Microsoft Azure

Project Batcomputer

A working DevOps implementation for Machine Learning

Ben Coleman **@BenCodeGeek**
Phil Harvey **@CodeBeard**

v0.0.2 (Alpha)

Background

Motivation

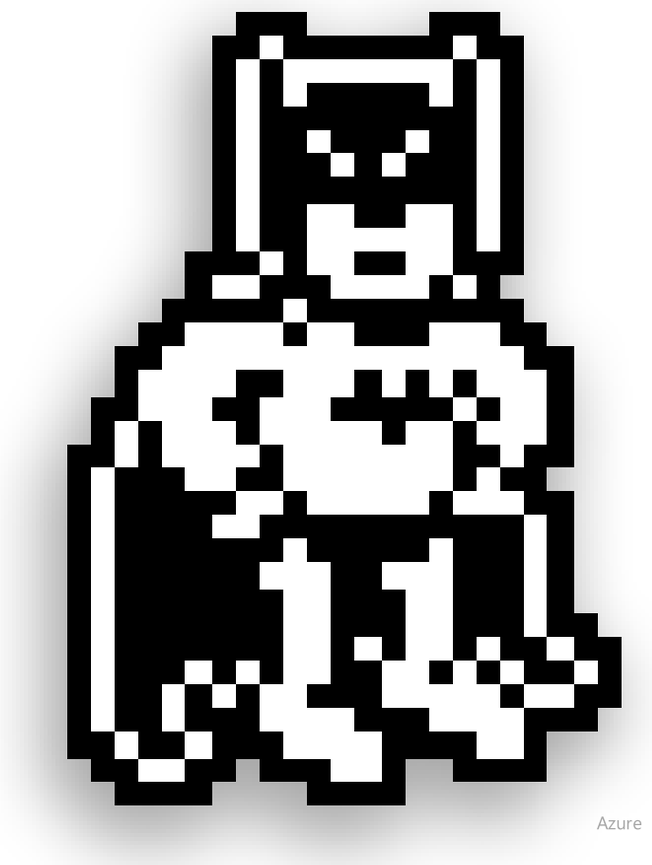
- Understand challenges in operationalisation of ML models
- Existing processes (e.g. Azure Machine Learning Service) deemed problematic
- “DevOps for AI”

Why Batcomputer?

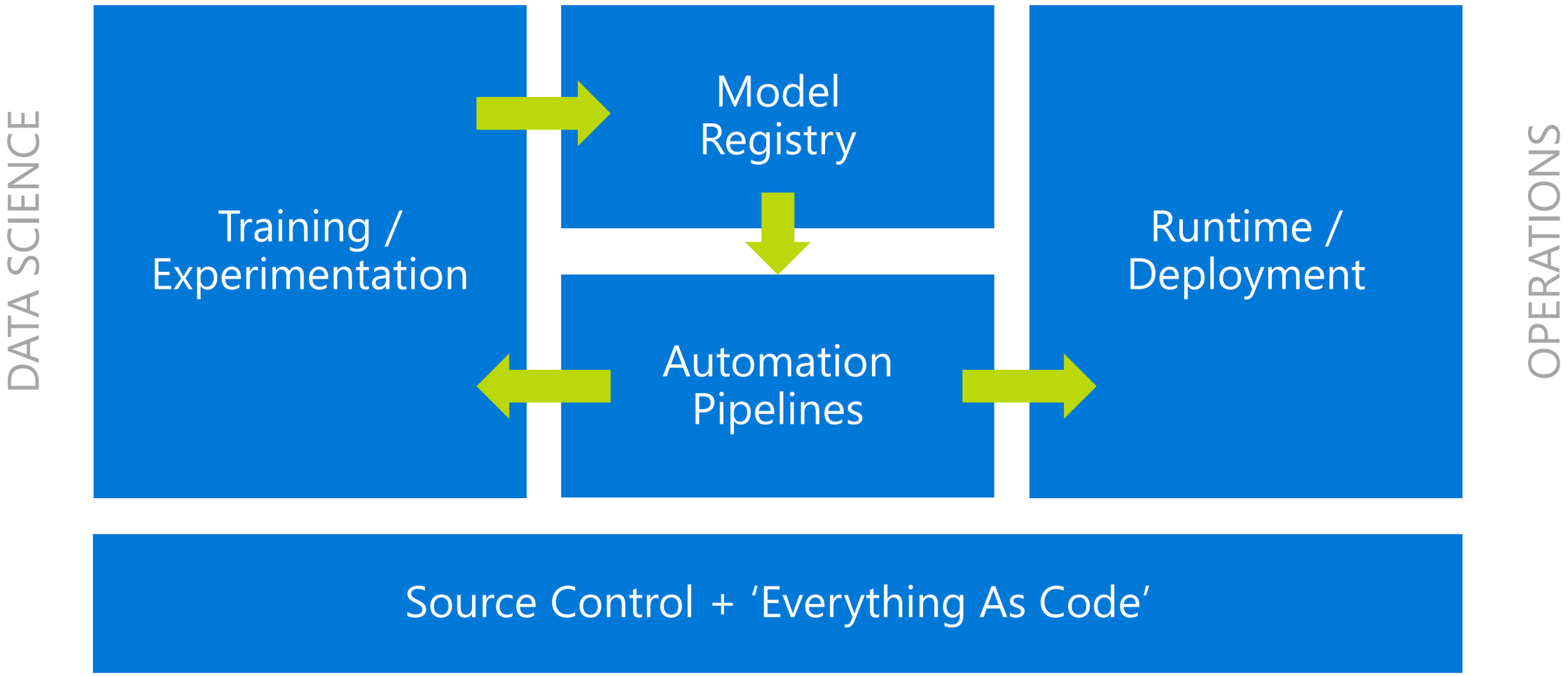
- Police recorded crime and outcomes open data tables
- https://www.gov.uk/government/statistics/police-recorded-crime-open-data-tables
- Build model of a given crime and/or region to predict outcome

Core Principals & Benefits

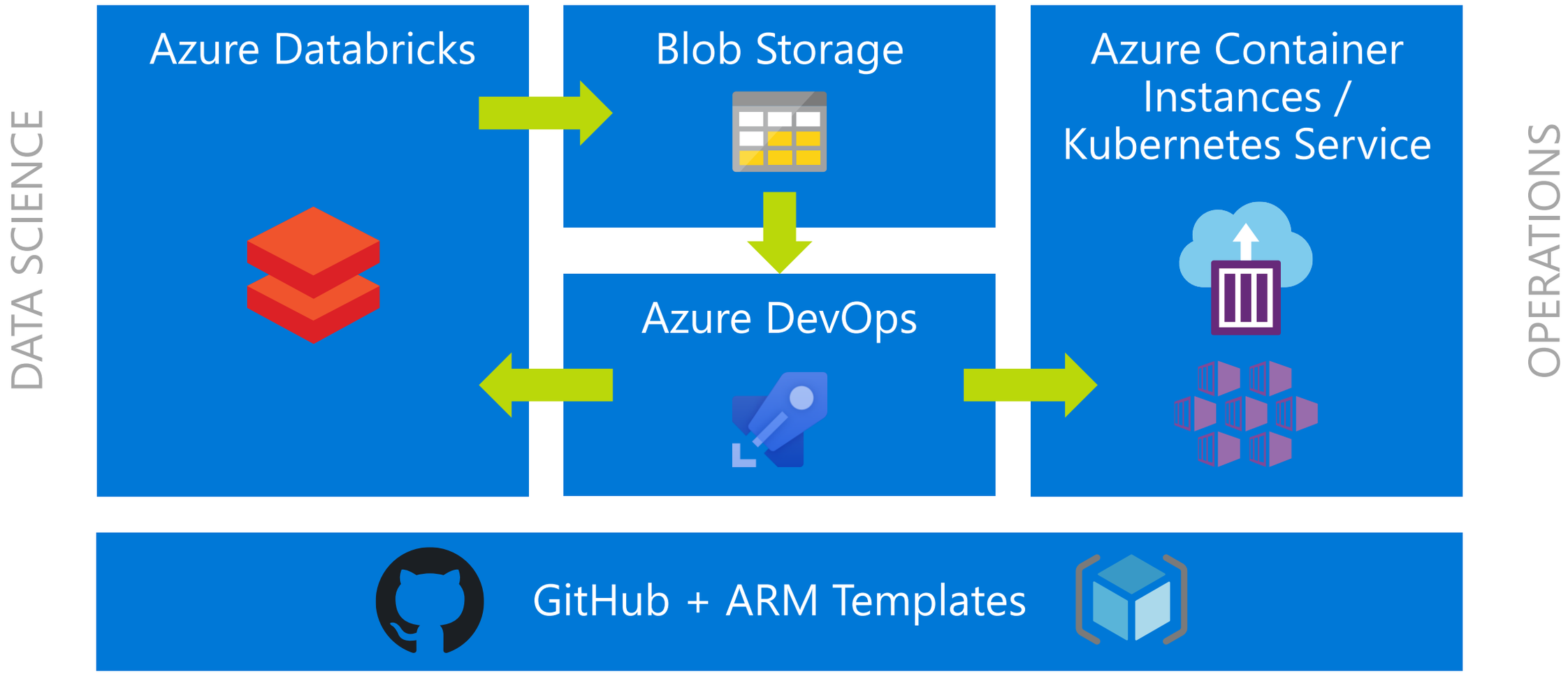
- Decouple model training experiments from operations/runtime
- Automated training, API builds & deployment
- Versioned models and API
- Config & infra as code
- Traceability



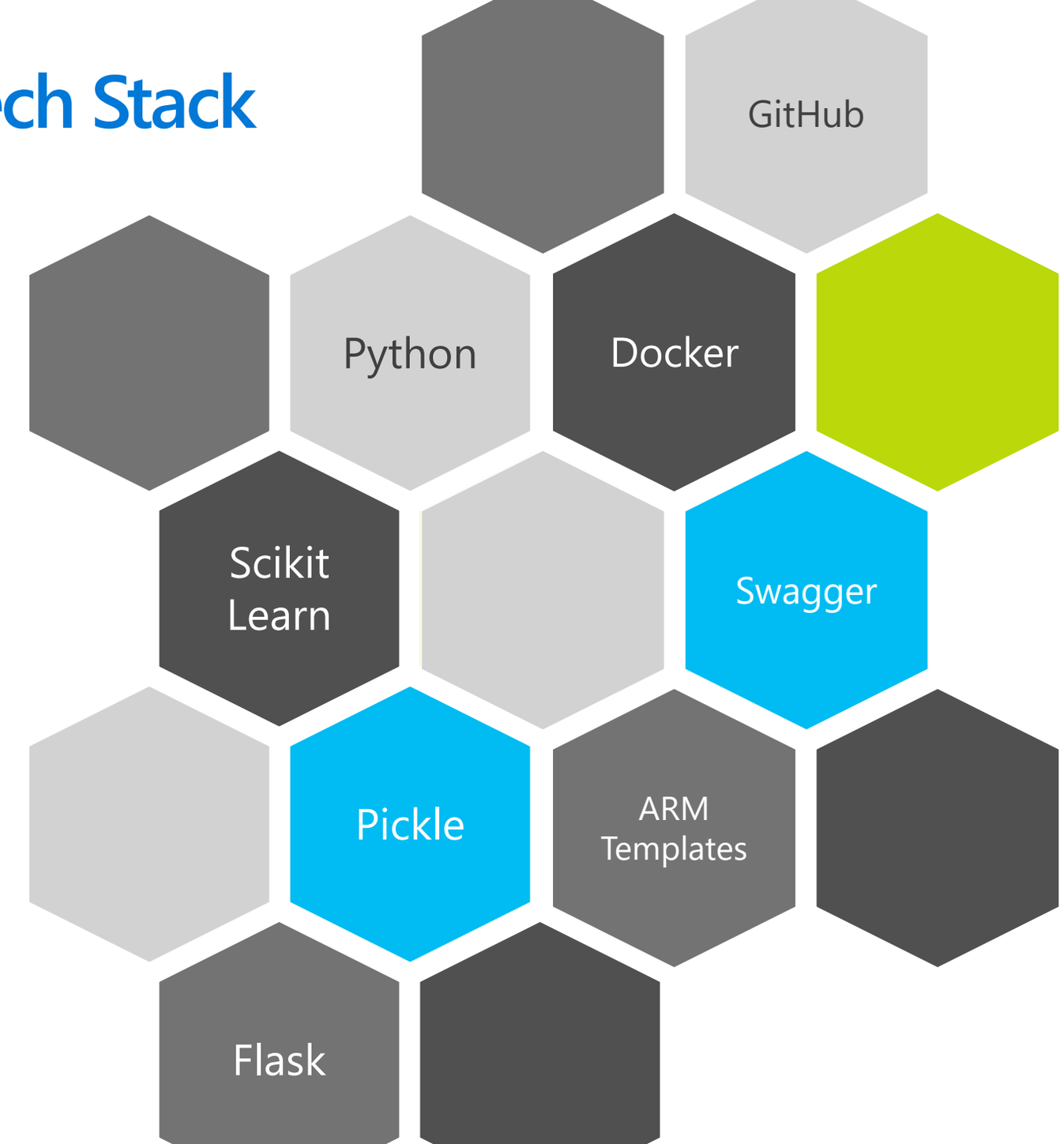
Conceptual Building Blocks



Conceptual Building Blocks – Project Batcomputer



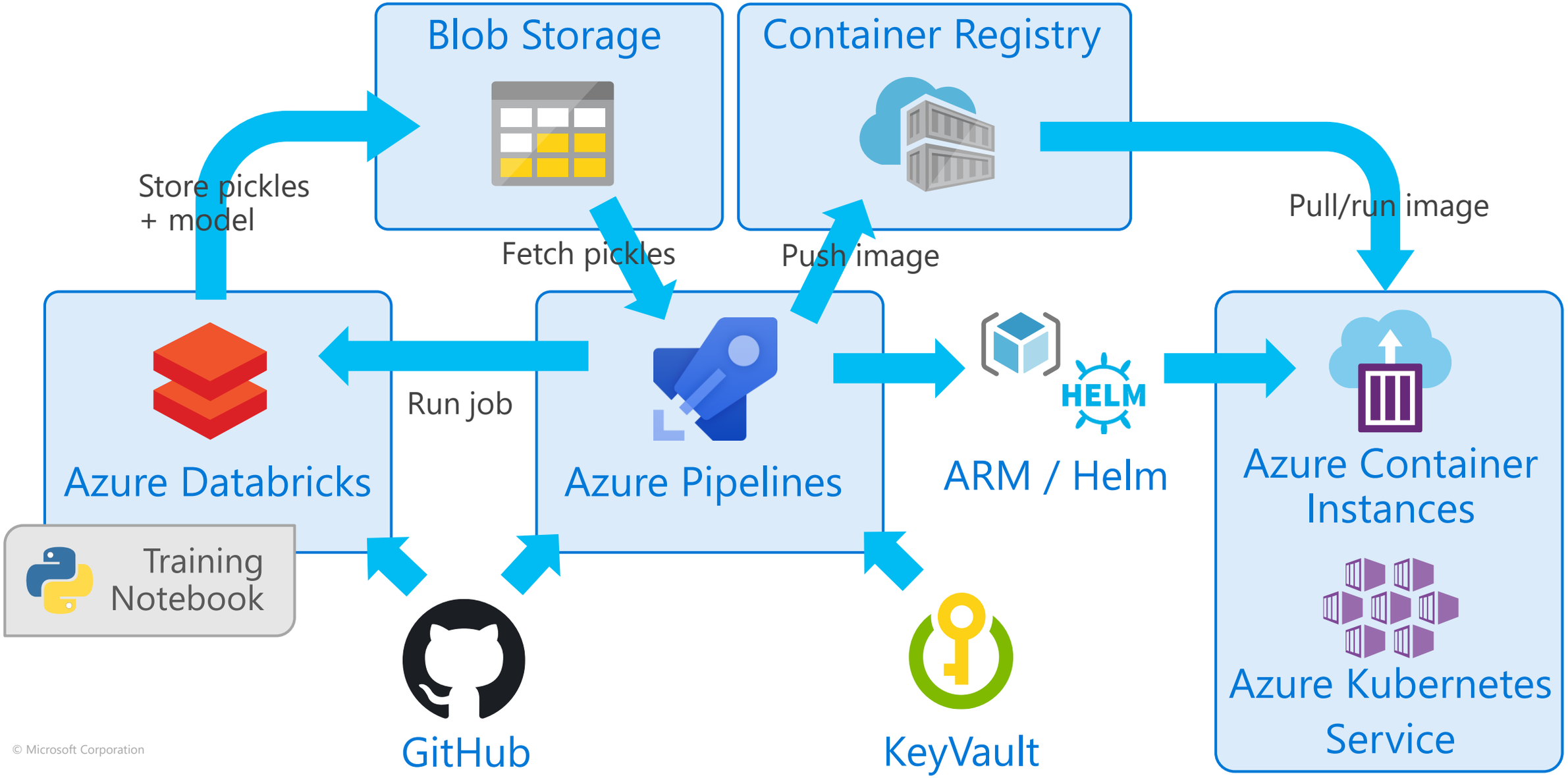
Low Level Tech Stack



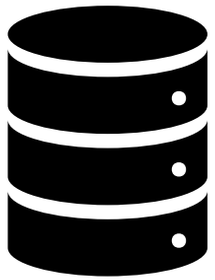
Low Level Technology Stack

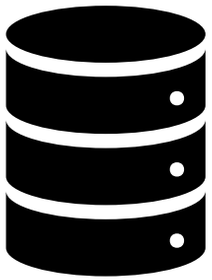
Swagger	API niceness
Flask	Web framework / server
Pickle	Serialisation
Scikit-Learn	Main ML framework
Python	Core language
Docker	Container Runtime

Automation – End To End Flow

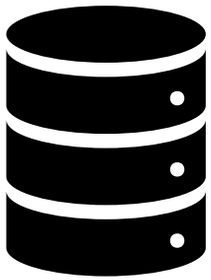


Pickling – Not Just The Model





```
{
“gender”: {
  “male”: 0,
  “female”: 1
}
```



```
[
  “conviction”,
  “dropped”,
  “settled”
]
```

model.pkl	lookup.pkl	flags.pkl
Scikit-learn model/classifier	Python dictionary of dictionaries	Python array
Standard object rehydration, version sensitive	Mapping parameters/strings to num for predict function	Maps output of prediction function to human readable strings or labels

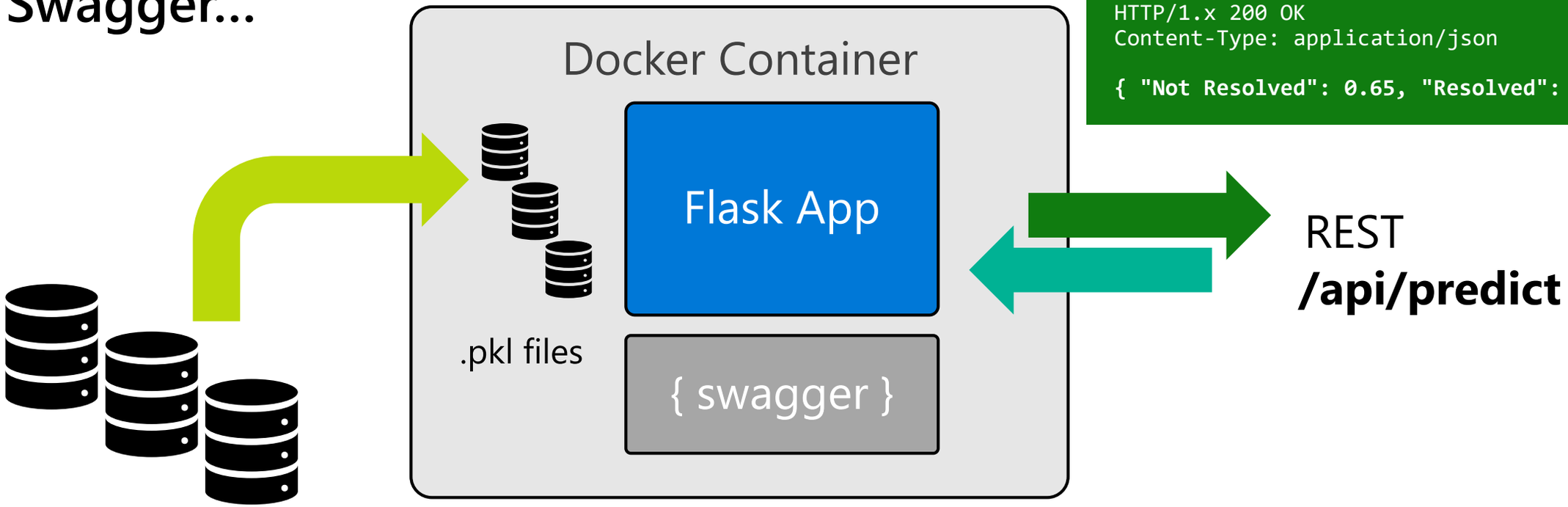
Wrapper App – Flask API

- Uses Flask web framework
- Creates RESTful API for model parameters
- Consumes .pkl files
- Swagger…

```
POST /api/predict
{
  “force_name”: “Suffolk”,
  “offence_description”: “Handling stolen goods”
}
```

```
HTTP/1.x 200 OK
Content-Type: application/json

{ "Not Resolved": 0.65, "Resolved": 0.35 }
```

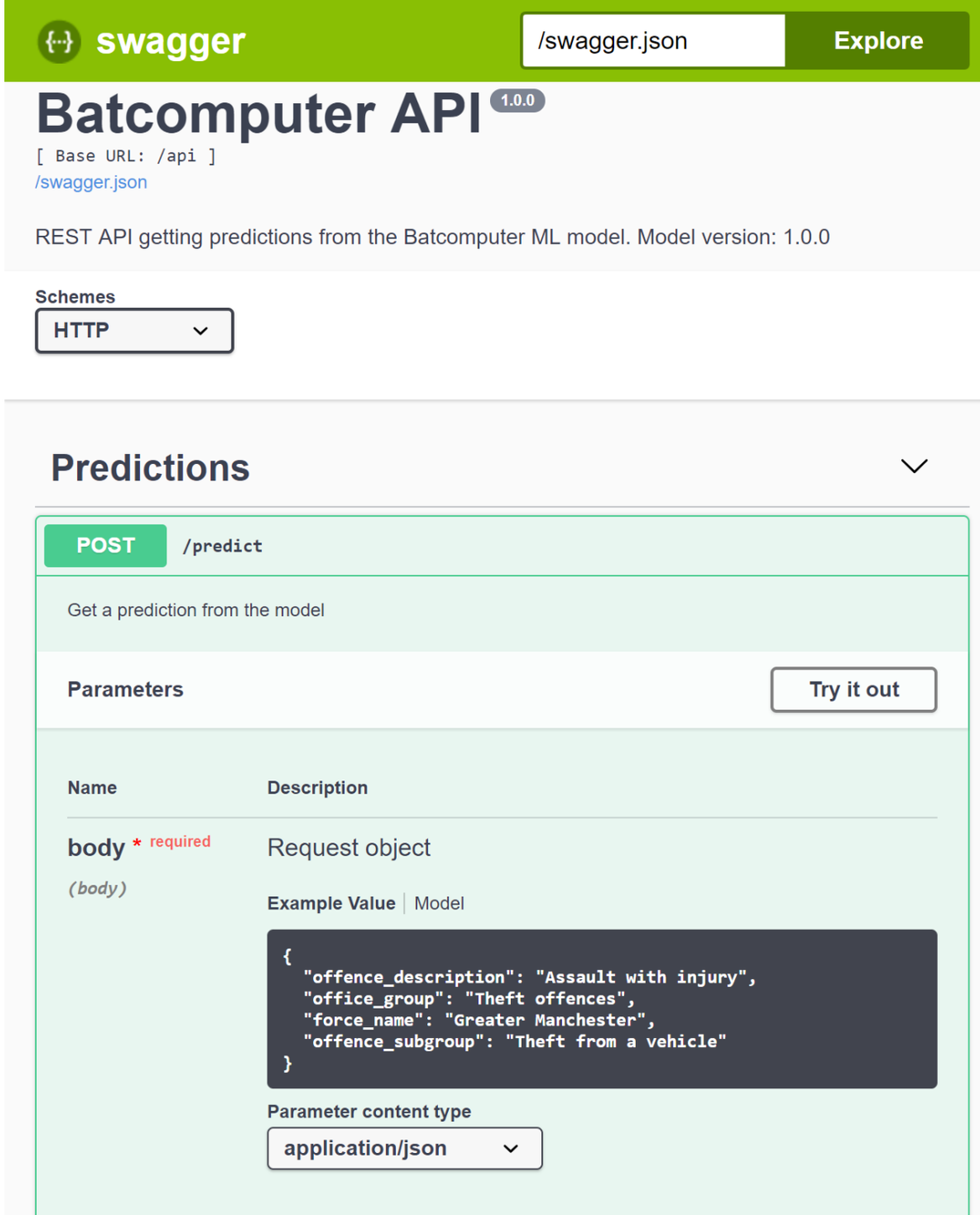


Some Decision Points

- Include model in container image or fetch at runtime?
- Wrapper app – make generic or tied into model?
- Convention based model registry – good enough?
- Use more robust web server than Flask?
- Automate training?

Swagger

- **We want to be RESTful**
- **Dynamic**
  - Generated from lookup & flags pickles at runtime
- **Swagger UI**
  - For testing & eye candy



OPENAPI INITIATIVE

Semantic Versioning

MAJOR Version

- Incompatible API or other breaking changes
- **Scoring parameter changes i.e. API change**

MINOR version

- Add functionality in a backwards-compatible manner
- **Trained using different parameters/classifiers, but API same**

PATCH version

- Backwards-compatible bug fixes
- **Trained the model on different data**

Versioning – Touches Everything

Blob Name
- /2.0.8/model.pkl

Pipeline Variables
- $version = 2.0.8

Docker Image Tag
- repo/batcomputer:2.0.8

Kubernetes API Path
- /v2.0.8/api/predict

Also…

- Resource names in Azure controlled via ARM templates
- ACI DNS names & prefixes,
  e.g. `batcomputer-2-0-8.westeurope.azurecontainer.io`
- Object names in Kubernetes (pods, services), controlled via Helm chart

Some Learnings

- Pickled Scikit-learn models are version sensitive
- Keep Python version in sync with DataBricks
- Installing numpy, scipy and scikit-learn is SLOW, pre-build base image
- Version number is the key parameter for the whole process
- Writing your own wrapper isn’t hard
- DataBricks is easy to use & has a great CLI

Microsoft Azure