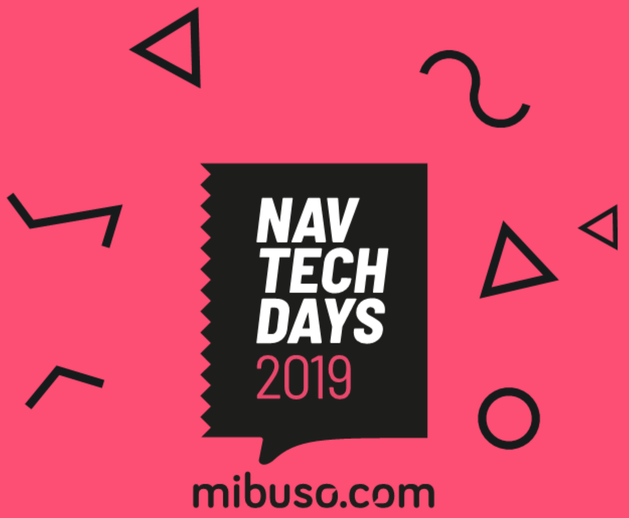


Unlocking new integration potential for Dynamics 365 BC with Azure Event Grid and Azure Integration

Dmitry Katson, Tharanga Chandrasekara

When you are passionate about
Microsoft Dynamics NAV/365 Business Central

About this guy on the stage

- **Dmitry Katson, MVP**
  - 15Y of NAV experience
  - 2Y of BC experience ☺
  - 3Y of AI experience
- Lives in St. Petersburg, Russia
- Father of 2

NAV TECH DAYS 2019
mibuso.com

About this guy on the stage

- **Tharanga Chandrasekara, MVP**
  - 7Y of NAV experience
  - 2Y of BC experience

- Lives in Auckland, New Zealand
- Originally from Sri Lanka

Agenda

NAV TECH DAYS 2019
mibuso.com

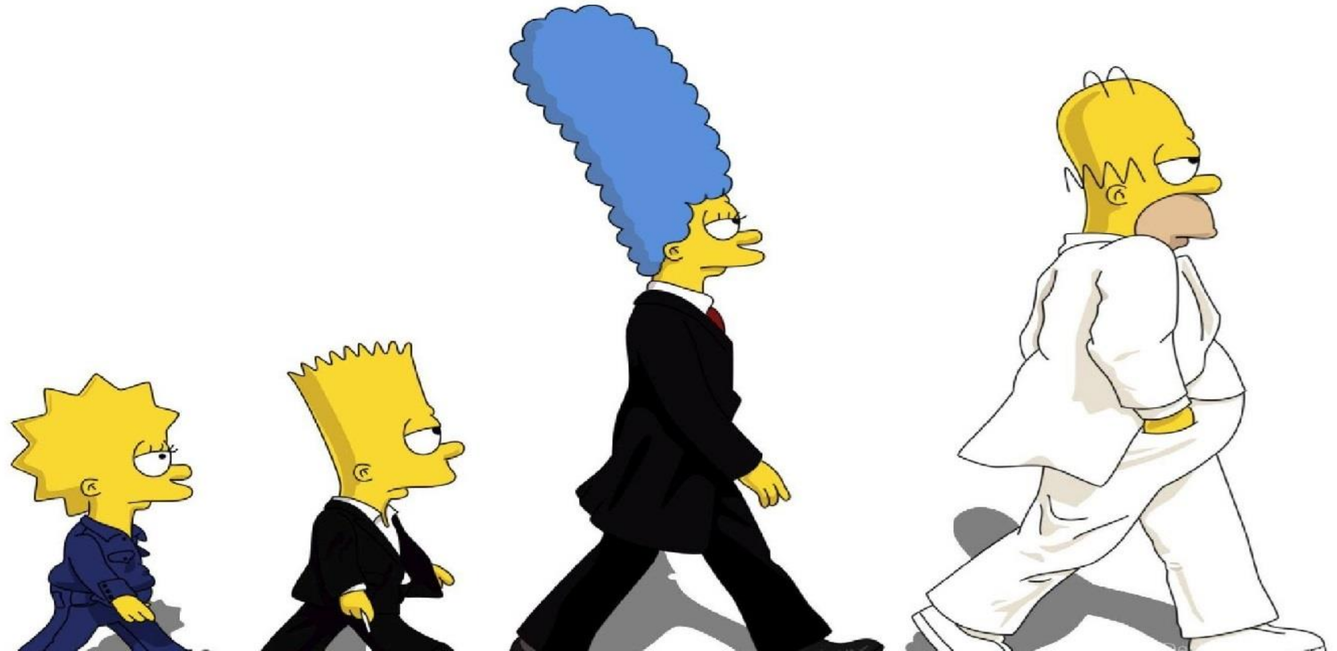
- Introduction to reactive programming
- Admin API
- Azure Event Grid
- Hands On
- Design Considerations
- Takeaways

Keeping systems in sync

- A common integration problem
- Multiple patterns
  - Polling
  - Push

Polling challenges

- Eventual consistency
- Polling time sweet spot
- Can drain resources



Push challenges

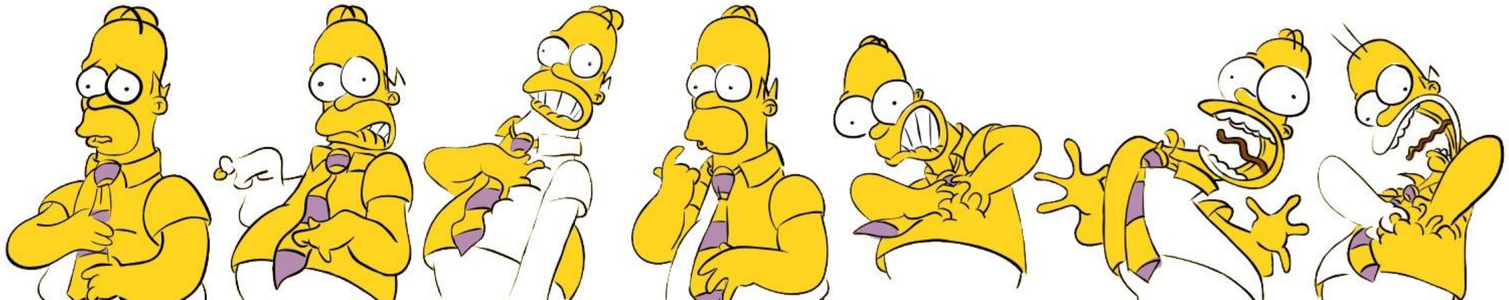
- Onus on source to keep systems up-to-date
- Integration must understand all downstream systems
- Can become very complex

Business Central scenario

NAV TECH DAYS 2019
mibuso.com

Holding company maintain all the COA, Customer, Vendor related master data.

How to synch data between holding company and other sub companies?



Environment management Challenges

NAV TECH DAYS 2019
mibuso.com

You need to manually ….

- Create new environment for each localization
- Create companies for each legal entity
- Publish and install extensions for each environment
- Configure upgrade windows
- Manage backups
- Check health of the environments

Administration API

- Authenticate (Get Token)
- Get / Create environments
- Get / Create companies
- Set upgrade windows
- Check telemetry
- Manage notifications
- Manage support cases
- Manage backups

https://api.businesscentral.dynamics.com/admin/v2.0/{api_request_command}

Data Sync Challenges

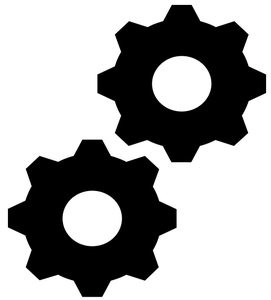
- Sub companies may use different language as their business language.
- Multiple companies might be interested in the master data changes on holding company, therefore have to push or pull the data into to multiple places.
- Complexity of building an integration when a new sub company is established.
- Complexity of adding a new entity to integration layer.

How to handle it through BC

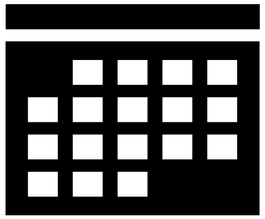
NAV TECH DAYS 2019
mibuso.com

- Push all the events into a internal table and build an API around that.
- Use the in-built subscriber functionality in BC to call the interested endpoints.

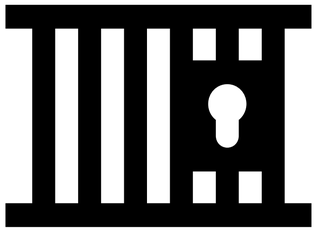
First approach



Event Happen in Business Central

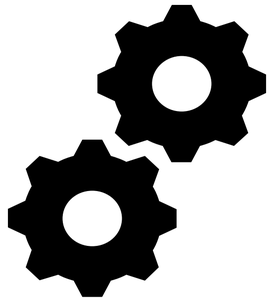


Push Event Details to Internal Table

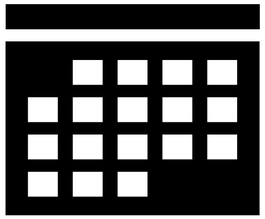


Allow to Access through API

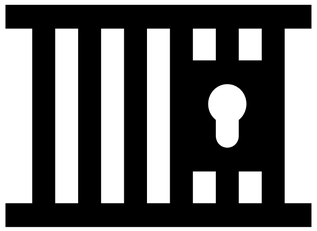
Second approach



Event Happen in Business Central

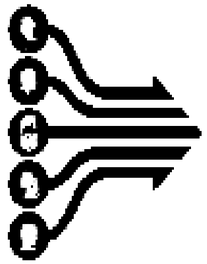


Business Central Call Subscribers



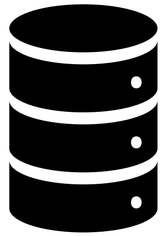
Allow to Access through API

Challenges faced



Too many API requests

Increase of Security Threats



Rapid growth of the database size

Deadlocks

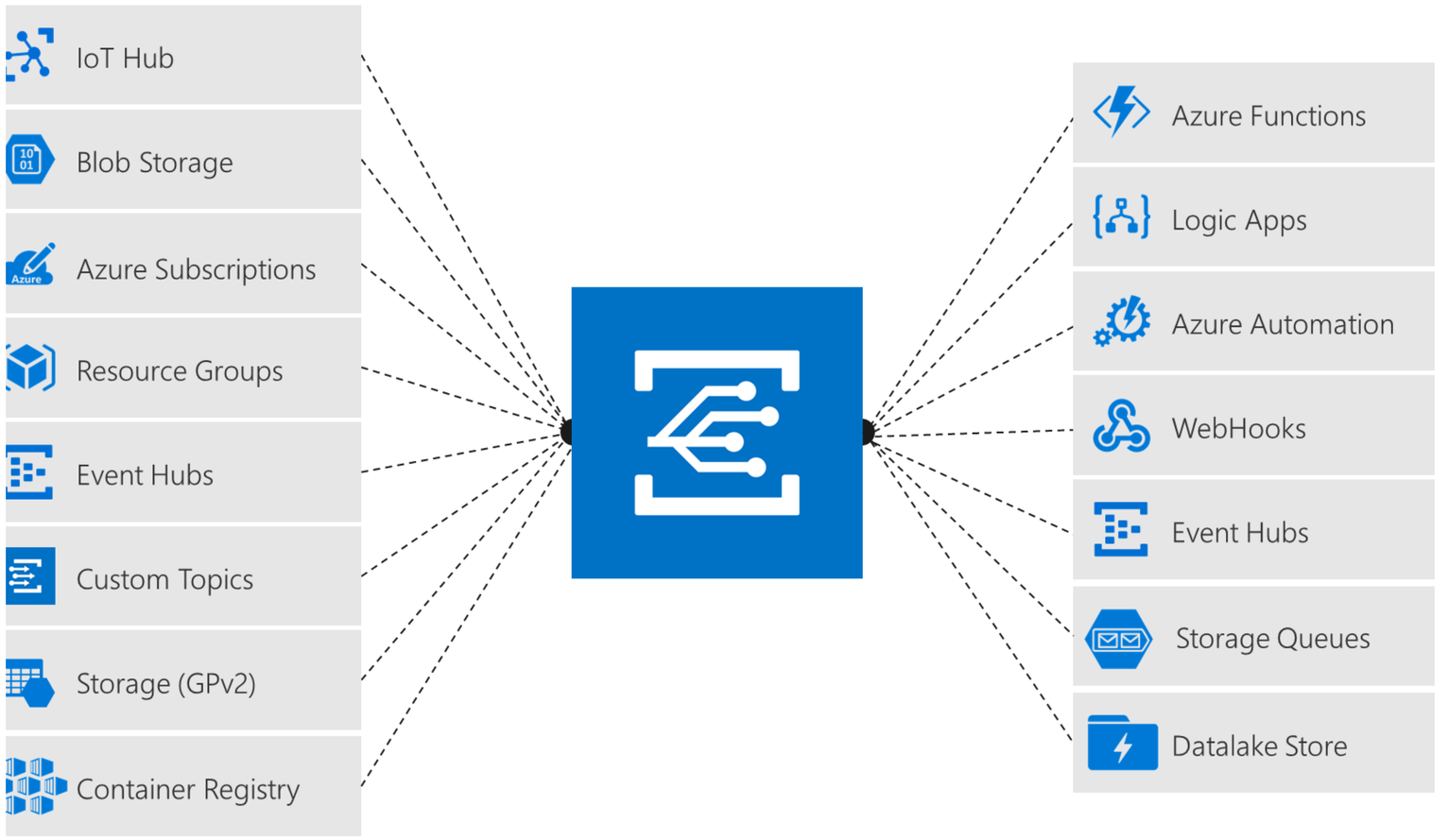
What is a better solution?

NAV TECH DAYS 2019
mibuso.com

- A brokered event system
- Source publish to broker
- Interested parties subscribe to broker
- Decouples publishing to subscribing processes
- Introduces a new player – the pub/sub broker

What is a better solution?

- Simplify event-based app development with a publish-subscribe model
- Simple HTTP-based event delivery
- Build better, more reliable applications through reactive programming
- Focus on product innovation



Introducing Azure Event Grid

What is an event?

NAV TECH DAYS 2019
mibuso.com

Happen even if no one is listening


```
[{
    "topic": "BC",
   "subject": "bc/masterdata/glAccount/translate",
   "eventType": "bc/masterdata/glAccount/update",
    "eventTime": "2019-11-11T23:20:49",
   "id": "61c1308e-c10a-4fa6-a0b3-8eb098da2114",
   "data": {
      "source": "BC",
      "destination": "CRM",
      "details": [
            {
                  "number": "10000",
                  "accountName": "Jordan Moresby"
            }
      ]
       }
}]
```


Sample scenarios for Event Grid

NAV TECH DAYS 2019
mibuso.com

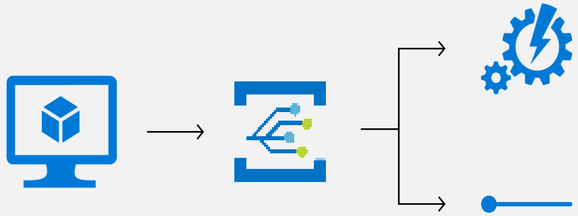
Serverless apps

Trigger a function to run Cognitive API when a file is added to storage

Ops automation

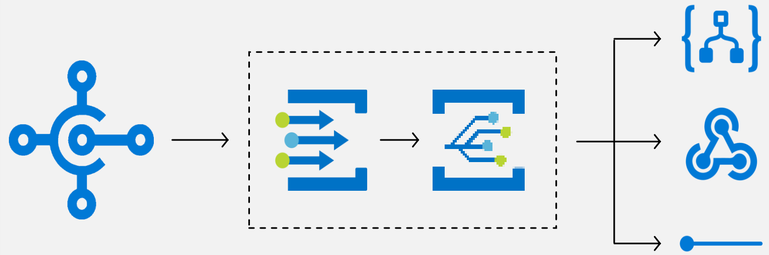
Use a function to run a compliance check on each newly created SQL database

Tag newly provisioned VMs with Azure Automation and add to metadata store



Third-party integration

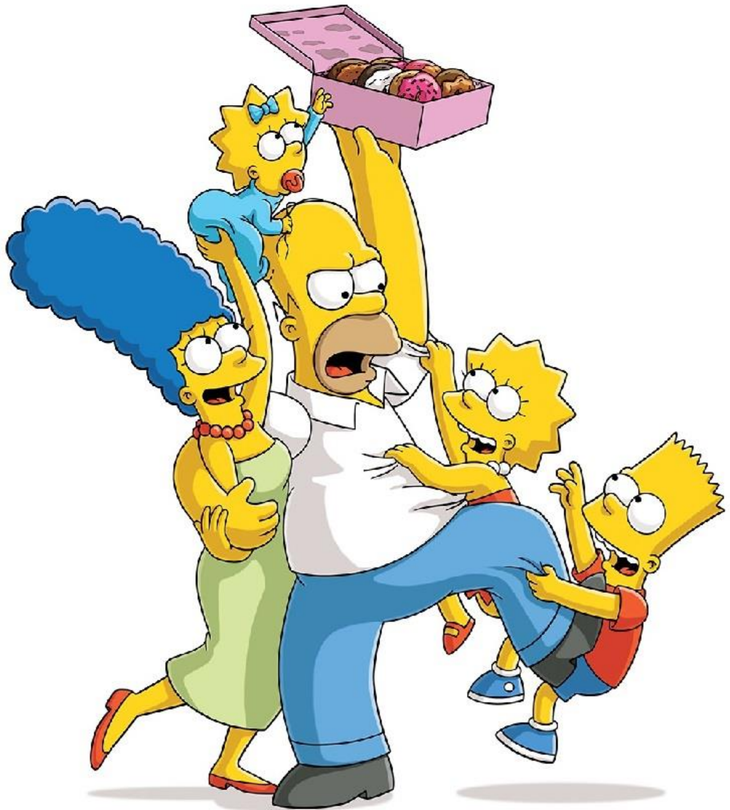
Use custom “Service Order Released” event to drive integration with external systems



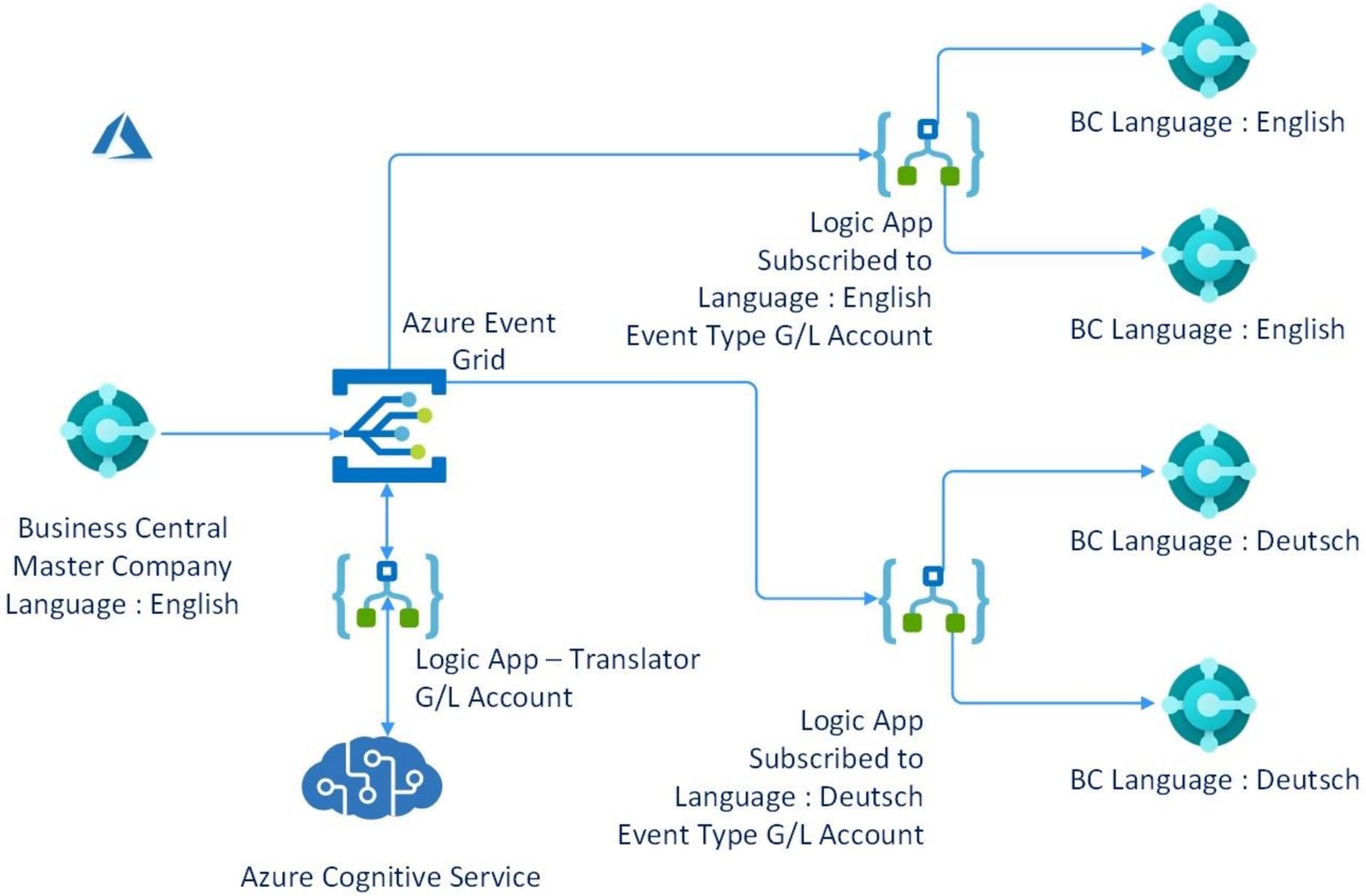
Hands on

NAV TECH DAYS 2019
mibuso.com

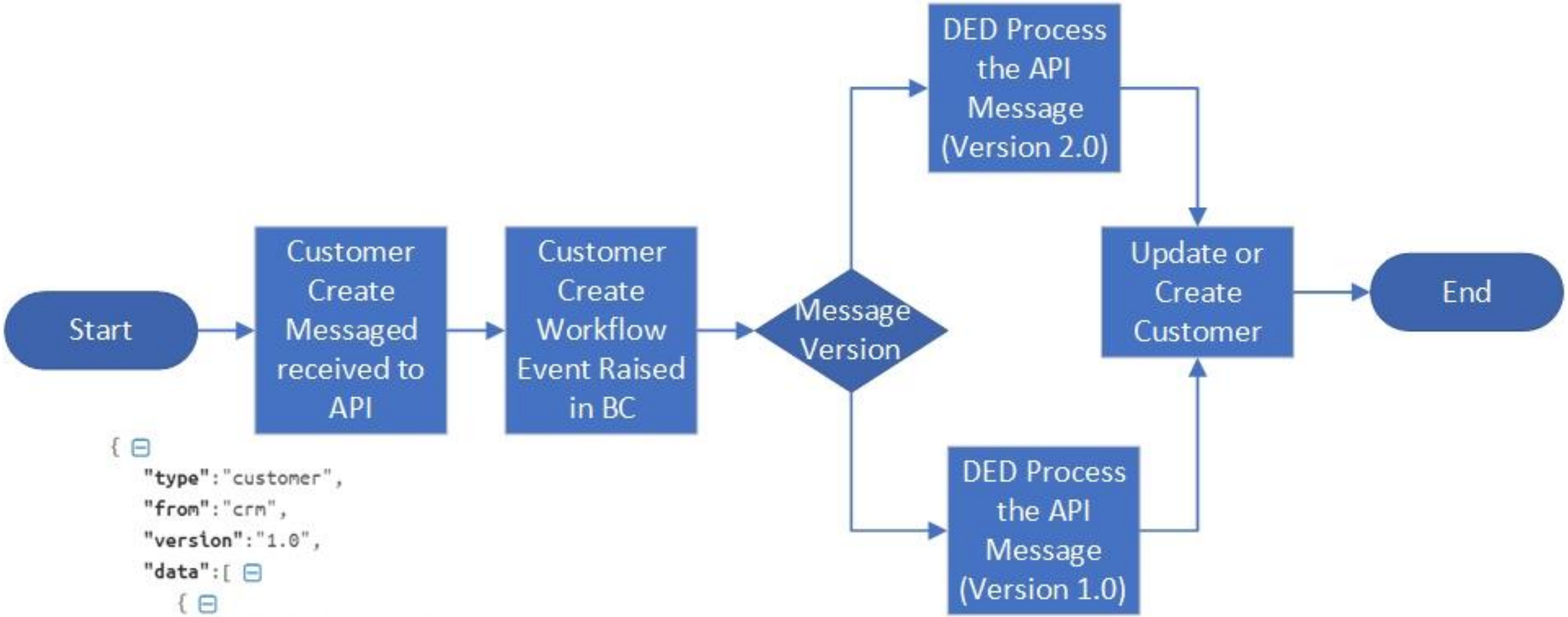
- How to push a COA to multiple companies which uses different languages.
- How to add an new entity (Vendor) to the integration



Hands on :Sync COA to multiple companies



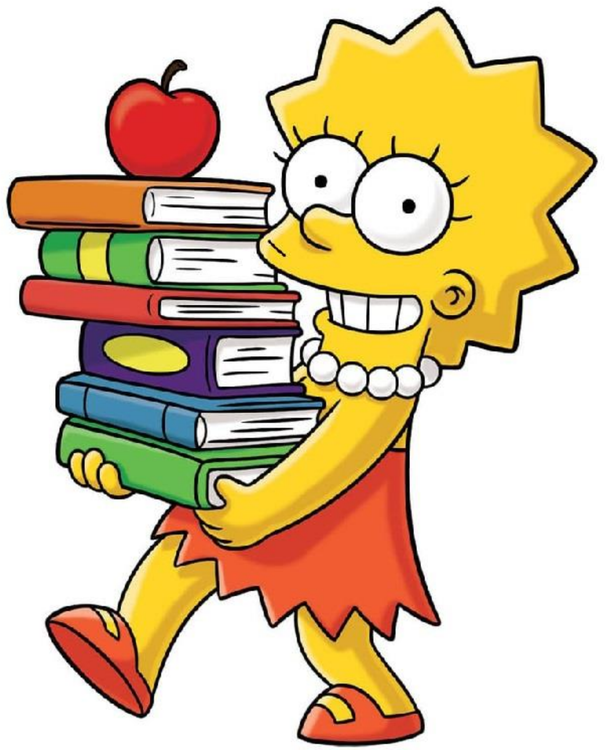
How to handle custom fields




```
{
  "type":"customer",
  "from":"crm",
  "version":"1.0",
  "data":[
    {
      "number":"CUST-0213",
      "name":"bart simpson",
      "age":"10",
      "occupation ":"Student at Springfield Elementary School",
      "address":"742 Evergreen Terrace, Springfield"
    },
```

Design considerations

- Follow the event core concepts
- Define the event triggers
  - Generic (CRUD)
  - Status changes
- How much data is enough?
  - Think about the Event Grid limits
  - Claim check pattern



Takeaways

- Event grids decouples event publishing from handlers
- Enables real time scenarios
- Easy extensibility in BC to support it
- Think about the types of event to publish
- Think about the data strategy

NAV TECH DAYS 2019
mibuso.com

Q&A

Any Questions?

NAV TECH DAYS 2019
mibuso.com

Thank You!