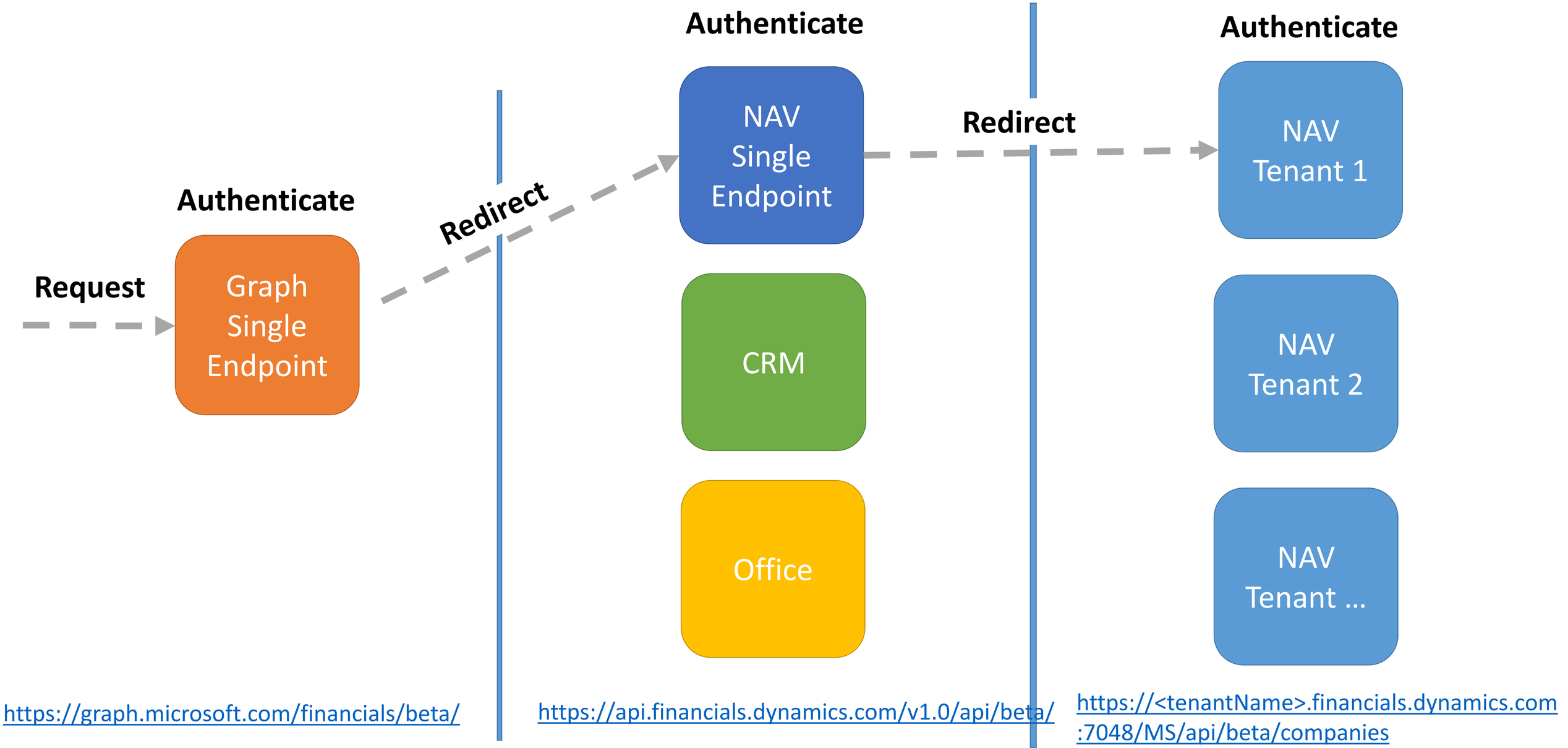
The Graph

The one endpoint to rule them all
Get started at: graph.microsoft.io

The Graph



Architecture overview



https://graph.microsoft.com/financials/beta/

https://api.financials.dynamics.com/v1.0/api/beta/

https://<tenantName>.financials.dynamics.com:7048/MS/api/beta/companies

Endpoints Overview

NAV TECH DAYS 2017
mibuso.com

	Dynamics 365 for Financials and Operations, Business Edition (Online)			Microsoft Dynamics NAV 2018 (On-prem)
Means of connection	***Microsoft Graph***	***Common endpoint service***	***Direct tenant***	***Direct installation***
Usage	Production	Production	Rapid development and testing only	Production
End-point	https://graph.microsoft.com/financials/beta/	https://api.financials.dynamics.com	https://<tenant url>:7948/MS/api/<API version>/ Example: https://contoso.com:7048/api/beta	OData base url in installation https://<base URL>:<port>/v1.0/api/<API version>/ Example: https://nav.contoso.com:7048/v1.0/api/beta/ Must be exposed through firewall
Availability	Always enabled			Disabled by default Must be enabled by admin
Authentication	Azure Active Directory (AAD)		Basic authentication Azure Active Directory (AAD)	Basic authentication Username and web service access key as password
API/data access	Based on user's permissions			
API update cycle	Monthly			Hotfixes installed by partner
Development instance	Sign up for tenant at http://portal.microsoft.com			Get Docker instance

Graph design review

All APIs must confirm to the same standard
Same naming and behavior across endpoints

NAV TECH DAYS 2017
mibuso.com

Enabling APIs On Prem

Enabling APIs On Prem

API is disabled by default.

To enable API endpoint:

1. Make sure OData web services are enabled (follow existing guidelines)
2. In the NAV administration console - check Enable API Services checkbox
3. Click the Integrate APIs button in the API Setup page (may be a fairly long running process)
4. Optional – Setup templates

NAV TECH DAYS 2017
mibuso.com

How to Develop your APIs

Build your own API

Property on Page
Complex Type
Bound actions

NAV TECH DAYS 2017
mibuso.com

Design “Patterns”

Design “Patterns”

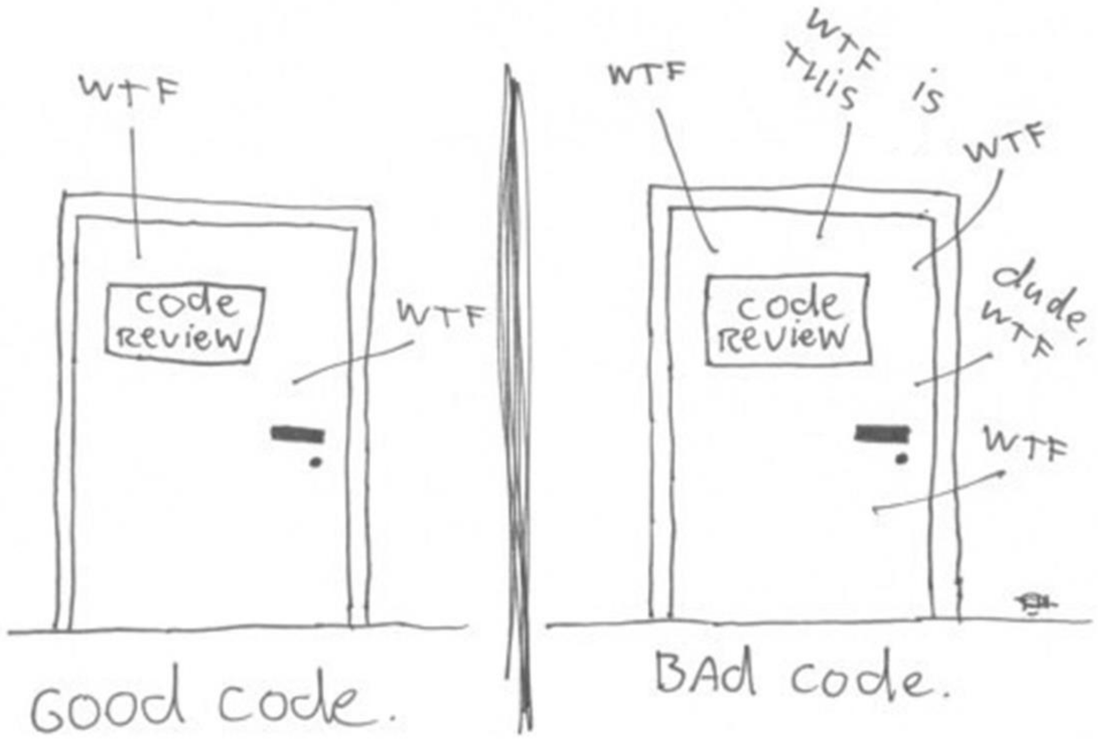
1. Shaddow Table
2. Register Field Set
3. Integration records
4. Temporary Tables

Commonly Made mistakes

How do you measure code quality?

The only valid measurement of code quality:

WTFs/Minute

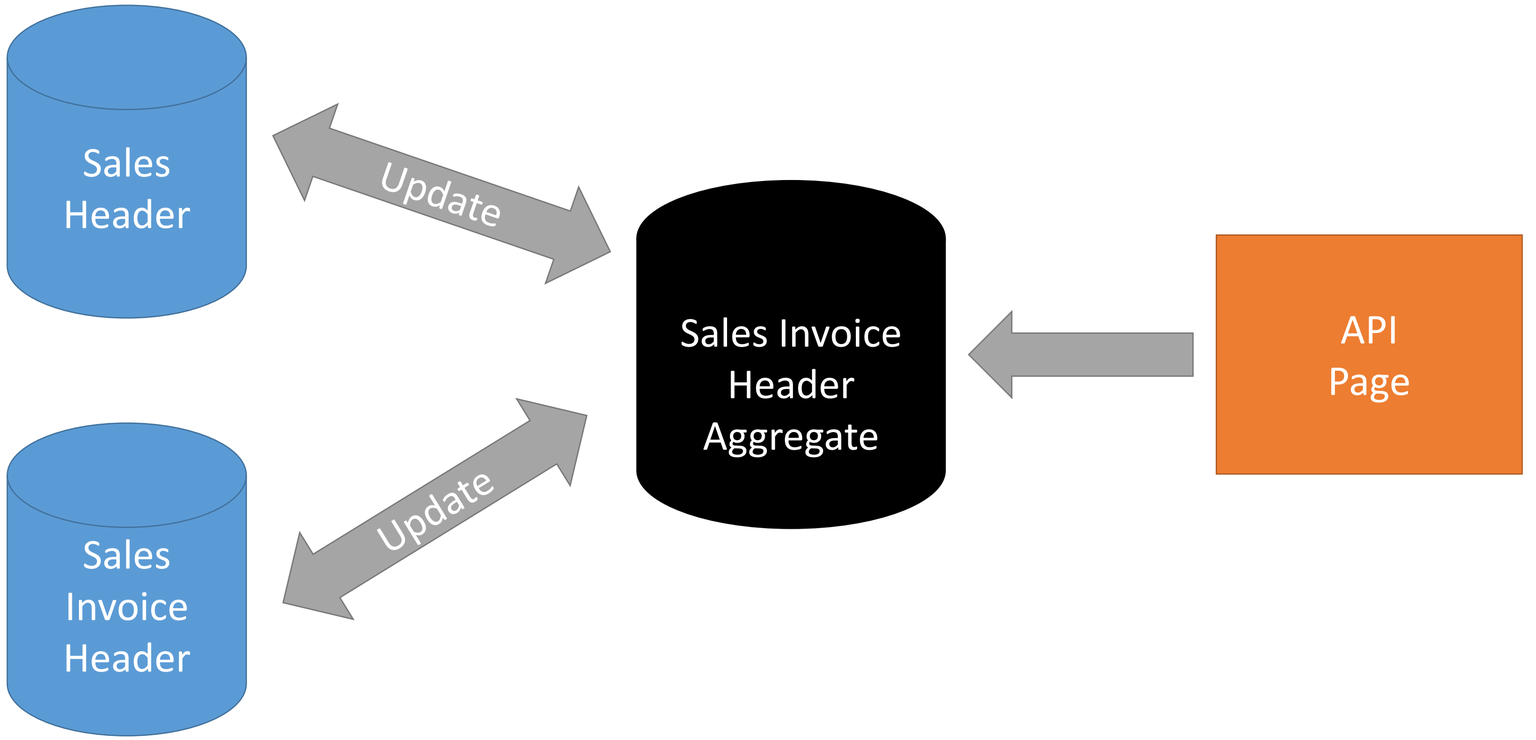


What The F

Why The F – component level

Who The F – ultimate level

Shadow Table



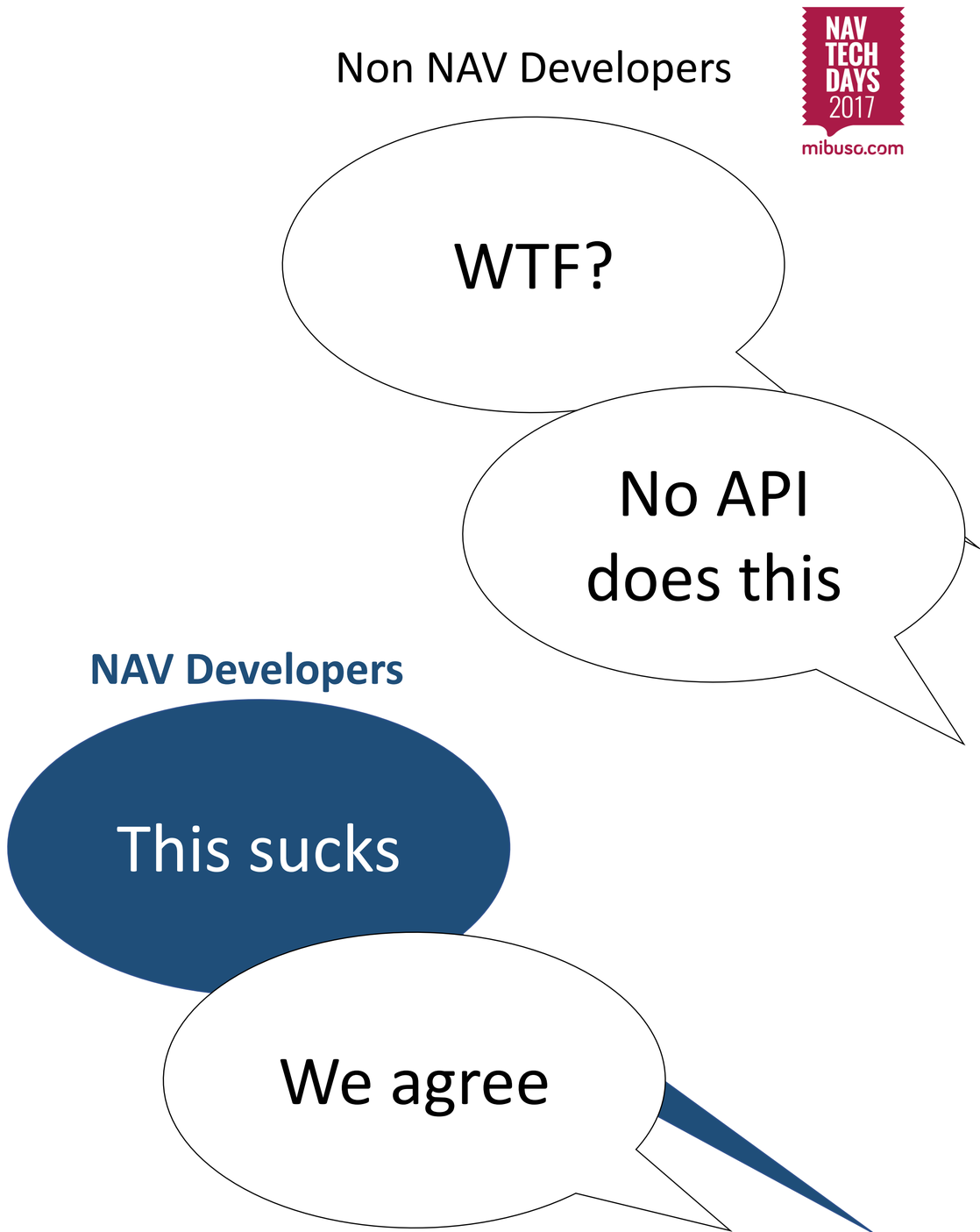
Shadow Table

Problem

Data is split in two tables:
Sales Header and Sales Invoice Header

Split APIs like:
draftSalesInvoices
salesInvoices

Flow:
First check if it is in drafts,
If not double check in salesInvoices
If not then it is probably deleted



Shadow Table

Lets base API on a temp table - Horrible Idea

Rule - Never base API page on temporary table if a call can return large number of records

Performance – you have to load all records no matter what
$filter – filtering is problematic
Continuation tokens (paging) is problematic

Sales Invoices – not OK to base on TEMP
Tax Setup – OK to base on TEMP

Shadow Table

Problem

Cannot split APIs when data is in two tables
Calculations are expensive – 30ms to calculate totals * 1000 invoices = 30s
API call must return within 200 ms (most cases)

Solution

Precompute the values in a shadow table

Shadow Table – Update from main tables

NAV TECH DAYS 2017
mibuso.com

Subscribe to global events and update on every change
Don't fire on Temp and if API is not enabled

```
LOCAL [EventSubscriber] OnAfterInsertSalesHeader(VAR Rec : Record "Sales Header";RunTrigger : Boolean)
IF NOT CheckValidRecord(Rec) OR (NOT GraphMgtGeneralTools.IsApiEnabled) THEN
  EXIT;

InsertOrModifyFromSalesHeader(Rec);

LOCAL [EventSubscriber] OnAfterModifySalesHeader(VAR Rec : Record "Sales Header";VAR xRec : Record "Sales
IF NOT CheckValidRecord(Rec) OR (NOT GraphMgtGeneralTools.IsApiEnabled) THEN
  EXIT;

InsertOrModifyFromSalesHeader(Rec);

LOCAL [EventSubscriber] OnAfterDeleteSalesHeader(VAR Rec : Record "Sales Header";RunTrigger : Boolean)
IF NOT CheckValidRecord(Rec) OR (NOT GraphMgtGeneralTools.IsApiEnabled) THEN
  EXIT;
```

Shadow Table – Update from Shadow Table


```
OnInsertRecord(BelowxRec : Boolean) : Boolean
CheckCustomerSpecified;
ProcessBillingPostalAddress;

SalesInvoiceAggregator.PropagateOnInsert(Rec,TempFieldBuffer);
SetDates;

UpdateDiscount;

SetCalculatedFields;

EXIT(FALSE);
```

Push the changes back to the main record in the OnInsert, OnModify and OnDelete triggers

Update to shadow must only happen via main table, so exit false from the trigger

Shadow Table - Fields

NAV TECH DAYS 2017
mibuso.com

All fields over 9000 are API specific

Everything under 9000 belongs to Sales Header / Sales Inv. Header and must match exactly (Enforced by test)

There should not be many calculated fields exposed

✔	100	External Document No.	Code	35
✔	114	Tax Area Code	Code	20
✔	115	Tax Liable	Boolean	
✔	116	VAT Bus. Posting Group	Code	20
✔	121	Invoice Discount Calculation	Option	
✔	122	Invoice Discount Value	Decimal	
✔	167	Last Email Sent Status	Option	
✔	1304	Cust. Ledger Entry No.	Integer	
✔	1305	Invoice Discount Amount	Decimal	
✔	5052	Sell-to Contact No.	Code	20
✔	8000	Id	GUID	
✔	9600	Total Tax Amount	Decimal	
✔	9601	Status	Option	
✔	9602	Posted	Boolean	
✔	9603	Subtotal Amount	Decimal	
✔	9624	Discount Applied Before Tax	Boolean	
✔	9630	Last Modified Date Time	DateTime	
✔	9631	Customer Id	GUID	
✔	9632	Order Id	GUID	
✔	9633	Contact Graph Id	Text	250

Shadow Table - Code

Rule - Only update Shadow table form listeners. If you start updating other records, you are risking another user has modified this record error.

```
OnRun()
SalesHeader.MODIFY;
SalesLine.MODIFY; // If Shadow table listener updates the line, this call will fail
```

Make sure to enforce Read Permissions

```
OnOpenPage()
SetPermissionFilters;
```

Shadow Table

Pros

Quick load time, read is cheap

Single API

Filtering is possible since fields are persisted

API fields not increasing SQL row size

Can add API specific logic (proxy)
Write permissions are kept

Read permissions must be coded

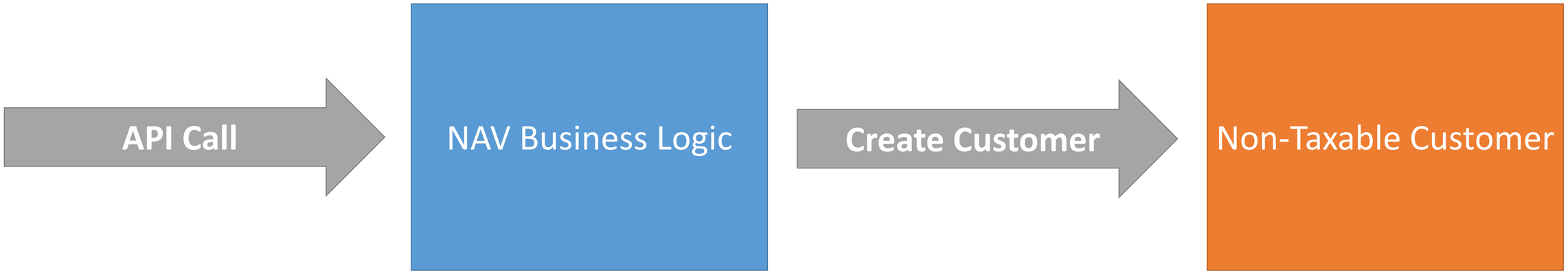
Cons

Write operations cost more

Data is duplicated

Read permissions are hard if there are field level permissions

“Patterns” – Register field set




```
{
  "displayName": "Nikola Gray Account Tax",
  "taxable": false
}
```

“Patterns” – Register field set

Problem

```
{
  "displayName": "Nikola Gray Account Tax",
  "taxable": false
}
```

```
{
  "displayName": "Nikola Paying Tax",
}
```

After validation, before insert both records will look the same

Customer:
Name: “Nikola Gray Account Tax”
Taxable: False

Customer:
Name: “Nikola Paying Tax”
Taxable: False

Insert trigger defaults taxable to true on both

Wrong

Correct

“Patterns” – Register field set

NAV TECH DAYS 2017
mibusa.com

Problem

No generic solution. Depending on the scenario we need to:

- Throw an error – serious mistake would be made
- Fix the value silently – e.g. Updating the casing on the name, validation order etc.
- Enforce the set value

We need to code the solution on use case basis

Order of the fields on the API page is important – It determines the validation order

“Patterns” – Register field set

Solution

Decorate the validation triggers to populate temp buffer so we know field was part of the payload

Override the logic on the Insert/Modify triggers

```
pricesIncludeTax - OnValidate()
RegisterFieldSet(FIELDNO("Prices Including VAT"));
```

```
LOCAL RegisterFieldSet(FieldNo : Integer)
LastOrderNo := 1;
IF TempFieldBuffer.FINDLAST THEN
  LastOrderNo := TempFieldBuffer.Order + 1;

CLEAR(TempFieldBuffer);
TempFieldBuffer.Order := LastOrderNo;
TempFieldBuffer."Table ID" := DATABASE::"Sales Cr. Memo Entity Buffer";
TempFieldBuffer."Field ID" := FieldNo;
TempFieldBuffer.INSERT;
```

“Patterns” – Register field set

Pros

Can control the flow and the validation logic

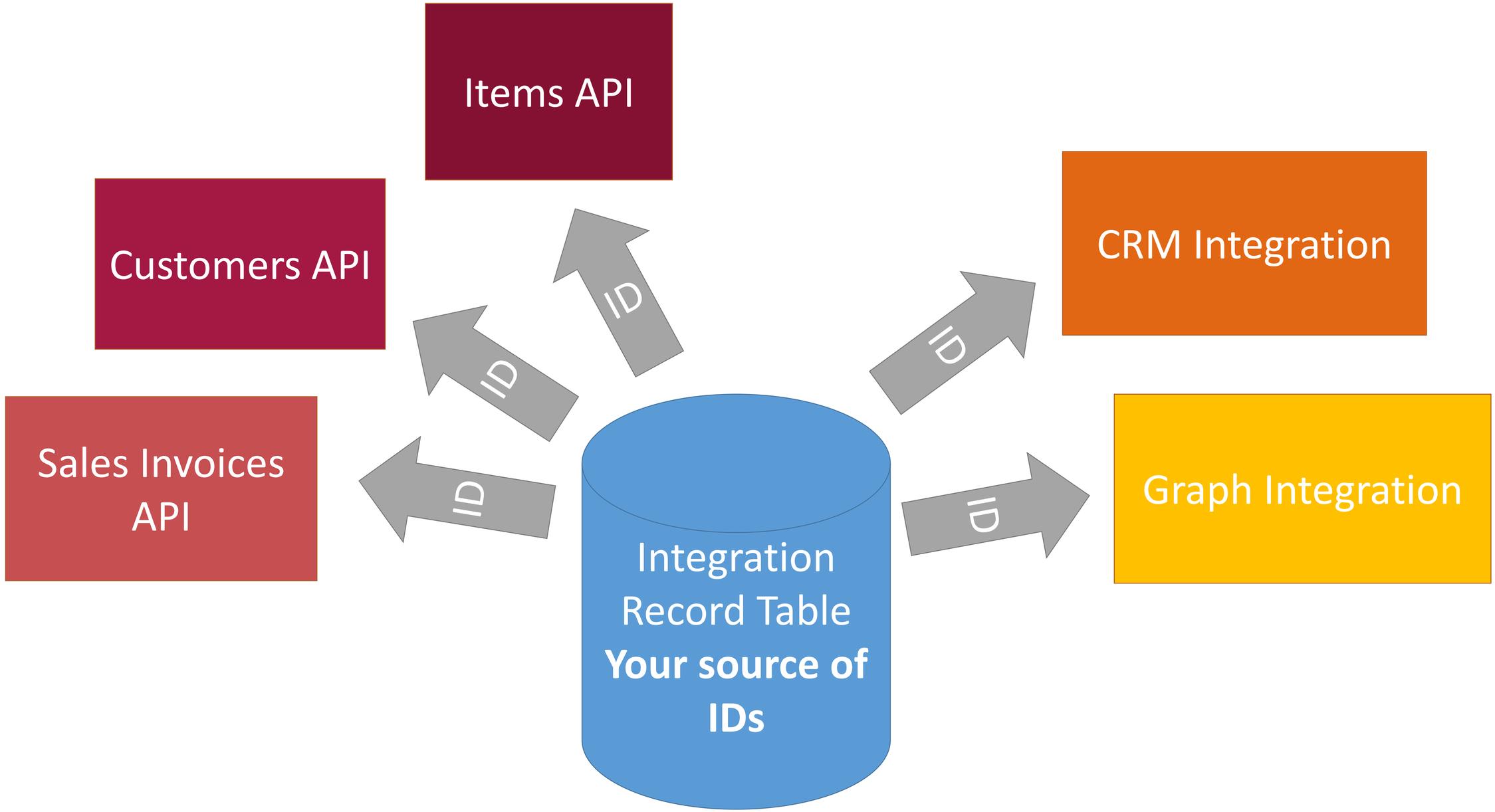
Cons

Code is harder to understand

It may be hard to control the logic

If you forget to decorate the validate trigger there is a bug

“Patterns” – Integration Id



“Patterns” – Integration Id

Problem:

Ensure an immutable public unique id for an entity

Must be the same for all integrations (API, CRM sync, Graph Sync…)

Solution:

Generate the Id and creation time for the entity and store the record id and unique id in Integration Record Table.

Whenever the record gets modified restore the in case it gets modified.

“Patterns” – Integration Id

Codeunit 1 is a starting point, Integration Management main codeunit

```
OnDatabaseInsert(RecRef : RecordRef)
OnBeforeOnDatabaseInsert(RecRef);
ChangeLogMgt.LogInsertion(RecRef);
IntegrationManagement.OnDatabaseInsert(RecRef);
OnAfterOnDatabaseInsert(RecRef);

OnDatabaseModify(RecRef : RecordRef)
OnBeforeOnDatabaseModify(RecRef);
ChangeLogMgt.LogModification(RecRef);
IntegrationManagement.OnDatabaseModify(RecRef);
OnAfterOnDatabaseModify(RecRef);

OnDatabaseDelete(RecRef : RecordRef)
OnBeforeOnDatabaseDelete(RecRef);
ChangeLogMgt.LogDeletion(RecRef);
IntegrationManagement.OnDatabaseDelete(RecRef);
OnAfterOnDatabaseDelete(RecRef);

OnDatabaseRename(RecRef : RecordRef;xRecRef : RecordRef)
OnBeforeOnDatabaseRename(RecRef,xRecRef);
ChangeLogMgt.LogRename(RecRef,xRecRef);
IntegrationManagement.OnDatabaseRename(RecRef,xRecRef);
OnAfterOnDatabaseRename(RecRef,xRecRef);
```

“Patterns” – Integration Id

Child enties do not have an Integration ID, they update parents integration ID

When record is deleted RecordID is blanked on the table

Never delete Integration records

“Patterns” – Integration Id

NAV TECH DAYS 2017
mibusa.com

Pros

Single place for IDs for all integrations
Tracks Deleted Records

Works if record is renamed

Allows specifying a record from outside

Cons

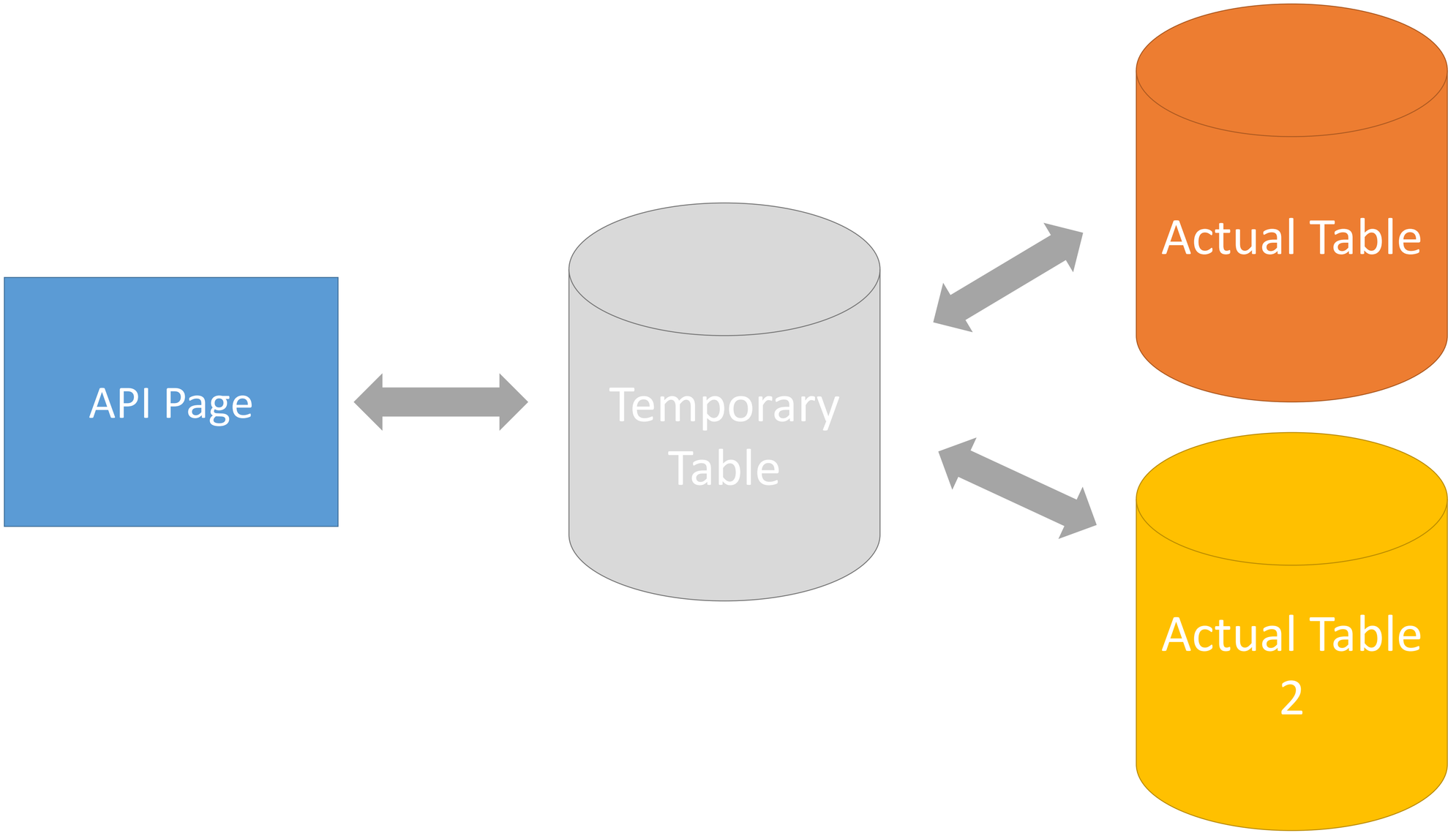
Fragile – can be deleted, Record references can be broken

Long running processes can put a lock, timing out other users

Some entities should not be tracked – e.g. sales invoice lines, since they would result in large number of records

Should be implemented in platform for all records

“Patterns” – Temporary Tables



“Patterns” – Temporary Tables

NAV TECH DAYS 2017
mibusa.com

Problem:

Cannot put some tables into Integration Record Table since they would generate to many entries

The response can only contain few records

It is too costly to maintain aggregate table for few records

Solution:

Use temporary table

“Patterns” – Temporary Tables

NAV TECH DAYS 2017
mibuso.com

```
OnFindRecord(Which : Text) : Boolean
IF NOT LinesLoaded THEN BEGIN
  FilterView := GETVIEW;
  DocumentIdFilter := GETFILTER("Document Id");
  SalesInvoiceAggregator.LoadLines(Rec,DocumentIdFilter);
  SETVIEW(FilterView);
  IF NOT FINDFIRST THEN
    EXIT(FALSE);
  LinesLoaded := TRUE;
END;

EXIT(TRUE);
```

Web Service Page (Odata, API) behaves differently than actual page – it executes different platform code.

Find record was used since page was closed after OnOpen trigger

“Patterns” – Temporary Tables

NAV TECH DAYS 2017
mibusa.com

Pros

No new permanent tables are introduced

No integration records are introduced

Can have a different flow via Proxy Table

In some cases if we get support for platform Integration Ids, we will be able to refactor to actual records without changing the signature

Cons

Have to code transfer of data from temp to physical

Must be sure data set is small

NAV TECH DAYS 2017
mibuso.com

Plans for “Tenerife”

Plans for Tenerife

media-write
delta support
WEB hooks
Advanced filtering
Extensibility
Api/custom
..
Planned API's

Main take-aways

ODataV4 introduced new capabilities, we are adding more.

Easy to learn. You can develop in any language.

New /api endpoint is available and supported by Microsoft. Alternative to extensions.

You can develop ODataV4 web services in the extensions.

How to proceed from here

NAV TECH DAYS 2017
mibusa.com

Get a tenant and play with the APIs

Get started – http://aka.ms/getstartedwithapis

For quick connection, e.g., from Postman, directly access tenant URL and use basic authentication (web service access key)

Give us feedback

Missing entities?

Desired fields?

Requested methods (bound actions)?

Start building connect apps!

Be part of the first wave

NAV TECH DAYS 2017
mibusa.com

QUESTIONS?