



Integration in einer serverlosen Welt

Mohammed Brückner @ platformeconomies.com

Mohammed Brückner // "All views are my own"



Präfix: Schwierige Beziehung

Die anstrengende Geschichte der IT-
Wertschöpfung

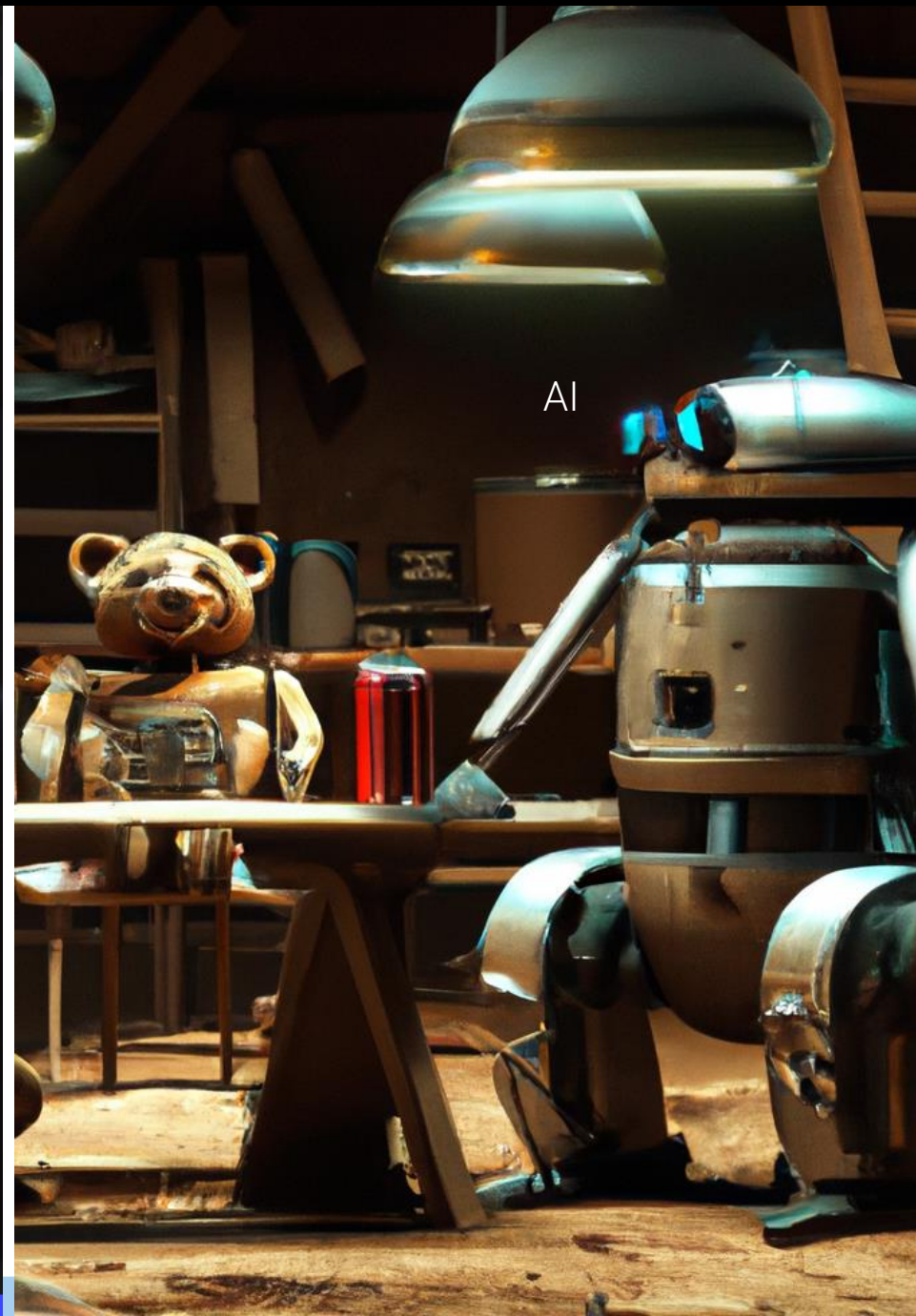


“Business” Vertreter

IT “Delivery Partner”

“Projektmanager”

AI





Frustrierte Entwickler



Frustrierter CIO



GOODBYE, WATERFALL!

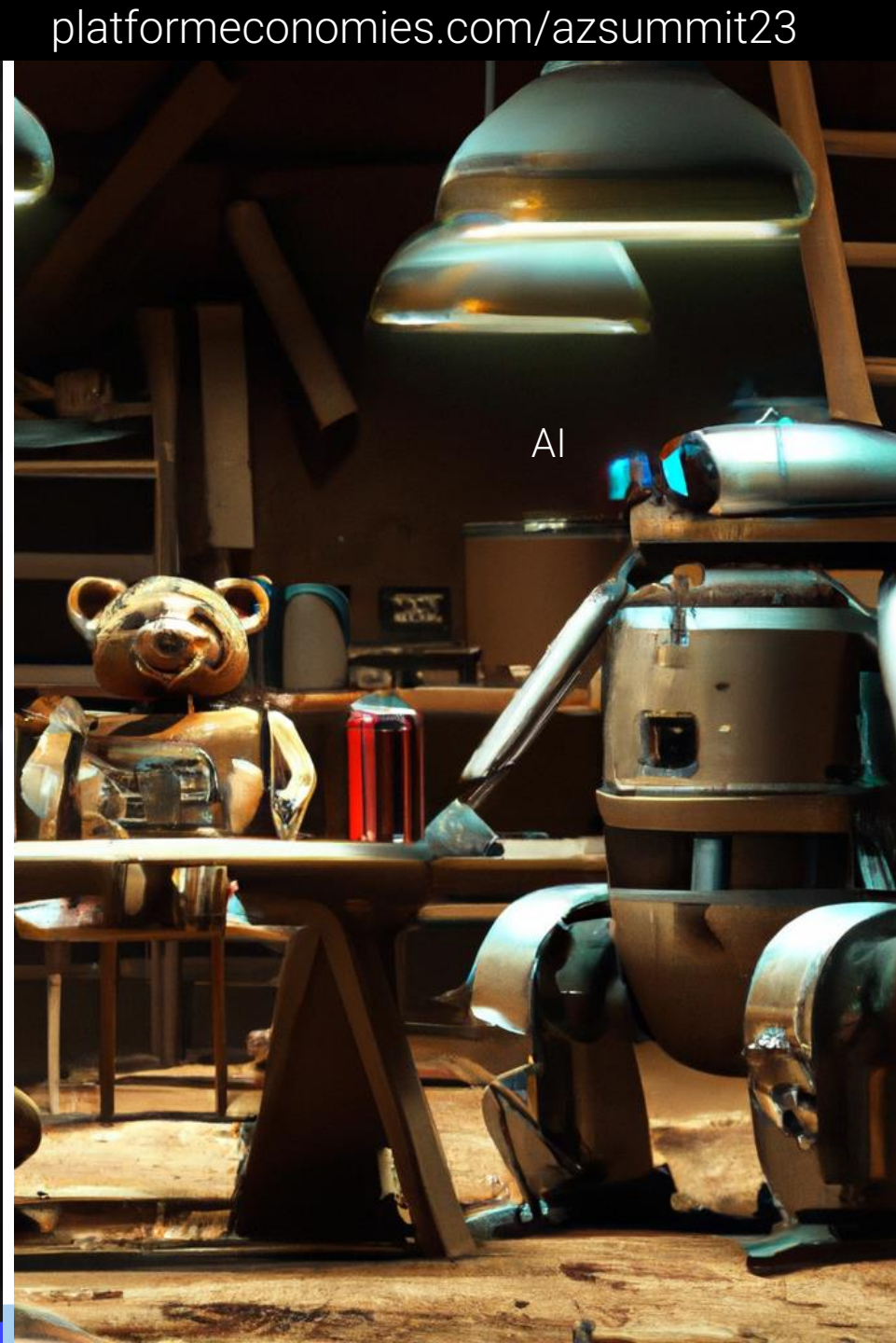
WELCOME, AGILE!

"Business" Vertreter

"IT Product Owner"

"Scrum Master"

AI



“Business” Vertreter

“IT Product Owner”

“Scrum Master”

Näher am Business, Chance
zu “partizipieren” (“seat at the table”)

Was ist mit der Teilnahme an der
Erzeugung von IT in die andere Richtung?

“Business” Vertreter

“IT Product Owner”

“Scrum Master”

Näher am Business, Chance
zu “partizipieren” (“seat at the table”)

WELCOME, LOW-CODE!



BEST RUN COMPANIES RUN ON EXCEL



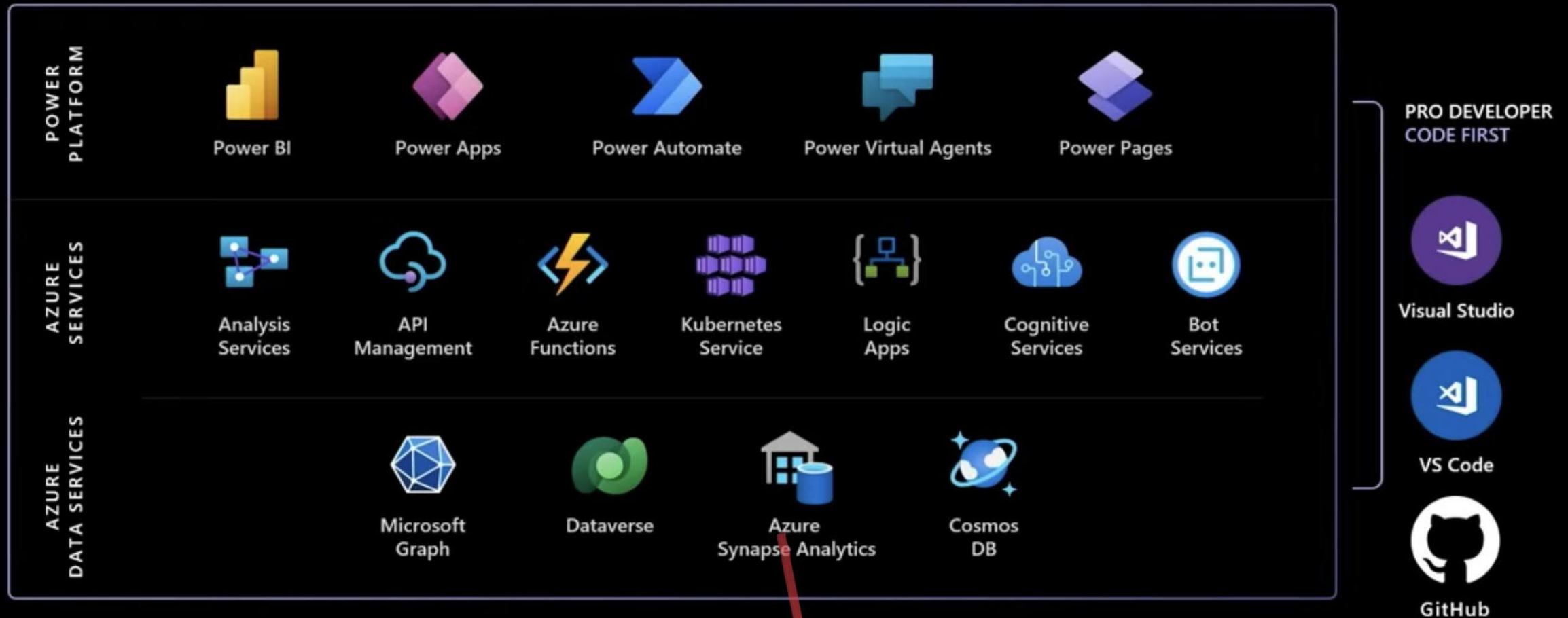
What are benefits of combining low-code and pro-code?

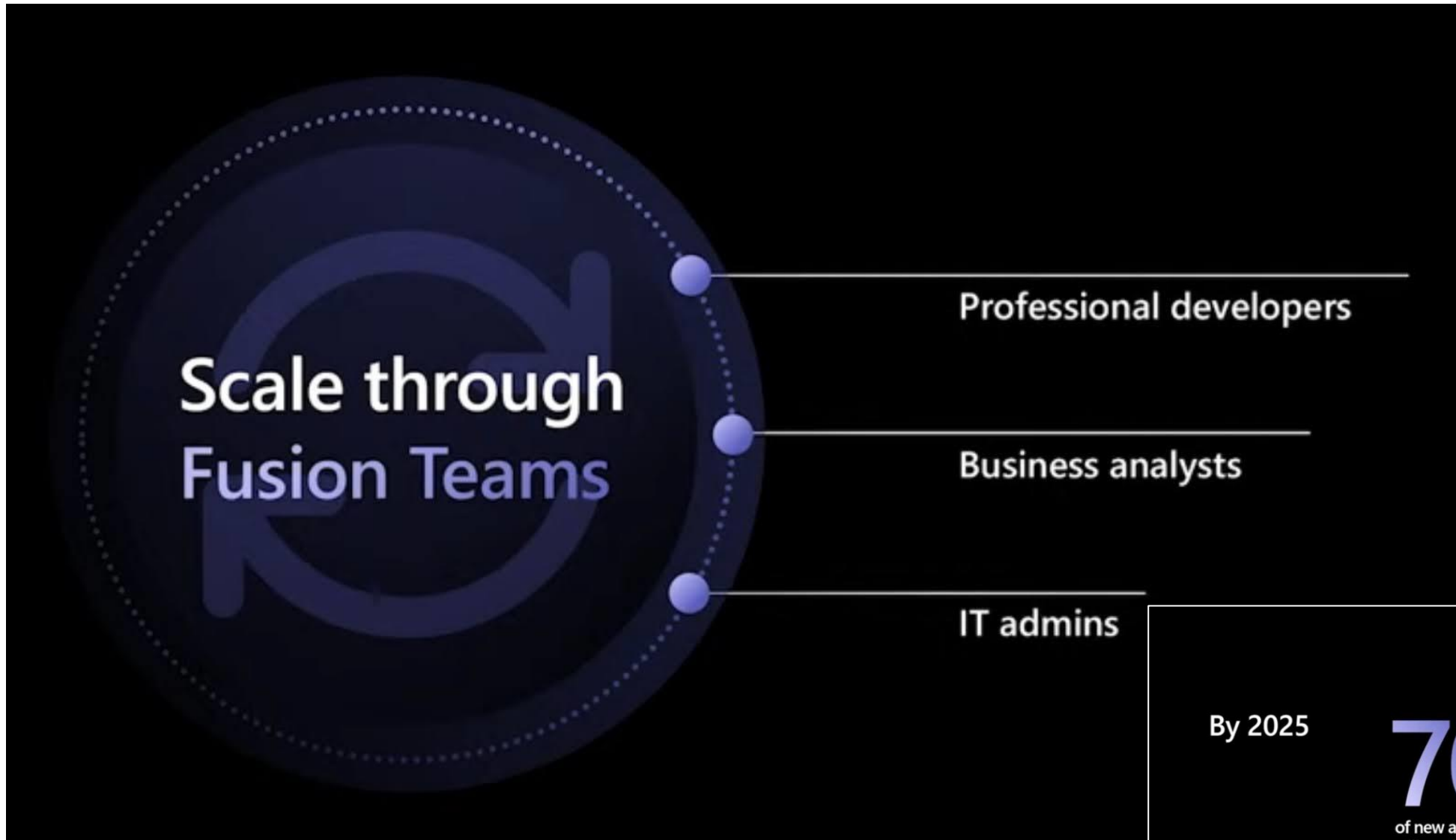
Some benefits of combining low-code and pro-code include:

- **Faster development:** Low-code platforms can be used to quickly prototype and build the basic structure of an application, while pro-code can be used to add custom functionality and fine-tune the application.
- **Improved collaboration:** Low-code platforms can be used by non-technical team members, such as business analysts, to contribute to the development process. This can improve collaboration and communication within the team.[\[0\]\[2\]](#)
- **Increased flexibility:** Combining low-code and pro-code allows for a balance between rapid development and customizability. This can be useful in situations where the requirements of the project are not fully known or are subject to change.[\[0\]](#)
- **Better performance:** Pro-code can be used to optimize the performance of an application, which can be especially important for applications with high traffic or complex data processing requirements.

Overall, combining low-code and pro-code can provide a more efficient and flexible approach to application development.

Pro Code + Low-Code = No Limits





By 2025

70%

of new applications developed by enterprises
will use low-code or no-code technologies

COMPARE: [#LOWCODEFEBRUARY](#)

Microsoft ist „Leader“ im „Magic Quadrant for Enterprise Integration as a Service 2023“

5 Reasons Enterprises choose Azure Integration Services

Continuous innovation

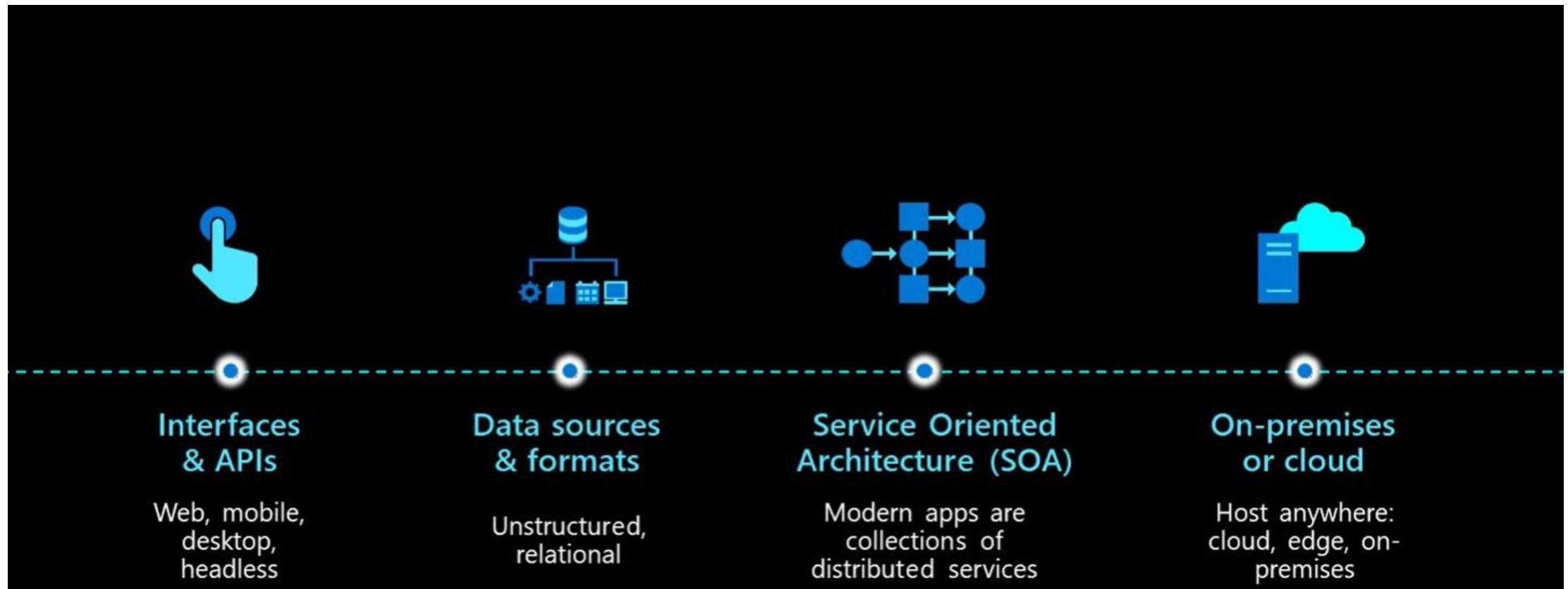
Globally managed offerings

Strong security and compliance

Flexible pricing

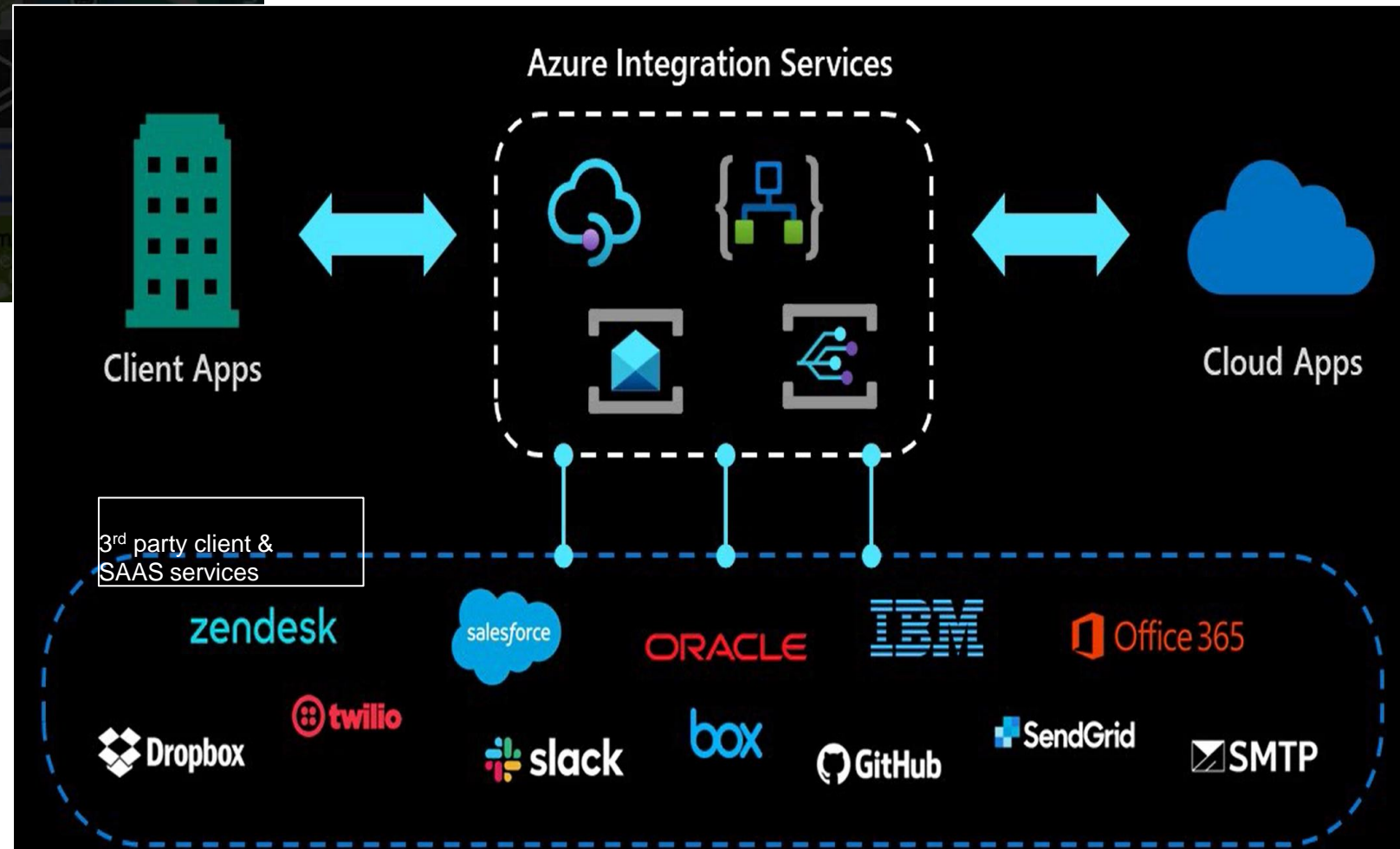
Supports both citizen and professional developers

COMPARE: [#LOWCODEFEBRUARY](#)



COMPARE: [#LOWCODEFEBRUARY](#)

900+
Data connectors



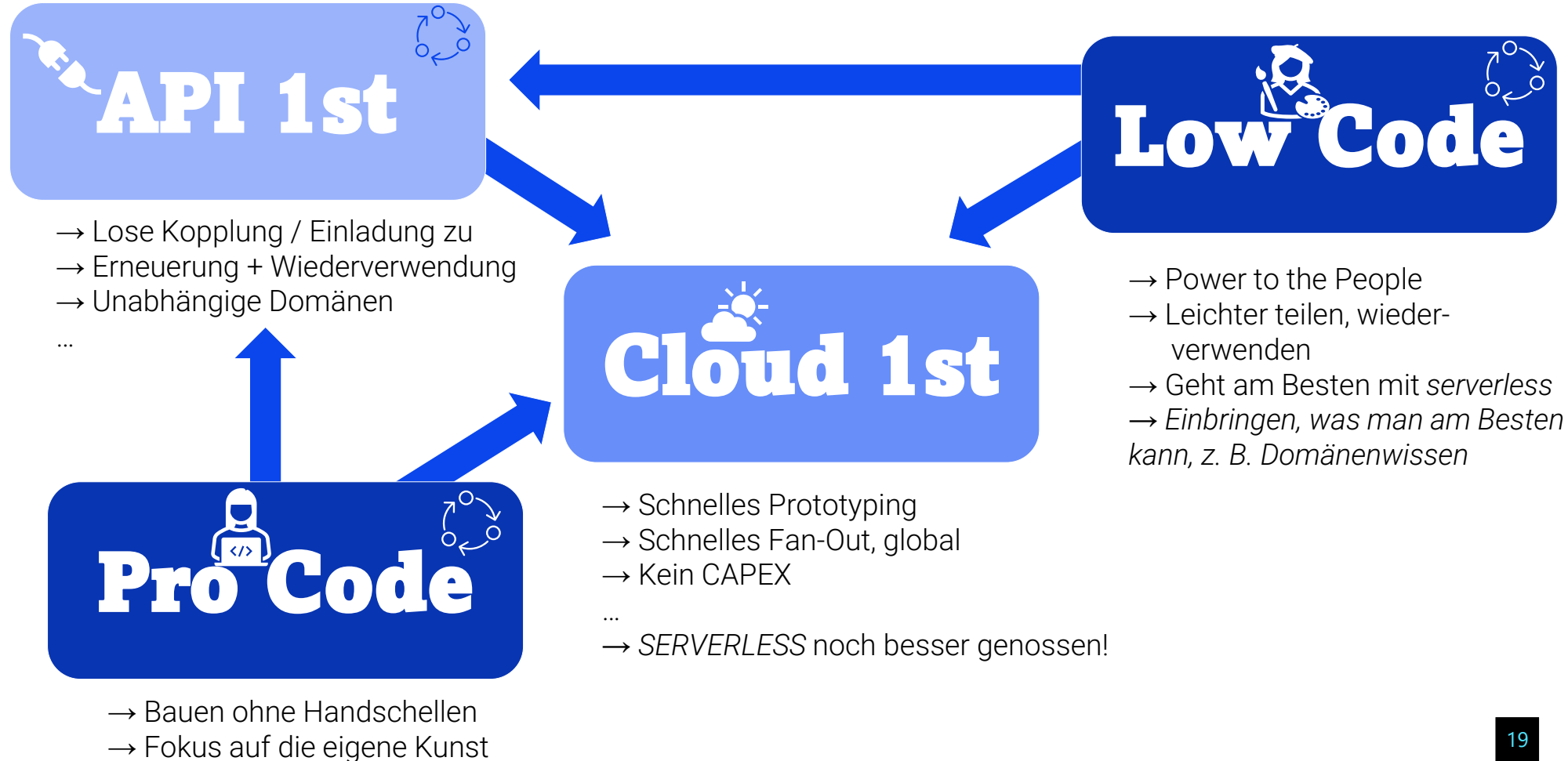
COMPARE: [#LOWCODEFEBRUARY](#)

Die "Secret Sauce" für den Erfolg von Fusion Teams

Überzeugung:

Das Operating Model muss weg von zentralisierten "Silo" Teams hin zu **autonomen, interdisziplinären zweckgebundenen Teams**.

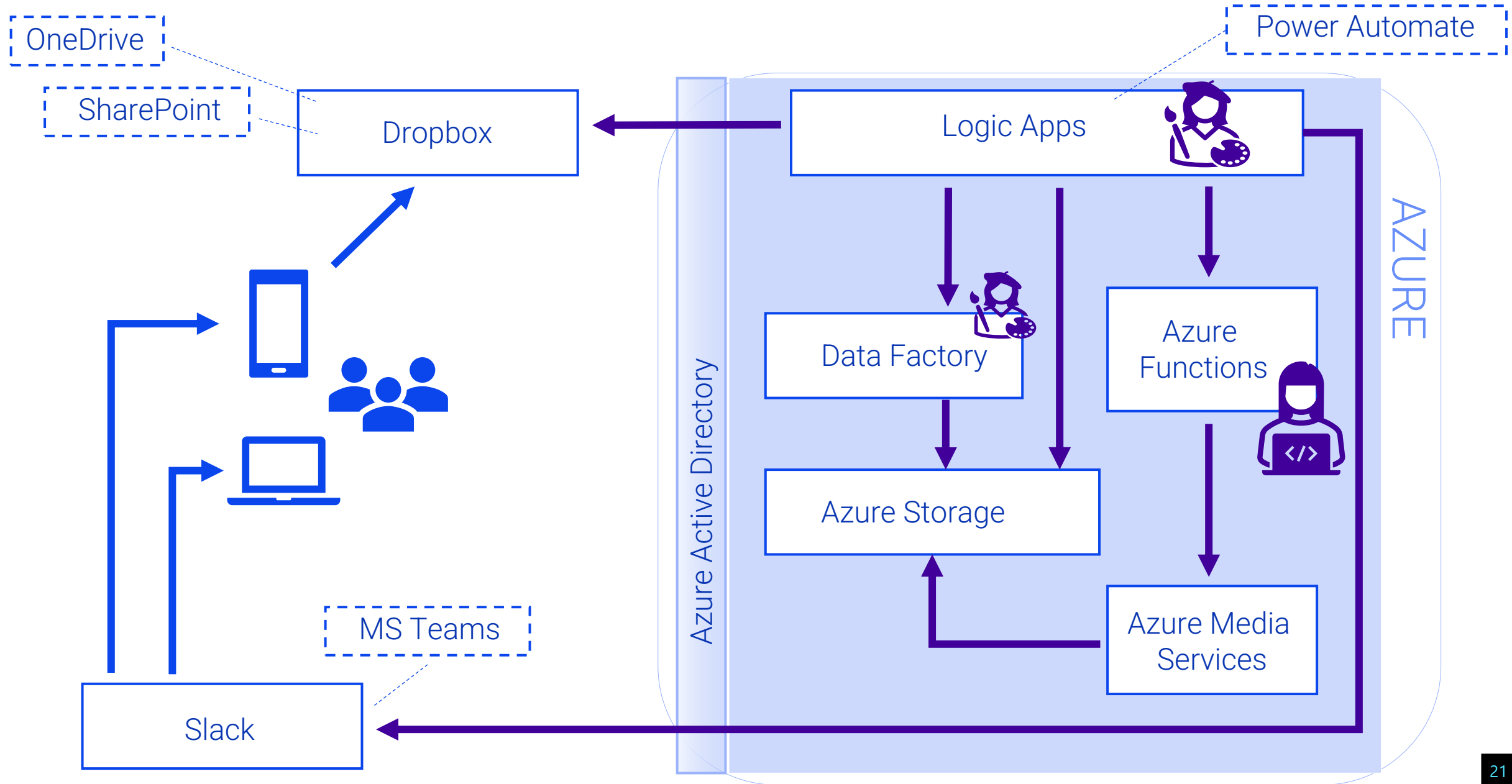
Mit Fokus **auf Service Domänen**, die fortlaufende **Produktlebenszyklen** unterstützen. Statt Projektlebenszyklen.



Endlich zum spaßigen Teil

EXAMPLE SCENARIO
WALK-THROUGH



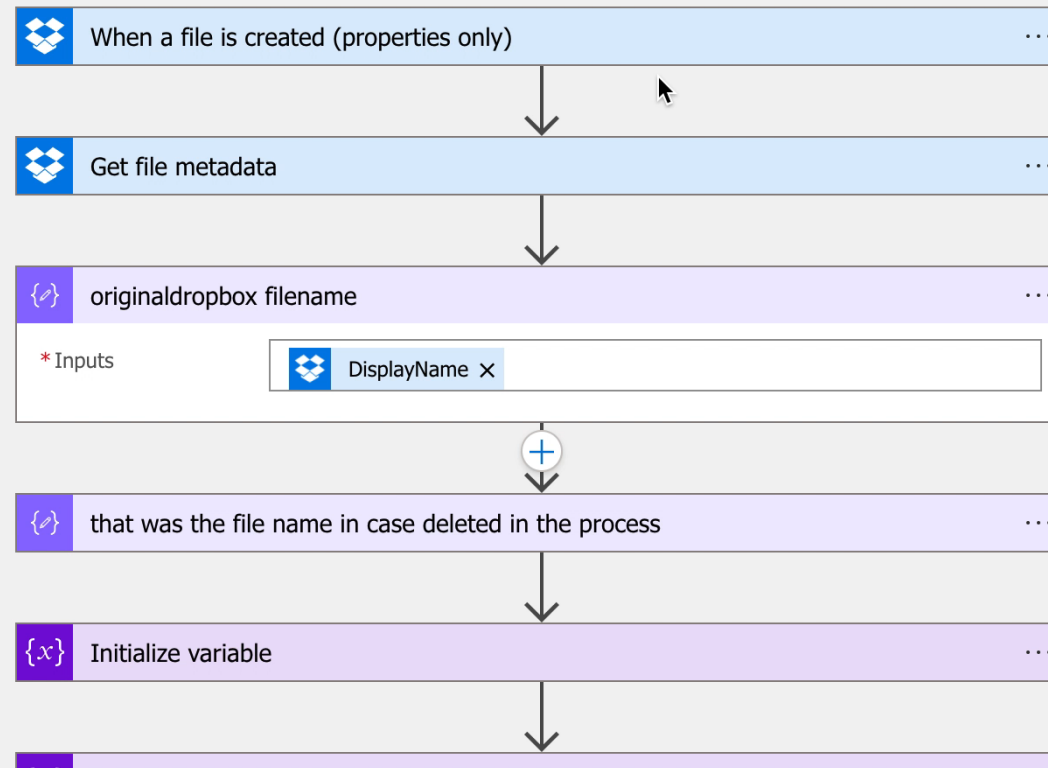


Logic Apps Designer ...



Save Discard Run Trigger Designer Code view Parameters Templates Connectors Help Info

100%



obtain a valid access token ...

↓

Parse JSON ...

+

Create a pipeline run ...

DataFactory will use application/octet-stream as content-type when copying over files to BlobStorage, which means images etc will not shown in browser but downloaded -as of 2022 it seems there is no way to set the content-type ✕

* Subscription ✓

* Resource Group ✓

* Data Factory Name ✓

* Data Factory Pipeline Name ✓

parameters ✕

```
{
  "a_token": "{access_token}",
  "container": "pubtmp",
  "file_name": "{DisplayName}",
  "modified_file_name": "{Outputs}",
  "subfolder": "large4publish"
}
```

Add new parameter ✓





do plenty of stuff to convert the video file



Set variable 2



kickofftransformationams



Request Body

```
{
  "filesasuri": "Web Url ×",
  "origfilename": "Outputs ×"
}
```

Add new parameter



Parse JSON 2



Data Factory

Validate all

Publish all

Preview experience

Off

Factory Resources

Filter resources by name

Pipelines

4

copyfilefromblob2fileshare

copyfilefromBlobContainer2BlobCon...

copyfilefromPub2Pub

pipeline1

Change data capture (preview)

0

Datasets

5

Data flows

0

Power Query

0

Activities

Search activities

Move & transform

Synapse

Azure Data Explorer

Azure Function

Batch Service

Databricks

Data Lake Analytics

General

HDInsight

Iteration & conditionals

Machine Learning

Power Query

copyfilefromBlobC...

pipeline1

copyfilefromPub2P...

copyfilefromblob2fi...

Validate

Validate copy runtime

Debug

Add trigger

Copy data

get data from dropbox

General

Source

Sink

Mapping

Settings

User properties

Source dataset *

dropboxdata

Open

New

Learn more

Request method *

POST

Additional headers

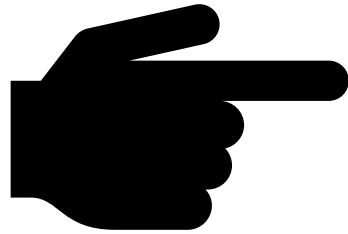
Authorization: Bearer @{pipeline().par...

The screenshot displays the Visual Studio Code editor with a TypeScript file named `index.ts` open. The file is part of a project named `kickofftransformationams`. The code in `index.ts` includes imports for `AzureMediaServices`, `KnownAacAudioProfile`, `KnownH264Complexity`, `StandardEncoderPreset`, `AssetContainerPermission`, `AzureFunction`, `Context`, `HttpRequest`, `ManagedIdentityCredential`, `uuidv4`, `jobHelper`, `factory`, and `url`. The code also includes comments and variable declarations for `subscriptionId`, `resourceGroup`, `accountName`, `inputFile`, `inputUrl`, and `inputUrl`.

The Explorer sidebar on the left shows the file structure of the project, including `.vscode`, `azure-functions-core-tools`, `blobshuffler`, `deleteamsassets`, `dist`, `kickofftransformationams`, `blobStorage.ts`, `encodingJobHelpers.ts`, `function.json`, `heavyduty.ts`, `index.ts` (selected), `sample.dat`, `transformFactory.ts`, `node_modules`, `.funcignore`, `.gitignore`, `host.json`, `local.settings.json`, `package-lock.json`, `package.json`, and `tsconfig.json`.

The Output/Debug Console at the bottom shows the command `moes@Moooorrrs-MacBook-Pro vsstudiogetgroovy %`.

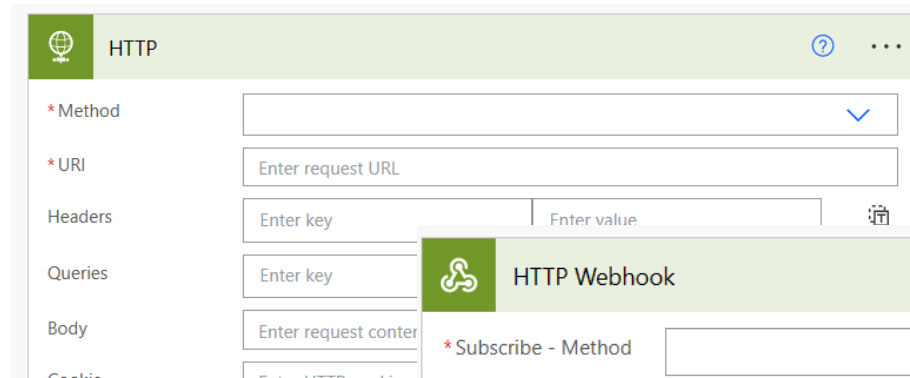
Mehr Details gefälltig?



[Using Logic Apps to orchestrate a complex video processing \(...\)](#)

THE GLUE MAKES IT STICK

APIS ★★ CONNECTORS ★★ WEBHOOKS ★★ MESSAGE QUEUES ★★ EVENT MGMT



HTTP

* Method

* URI

Headers

Queries

Body

Cookies



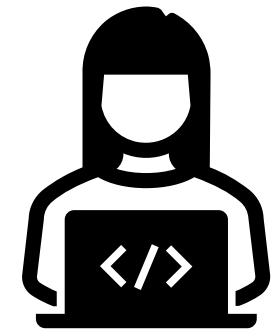
HTTP Webhook

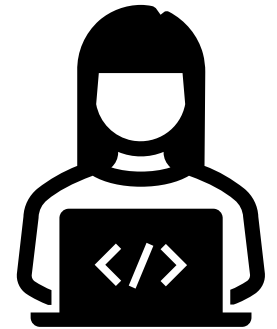
* Subscribe - Method

* Subscribe - URI

Subscribe - Body

Unsubscribe - Method







As a BIZ decision maker,
I am still worried all the
time.

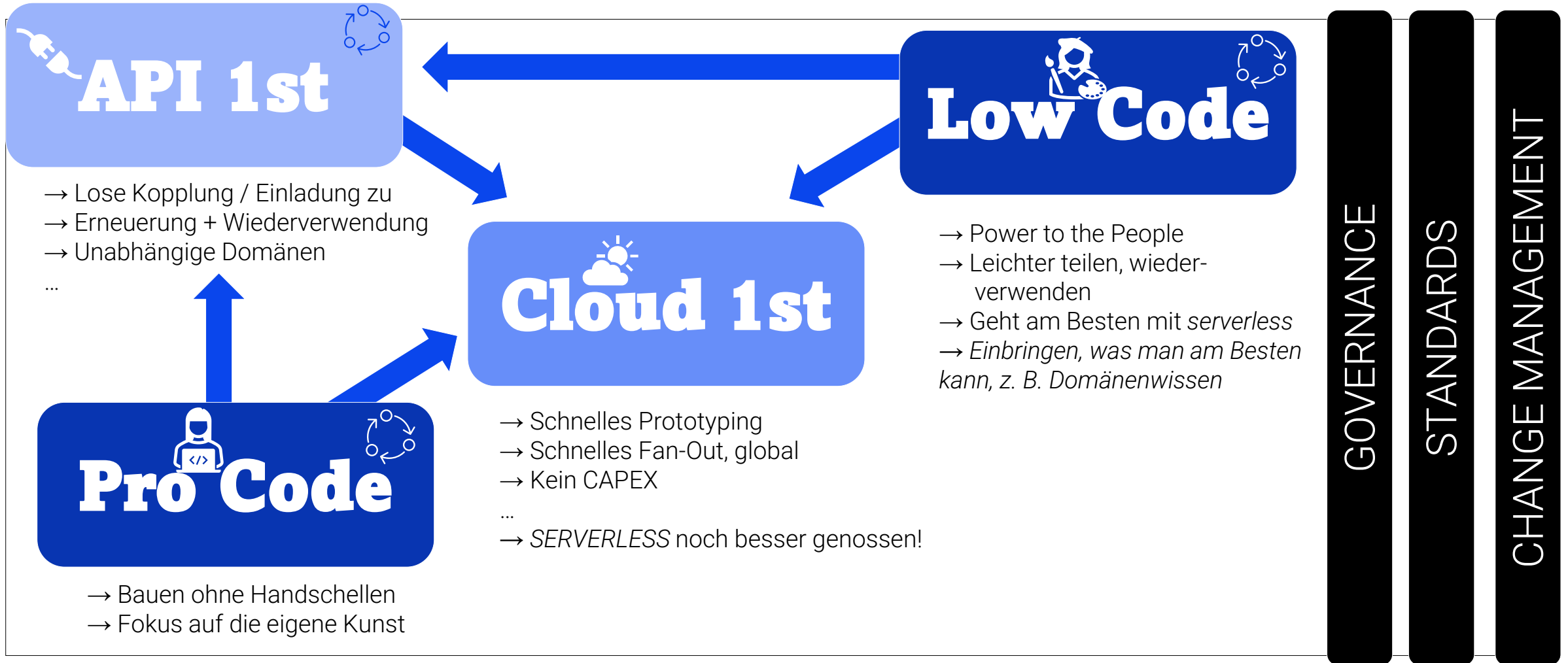
At least the
cross-functional
teams are complaining
a lot less now.

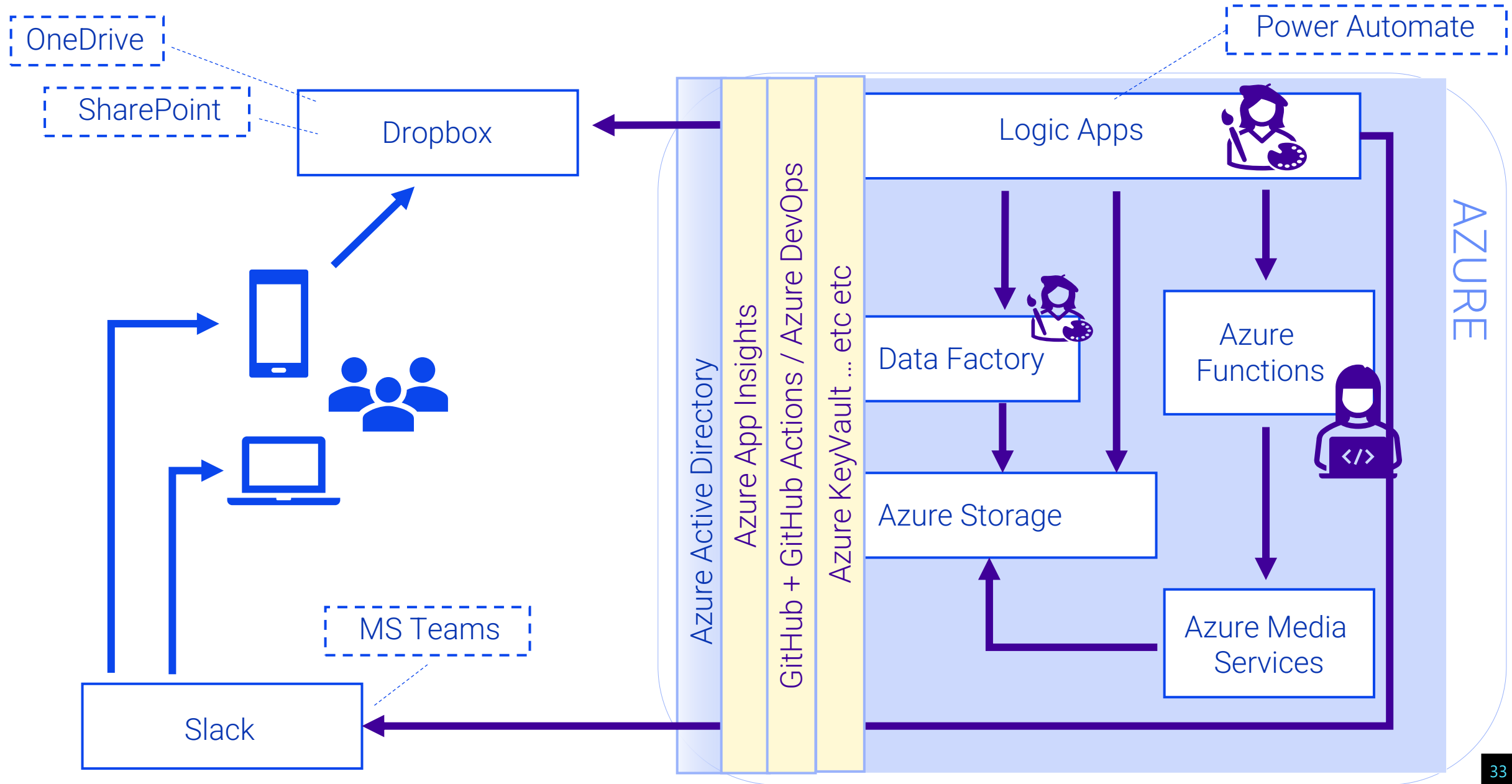
Hardened, proven
solution building blocks
and libraries make it
easier to get going – and
safer, too!



DIE REALITÄT IST EIN
BISSCHEN KOMPLEXER

Die "Secret Sauce" für den Erfolg von Fusion Teams





DIE MUSTER DER INTEGRATION SIND AUCH WICHTIG.

Sidecar, Circuit Breaker und viele mehr.

Ob Low-Code oder Pro-Code, Architekturentscheidungen haben IMMER eine große Auswirkung auf Kosten, Verlässlichkeit, Sicherheit usw... Aber das ist eine Geschichte für ein anderes Mal.

Die Präsentation mit ein paar Slides und Links mehr hier:

platformeconomies.com/azsummit23